Automation testing iOS Web App with Appium Guideline

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Revision history | | | | | | |
| # | Version | Description of Change | Created By | Revision Date | Approved By | Effective  Date |
| 1 | 0.1 | First Draft | An Nguyen | Jun-26-14 | - |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

Contents

[1. Introduction 3](#_Toc391996864)

[2. Requirements 4](#_Toc391996865)

[3. Test configuration 5](#_Toc391996866)

1. Introduction

This document describes step by step how to test automation using Appium for iOS web Application.

Appium is an open source automation tool for native and hybrid apps. It supports for both Android and iOS platforms. Appium drives the device/emulator/simulator using the Apple's UIAutomation library or Android's UiAutomator framework using Selenium's Webdriver JSON wire protocol. So we can use the languages for scripting, which all supported by Selenium.

1. Requirements
   * 1. MAC OS X 10.7 or higher.
     2. Xcode and Xcode command line tools (Xcode 5.0 or higher).
     3. Install JDK – version 1.6 or above: <http://www.oracle.com/technetwork/java/javase/downloads/index.html?ssSourceSiteId=otnjp>
     4. Install Eclipse <http://www.eclipse.org/downloads/download.php?file=/technology/epp/downloads/release/kepler/SR2/eclipse-standard-kepler-SR2-macosx-cocoa-x86_64.tar.gz>
     5. Download Selenium and Appium Libraries:
        + java-client-1.1.0.jar: <http://mvnrepository.com/artifact/io.appium/java-client>
        + selenium-server-standalone-2.40.0.jar: <http://selenium-release.storage.googleapis.com/2.40/selenium-server-standalone-2.40.0.jar>
2. Test configuration
3. **Install Node js with home brew**

We need node.js to work with appium. Just follow the below steps:

* First, install Homebrew:

ruby -e "$(curl -fsSL <https://raw.github.com/Homebrew/homebrew/go/install>)"

* Then, brew update to ensure your Homebrew is up to date.

brew update

* Next, add the Homebrew location to your $PATH and source your bash or zsh profile file after adding/saving this:

export PATH="/usr/local/bin:$PATH"

* Next, install Node:

sudo brew install node

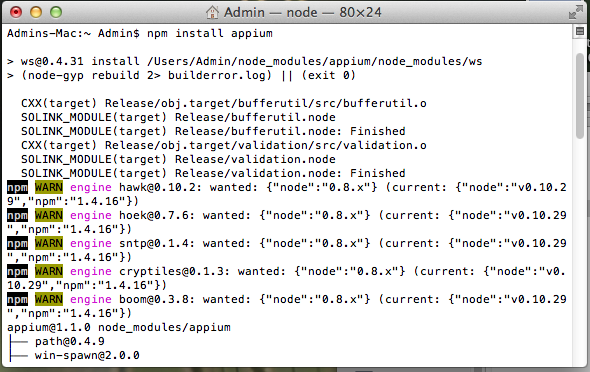
If this command failed, try :

sudo brew postinstall node

Check npm existed on /usr/local/bin. If not try:

ln –sf /usr/local/lib/node\_modules/npm/bin/npm-cli.js /usr/local/bin/npm

1. **Install and running Appium:**

* Install Appium by npm

npm install -g appium

Check exist file : appium, appium-doctor, authorize-ios on /urs/local/bin

If not, try:

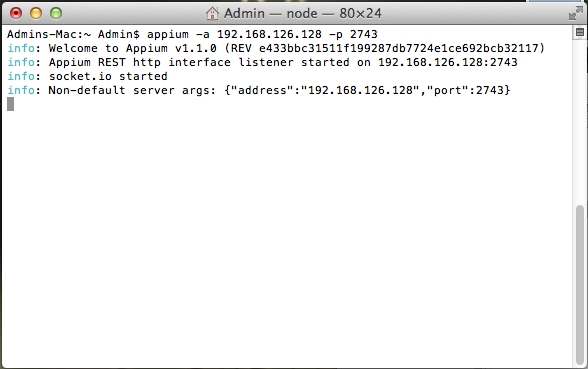
ln -sf /Users/<user>/node-modules/appium/bin/appium.js /usr/local/bin/appium

ln -sf /Users/<user>/node-modules/appium/bin/appium-doctor.js /usr/local/bin/appium-doctor

ln -sf /Users/<user>/node-modules/appium/bin/authorize-ios.js /usr/local/bin/authorize-ios

* Launch Appium:

appium -a 127.0.0.1 -p 4725



1. **Prepare the automation script with Java and Eclipse IDE:**

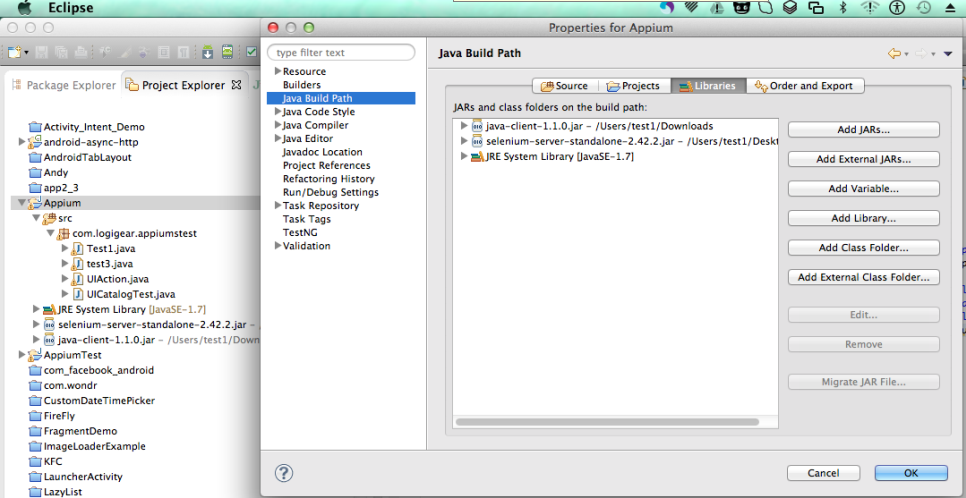
Download Selenium and Appium Libraries:

* + - 1. java-client-1.1.0.jar or higher: <http://mvnrepository.com/artifact/io.appium/java-client>

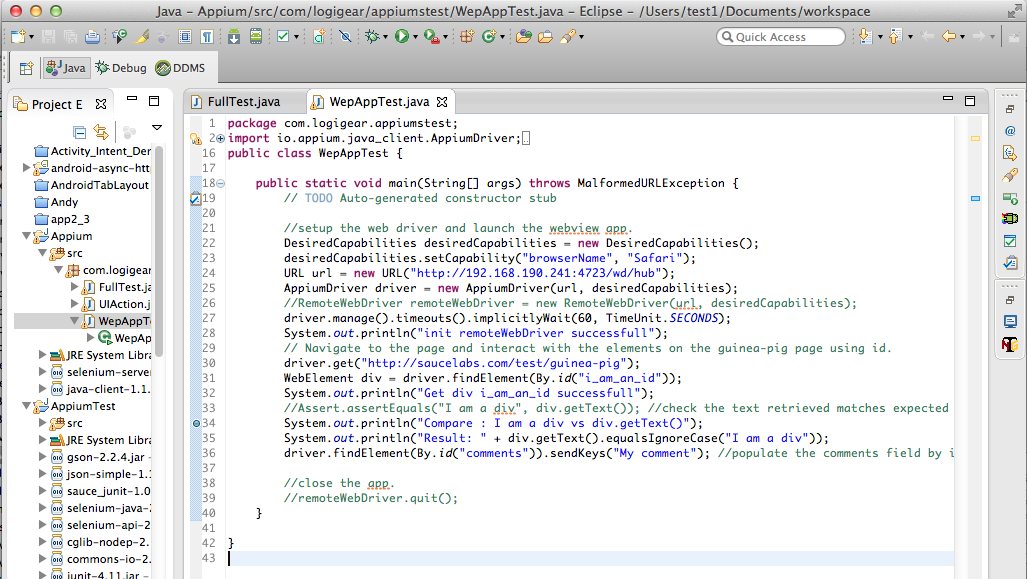
2. selenium-server-standalone-2.40.0.jar or higher: <http://selenium-release.storage.googleapis.com/2.40/selenium-server-standalone-2.40.0.jar>

Open Eclipse and create Java Application Project

Click right on Project and choose Properties, click on Java Build Path, click on Add External JARs…, and import two files just downloaded.



Next, create a class and implement source code as below



Make sure iOS simulator opened.

Build and run Project to execute test script.

1. **Running Appium iOS with selenium grid:**

- Create nodeconfig.json file with content:

{

"capabilities":

[

{

"browserName":"Safari",

"version":"7.1"

}

],

"configuration":

{

"cleanUpCycle":2000,

"timeout":30000,

"proxy":"org.openqa.grid.selenium.proxy.DefaultRemoteProxy",

"url":"http://192.168.190.241:4723/wd/hub",

"maxSession": 1,

"port": 4723,

"host":"192.168.190.241",

"register": true,

"registerCycle": 5000,

"hubPort": 4444,

"hubHost":"192.168.190.200"

}

}

- Run command:

appium –nodeconfig <path>/nodeconfig.json

1. **Running Appium iOS on real device:**

- Install ios webkit debug proxy:

cd /usr/local

git checkout 7e209f0 Library/Formula/libimobiledevice.rb

brew unlink libimobiledevice

brew install libimobiledevice

- Launch ios webkit debug proxy and appium

Open terminal to run command:

ios\_webkit\_debug\_proxy –c <udid> -d

Open new other terminal to launch appium

appium –a <address> -p <port> --app com.apple.mobilesafari –U <udid>

**Note: Make sure real device enable safari Web Inspector**