**Common issues, fixes, and tips**

**Invalid selector method**

Check the current context at the time, use the view switching functions at the bottom of First\_Test.java

**Yellow XCUI error (iOS only)**

Can be caused by an android simulator being open whilst trying to run ios test. Can also be caused by a java library conflict. Check only one Appium java library exists in your project.

**Text appearing and disappearing in text fields (iOS only)**

Occurs only in IOS (in my experience, see email field in iOS), when in webview trying to enter text into a field it disappears. It’s a firing issue, on appiums side. Until permanently fixed a workaround is to use the xpath of the field. (xpaths should generally be avoided as they are not transferable between different compilations of the app)

**Invalid class name**

If using a classname as an identifier for an element there must be no spaces. If it is the only available argument for finding an element, use css selector and edit your class name as following:

“Hello I am bored” becomes “.Hello.I.am.bored”

**Errors switching to webview in android**

Check you have the correct chromedriver. It usually gives in error some helpful advice.

**Touch actions not working in iOS 7 or 8**

If you decide to test on these (not a lot of point) then XCUI Automator does not support TouchAction functions. Use javascript mobile:scroll and a hashmap as a workaround. (google javascript mobile:scroll Appium should be enough to give you some good help)

**Touch Actions seemingly not firing, or firing at the wrong place**

If a touch action does not appear to be firing, but is not giving an error, add a delay, or increase it, between pressing and releasing.

It can also fire where you didn’t intend, often this is because when you get an elements location, it gives you its top left coordinates. Use .getCenter() to avoid this, and it could also be that the element is wrong or you have not successfully found an element.

**Locating Strategy**

I try to find an element in this order:

Use Appium desktop, and open the inspector, ideally find acc id, id, or name

If it’s a webelement, on ios open safari, develop, simulator, localhost.html. use classname, or name usually. On android, open chrome and go dev tools, devices, android x86

Often there will be multiple elements with the same identifier. List them and then use a property of the element. If it’s a label you can generally use .getText() to find it, particularly in webview. If in mobile view use .getAttribute() or .isVisible. Location can also be used, for example if you know it should be the left most element.

**Scrolling to elements**

I’ve struggled to get mobile:scrollTo to work, however you really don’t need to scroll if all you want to do is click on an element. In a scroll view, elements off the page are still findable. So find the element you want, click on it and it will scroll to for you