Rahul Shetty Appium

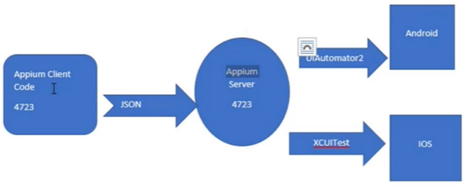
06 February 2022 12:09 PM

What is Appium:
Open source Mobile Automation framework
The only cross platform (Andriod,IOS) testing supporting tool
Works for native(Only to Andriod/IOS) ,hybrid(Can run on andriod as well as mobile browser) mobile apps
Supports Webdriver API-selenium family

Supported platforms:
IOS
Android
Firefox OS

Supported languages
Java,C#, Javascript, Python, Ruby

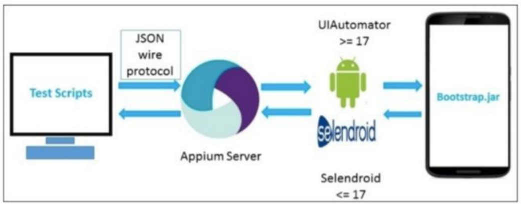
Appium Architecture



UIAutomator2 is a framework release by Andriod team to automate Andriod app.Similarly XCUITest for IOS

Browserstack definition:

Appium uses the UIAutomator framework (or Selendroid), which is meant for testing the Android user interface to automate applications on Android device testing. The *bootstrap.jar* file works as a TCP server that sends the test command for acting on the Android device with the help of the UIAutomator or Selendroid framework.



Steps to install:

1)Download Java JDK and set the JAVA_HOME path till navigating inside jdk folder
2)Download Andriod studio.The SDK will be in hidden folder Appdata C:\Users\kkrid\AppData\Local\Android\Sdk
3)Set environment variable ANDRIOD_HOME and paste path till sdk identified in step 2
4)Add under path variables the following values:C:\Users\kkrid\AppData\Local\Android\Sdk\tools , C:\Users\kkrid\AppData\Local\Android\Sdk\tools\bin, C:\Users\kkrid\AppData\Local\Android\Sdk\platform-tools
4)Download node(software) as it required for running appium server
6)Create a new System variable NODE_HOME and its path C:\Program Files\nodejs.Also add this under path variables C:\Program Files\nodejs\node_modules\npm\bin

Note:The 6 point to add node env variable set up as told by rahul did not work .Instead refer https://stackoverflow.com/questions/23412938/node-is-not-recognized-as-an-internal-or-an-external-command-operable-program

In command promt run as adminstrator "npm install -g appium"

Type command appium to start the server

Download the APK used for testing from https://github.com/appium/appium/blob/master/sample-code/apps/ApiDemos-debug.apk

Note:This worked only with Java version 11 selenium <version>3.141.59</version> appium <version>7.2.0</version>

To check the compatible appium and selenium version check https://github.com/appium/java-client changelogs

TO start uiAutomaterviewer navigate to C:\Users\kkrid\AppData\Local\Android\Sdk\tools\bin or in the command prompt enter uiautomatorviewer

If Uiautomator is throwing error use adb kill-server adb start-server or kill the appium server and start again uiautomator

For xpath //tagname[@attribute=''] tagname will be class field in uiautomator
Valid location strategies xpath, id, class name, accessibility id(IOS), css selector, -android uiautomator

Location strategies appium :https://www.browserstack.com/guide/locators-in-appium

Android Uiautomator syntax driver.findElementByAndroidUIAutomator("properties(\"value\")") [Eclipse doesn't shows up as autosuggestion]

Scrolling till the element is found

First way : driver.findElementByAndroidUIAutomator("new UiScrollable(new UiSelector()).scrollIntoView(text(\"Views\"));"); where Views is the text

Second Way: driver.findElement(MobileBy.AndroidUIAutomator("new UiScrollable(new UiSelector().scrollable(true).instance(0)).scrollIntoView(new UiSelector().textMatches(\"" + containedText + "\").instance(0))"));

Third way:Find the parent and scroll to the desired element : driver.findElement(MobileBy.AndroidUIAutomator("new UiScrollable(new UiSelector().resourceId(\"com.androidsample.generalstore:id/rvProductList\")).scrollIntoView(new UiSelector().textMatches(\"Jordan 6 Rings\").instance(0))"));

Gestures

1)Tap

```
TouchAction action = new TouchAction(driver);
WebElement viewMenu = driver.findElement(By.xpath("//android.widget.TextView[@text='Views']"));
		action.tap(TapOptions.tapOptions().withElement(ElementOption.element(viewMenu))).perform();
```

2)LongPress

```
action.longPress(LongPressOptions.longPressOptions().withElement(ElementOption.element(first))
				.withDuration(Duration.ofSeconds(10))).moveTo(ElementOption.element(second)).release().perform();
```

3)Drag And Drop

```
action.longPress(ElementOption.element(Source)).moveTo(ElementOption.element(dest)).release().perform();
```

Note:If there is no chaining of actions you can directly use as example 3 and also if you import static method you can direclty use it .ie instead of ElementOption.element(Source) we can direclty use element(Source)

Steps to install a .apk in virtual/real device

In cmd Navigate to andriod->sdk->platform-tools

Adb install apk apkfilefullpath

Eg: adb install c/downloads/test.apk

TO check the devices attached
adb devices

Toast message cannot be inspected.The workaround is that developer will always use "android.widget.Toast" class for toast message.
And use attribute "name" to get the text of toast msg
Eg:

```
String toastMessgae=driver.findElement(By.xpath("//*[@class='android.widget.Toast']")).getAttribute("name");
```

If there are multiple toast message displayed then use indexing xpath locator strategy

Bu default appium will have chromedriver for automating webApp/Hybrid app. Default location :C:\Users\kkrid\AppData\Roaming\npm\node_modules\appium\node_modules\appium-chromedriver\chromedriver\win
You can also pass it as desiredCapability

```
cap.setCapability("chromedriverExecutable", "C:\\Users\\kkrid\\AppData\\Roaming\\npm\\node_modules\\appium\\node_modules\\appium-chromedriver\\lib\\chromedriver.exe");
```

TO get the context ie either native app,webView

```
Set<String> context = driver.getContextHandles();
		System.out.println(context);

driver.context("WEBVIEW_com.androidsample.generalstore");
```

For back pressing android

```
driver.pressKey(new KeyEvent(AndroidKey.BACK));
```

Switch back to native appdriver.context("NATIVE_APP");

Note:w3c standard support only xpath and css as locators. Appium does not have internal redirection if id ,classname used.Hence it will throw error "error":"invalid argument","message":"invalid argument: invalid locator https://github.com/appium/appium/issues/13306

To inspect a webview
1)Using Chrome devtool in browser
GO to chrome://chrome-urls/   .Select inspect
2)Using chromebrowser inbuilt app ui viewer displayed ie toggle device toolbar (While inspect element)

Appium official test examples: https://github.com/appium/java-client/tree/master/src/test/java/io/appium/java_client/android

http://appium.io/docs/en/writing-running-appium/tutorial/swipe/simple-element/

RealTime example :Section 7
1)Fill the form and verify Toast message displayed of incomplete input
2)shop the items by scrolling
3)Validate the items added are available in cart and matching the selected
4)validate the total price in cart page
5)Validate the mobile gestures
6)Validate on swithing to webview and returning back to native view after some operations

You need App Activity and App package to be passed if you want to automate an installed app

cap.setCapability("appPackage", "com.android.contacts");
cap.setCapability("appActivity", "com.android.contacts.activities.DialtactsActivity");

Refer here how to get App Activity and App package https://www.automationtestinghub.com/apppackage-and-appactivity-name/

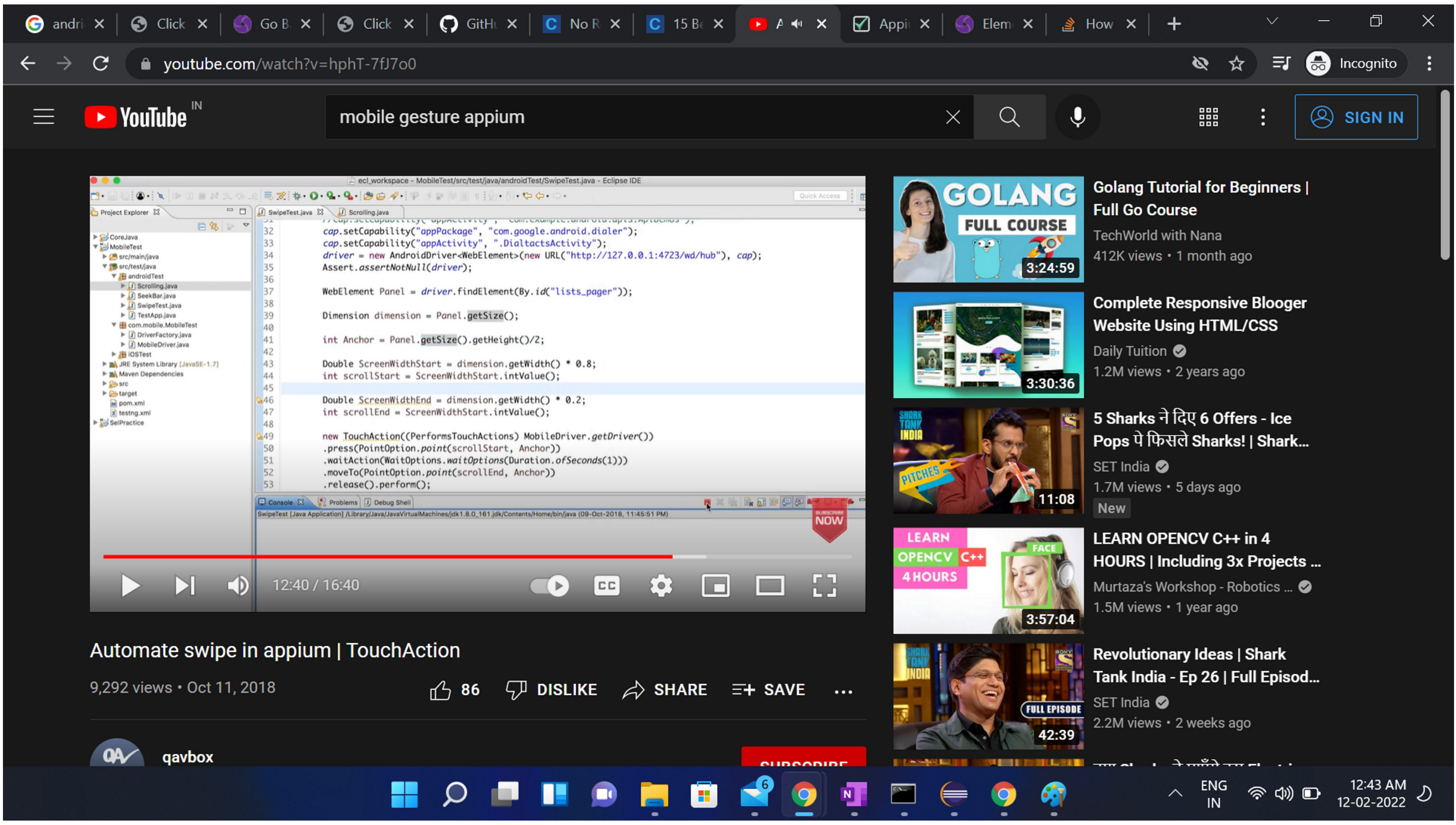
Refer this tutorial for Appium: https://qavalidation.com/appium-tutorial/ .Note:All mobile gesture tutorial are nicely given

Scrolling gesture : https://qavalidation.com/2016/07/scrolling-in-appium.html/
Seek bar: https://qavalidation.com/2016/07/how-to-move-drag-seek-bar-in-appium.html/
Swiping : https://qavalidation.com/2016/07/swipe-on-android-device-using-appium.html/

Framework:
To start a server appium server automatically ,Some dependencies are required .Please check pom.xml
To start emulator go to emulator folder and command emulator -avd emulator name

Dependencies Appium ,selenium compactible version need to be checked in appium documentation ,testng
Usb debugging

Adb stands for android debug bridge



youtube.com/watch?v=hphT-7fJ7o0

mobile gesture appium

ecl_workspace - MobileTest/src/test/java/androidTest/SwipeTest.java - Eclipse IDE

```
cap.setCapability("appPackage", "com.google.android.dialer");
cap.setCapability("appActivity", ".DialtactsActivity");
driver = new AndroidDriver<WebElement>(new URL("http://127.0.0.1:4723/wd/hub"), cap);
Assert.assertNotNull(driver);

WebElement Panel = driver.findElement(By.id("lists_pager"));

Dimension dimension = Panel.getSize();

int Anchor = Panel.getSize().getHeight()/2;

Double ScreenWidthStart = dimension.getWidth() * 0.8;
int scrollStart = ScreenWidthStart.intValue();

Double ScreenWidthEnd = dimension.getWidth() * 0.2;
int scrollEnd = ScreenWidthStart.intValue();

new TouchAction((PerformsTouchActions) MobileDriver.getDriver())
.press(PointOption.point(scrollStart, Anchor))
.waitAction(WaitOptions.waitOptions(Duration.ofSeconds(1)))
.moveTo(PointOption.point(scrollEnd, Anchor))
.release().perform();
```

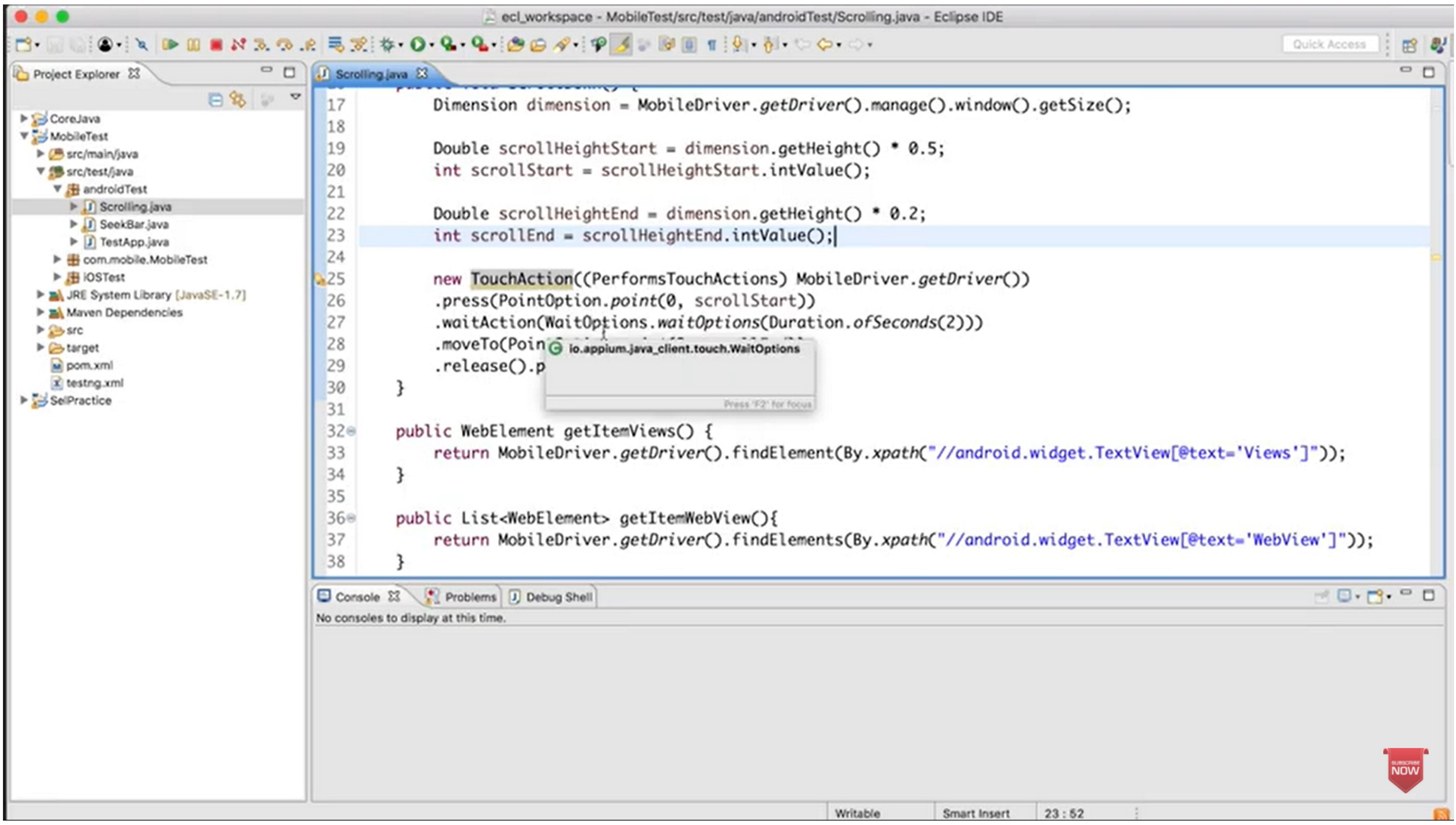
SwipeTest [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_161.jdk/Contents/Home/bin/java (09-Oct-2018, 11:45:51 PM)

Automate swipe in appium | TouchAction

9,292 views • Oct 11, 2018

qavbox

12:40 / 16:40
86
DISLIKE
SHARE
SAVE
SIGN IN
Incognito
SUBSCRIBE NOW
Golang Tutorial for Beginners | Full Go Course
TechWorld with Nana
412K views • 1 month ago
Complete Responsive Blogger Website Using HTML/CSS
Daily Tuition
1.2M views • 2 years ago
LEARN OPENCV C++ in 4 HOURS | Including 3x Projects ...
Murtaza's Workshop - Robotics ...
1.5M views • 1 year ago
Revolutionary Ideas | Shark Tank India - Ep 26 | Full Episod...
SET India
2.2M views • 2 weeks ago
12:43 AM
12-02-2022



```java
        Dimension dimension = MobileDriver.getDriver().manage().window().getSize();

        Double scrollHeightStart = dimension.getHeight() * 0.5;
        int scrollStart = scrollHeightStart.intValue();

        Double scrollHeightEnd = dimension.getHeight() * 0.2;
        int scrollEnd = scrollHeightEnd.intValue();

        new TouchAction((PerformsTouchActions) MobileDriver.getDriver())
        .press(PointOption.point(0, scrollStart))
        .waitAction(WaitOptions.waitOptions(Duration.ofSeconds(2)))
        .moveTo(Poin
        .release().p
    }

    public WebElement getItemViews() {
        return MobileDriver.getDriver().findElement(By.xpath("//android.widget.TextView[@text='Views']"));
    }

    public List<WebElement> getItemWebView(){
        return MobileDriver.getDriver().findElements(By.xpath("//android.widget.TextView[@text='WebView']"));
    }
```

No consoles to display at this time.