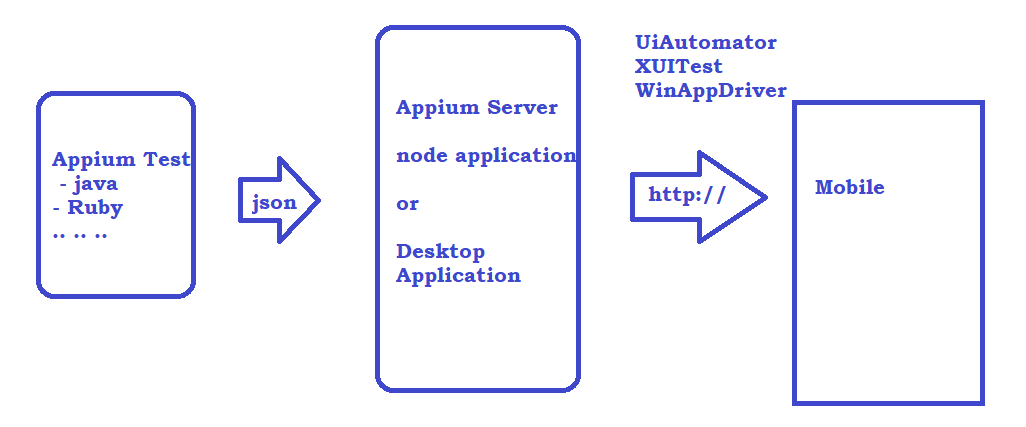
Appium

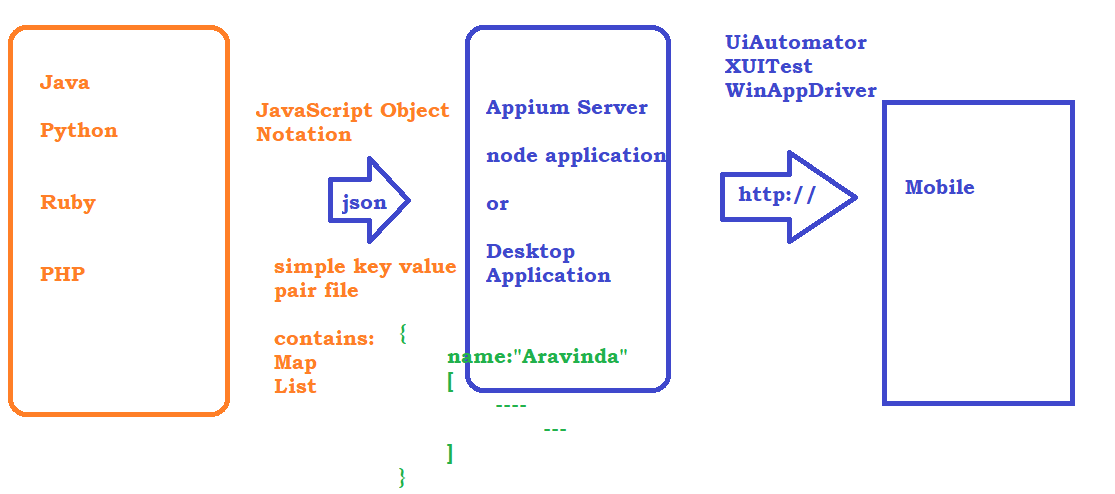
# Feature:

* Open Source free Software
* Supports Android and iOS
* Native Apps, Hybrid Apps, Web Apps
* Supports multiple programming language
  + Java
  + Ruby
  + Python
  + C#
* Internally Appium uses Selenium WebDriver APIs

## Architecture :



# Detailed Json Wired Protocol:

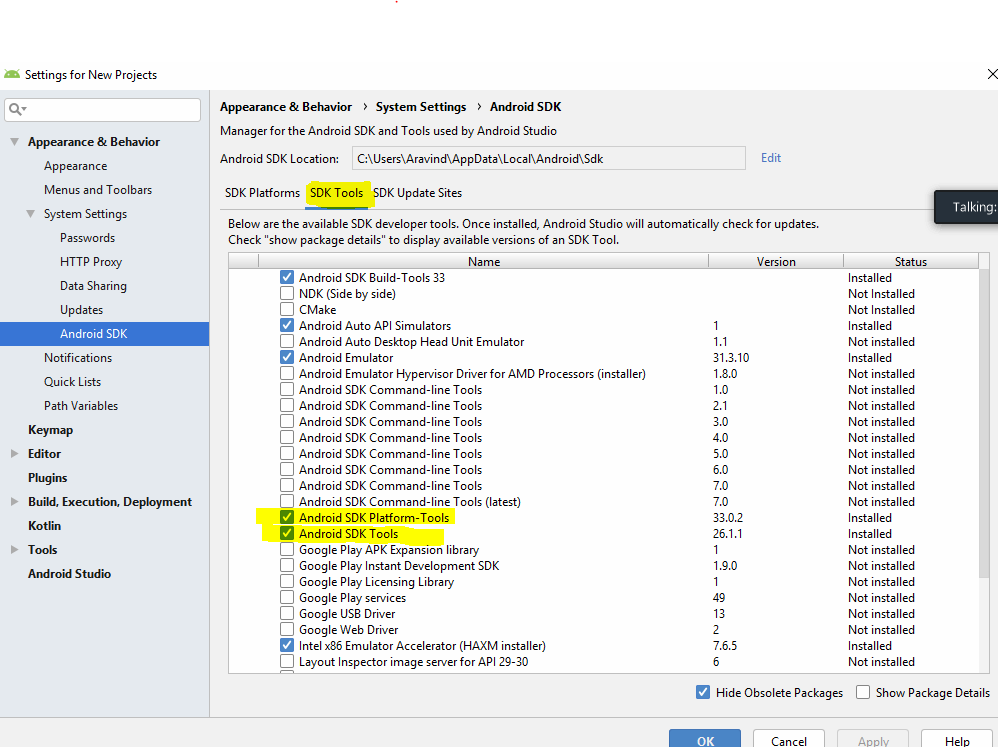


Appium Setup

# Java JDK 1.8

* 1. Download and install JDK from the official website <https://www.oracle.com/in/java/technologies/javase/javase8-archive-downloads.html>
  2. JAVA\_HOME - C:\Program Files\Java\jdk1.8.0\_201
  3. PATH : C:\Program Files\Java\jdk1.8.0\_201\bin

# Download and install Android Studio

1. <https://developer.android.com/studio?gclid=Cj0KCQjwr4eYBhDrARIsANPywChadzh-SC4CG7JIUsbT4pPXrm97nOJBu7FEIZlhuXSFnbwFG46hbzUaAnC0EALw_wcB&gclsrc=aw.ds>
2. Android Studio comes with 2 PACKAGES
   1. Android SDK : All the required software to run emulator
   2. Android Studio : Create a Virtual Device
3. Create a Virtual Device – Open Android Studio and create an empty project with **Empty Activity** Option
4. Once the build is complete Goto Tools -> Click on SDK Manager and make sure below options are selected under SDKTools
5. 
6. By Default SDK will be installed inside C:\Users\Aravind\AppData\Local\Android\Sdk
7. SET ENVIRONMENT VARIABLES
   1. ANDROID\_HOME : C:\Users\Aravind\AppData\Local\Android\Sdk
   2. Update P ATH :
      * C:\Users\Aravind\AppData\Local\Android\Sdk\tools
      * C:\Users\Aravind\AppData\Local\Android\Sdk\tools\bin
      * C:\Users\Aravind\AppData\Local\Android\Sdk\platform-tools
   3. ***abd*** *- C:\Users\Aravind\AppData\Local\Android\Sdk\platform-tools*
   4. ***uiautomateviewer*** *- C:\Users\Aravind\AppData\Local\Android\Sdk\tools\bin*
   5. ***emulator*** *- C:\Users\Aravind\AppData\Local\Android\Sdk\tools*

# Create a new Virtual Device

1. Go to Android Studio

2. Click on Tools -> AVD Manager OR Device Manager

3. Click on Create Virtual Device

4. Select the required phone and click on next

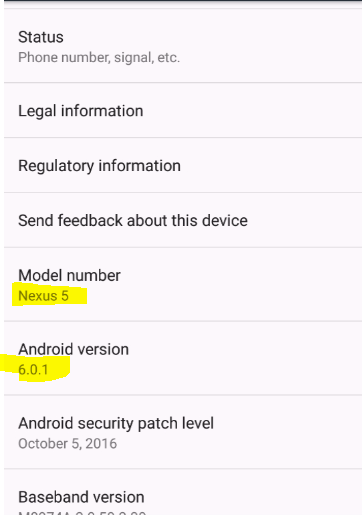
5. Download the required OS Version for the phone click on next

* 1. Provide AVD Name - >Aravind
  2. click on Advanced settings and increase or decrease the ram
  3. click on finish
  4. One the device by clicking on Run button

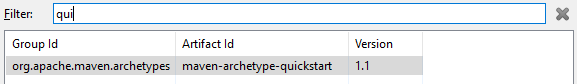
# Install Appium Server

* Install using node software
* <https://nodejs.org/en/download/>
* PATH : C:\Program Files\nodejs\
* PATH: C:\Users\Aravind\AppData\Roaming\npm
* Open command prompt and execute node –v and npm –v
* go to <https://www.npmjs.com/> and serach fro Appium
* from command prompt npm i Appium
* OORORORRRRRRRRRRRRRRRRRRRRRRRRRRRr
* <https://github.com/appium/appium-desktop/releases/tag/v1.22.3-4>
* <https://github.com/appium/appium-desktop/releases/download/v1.22.3-4/Appium-Server-GUI-windows-1.22.3-4.exe>

# Enable USB Debugging option on Real Device

* go to settings on the real device
* click on about phone
* on the build number tap for 7-8 times
* 

# Create a Project and configure Appium

1. Create a maven project, while creating select a template maven-archetype-quickstart
2. Update the correct version of JRE and JDK if required
3. update the POM with Appium client libs
   1. https://mvnrepository.com/artifact/io.appium/java-client/7.5.1
4. update the POM with Selenium client libs
   1. https://mvnrepository.com/artifact/org.seleniumhq.selenium/selenium-java/3.141.59
5. update the POM with TestNG client libs
   1. https://mvnrepository.com/artifact/org.testng/testng/6.14.3

# Installing App on a Virtual Device

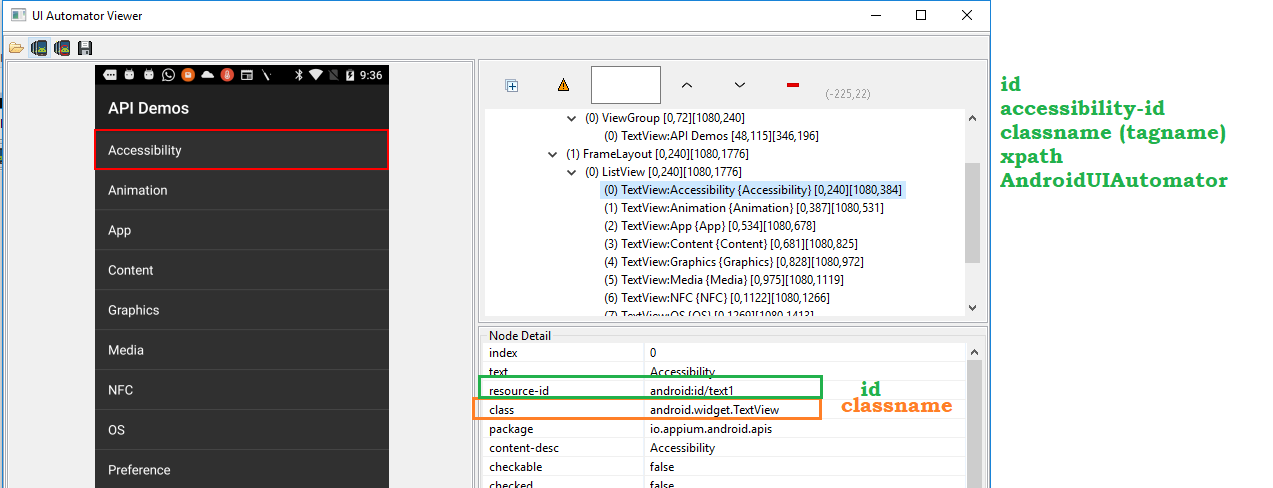
from the command prompt execute

adb install <<APK file location>>

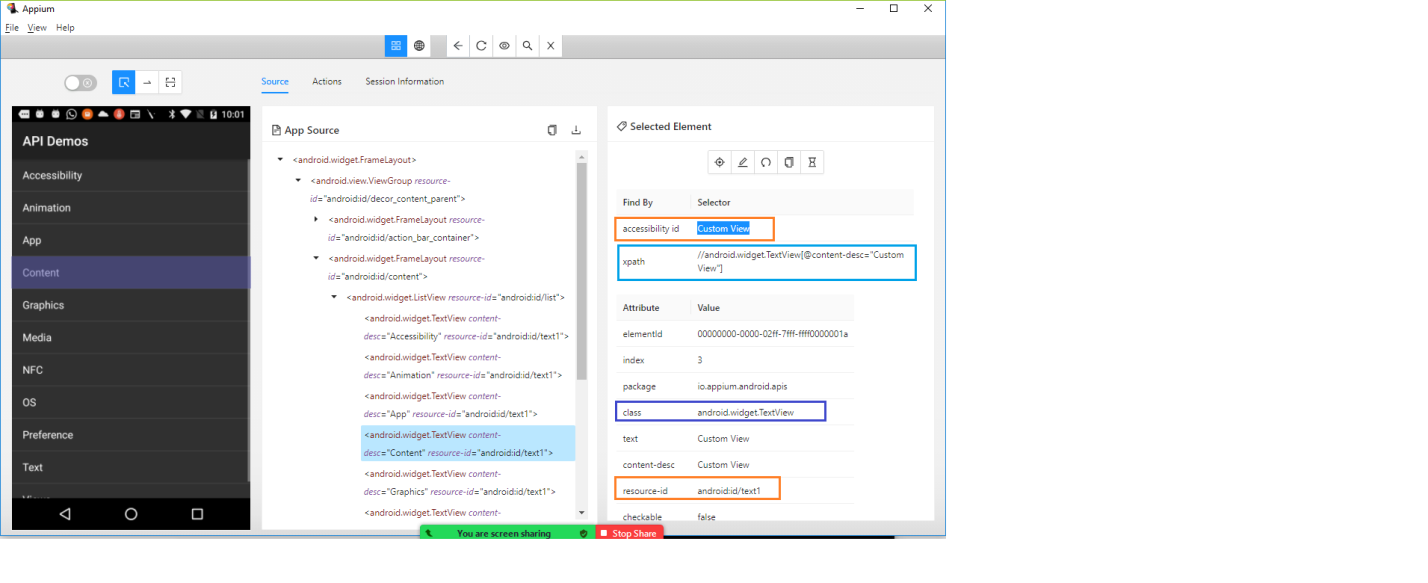
Object Identification Stratergy

# 1. UI Automator Viewer

To launch automator viewer go to C:\Users\Aravind\AppData\Local\Android\Sdk\tools\bin and double click on uiautomatorviewer



# 2. Appium Inspector

* Download the Appium inspector from the official website <https://github.com/appium/appium-inspector>
* Click on releases and download the required version <https://github.com/appium/appium-inspector/releases>
* Open Appium inspector and add the desired capabalities
  + deviceName
  + udid
  + platformName
  + app
* 

Launching App using package and activity

* install APK info on your device
* Open the app from the APK Info for which you want to get the apppackage and appactivity

# Getting package and activity name from command:

from the command prompt execute

**adb shell dumpsys window | find "mCurrentFocus"**

# Starting the Virtual Device from command prompt

cd C:\Users\Aravind\AppData\Local\Android\Sdk\emulator

emulator -avd <<DEVICE\_NAME>>

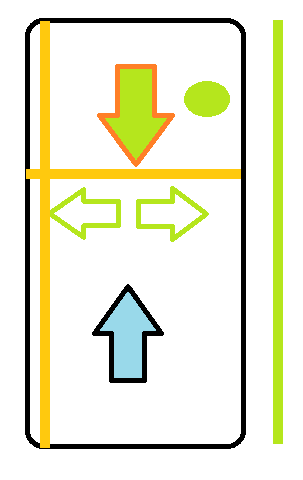
ex : emulator -avd Aravind

AndroidUIAutomator

when we want to find the element using android ui automator , we can use  [UiSelectors](http://developer.android.com/reference/android/support/test/uiautomator/UiSelector.html). [UiScrollable](http://developer.android.com/reference/android/support/test/uiautomator/UiScrollable.html)  or String

# String : attribute(“value”)

Playing with co-ordinates



Advanced Gestures - can be performed by 2 ways

1. Long Press
   1. Touch Actions
   2. Javascript Executor
      1. <https://github.com/clarabez/appium-1/tree/master/docs/en/writing-running-appium/android>
2. scroll
3. swipe
4. drag and drop
5. Tap / click

# Static Imports :



Advanced AndroidUIAutomator

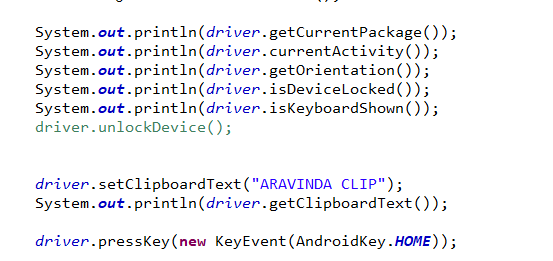
- text(“value”)

- <https://appium.io/docs/en/writing-running-appium/android/uiautomator-uiselector/index.html#uiautomator-uiselector>

- <https://developer.android.com/reference/android/support/test/uiautomator/UiSelector.html>

Ui Scrollable - "new UiScrollable(new UiSelector()).scrollIntoView(text(\"WebView\"))"

Miscellaneous



Type of apps

* Native Apps
* Hybrid Apps
* Web Apps

Enable internet on Virtual device

<https://www.seleniumeasy.com/appium-tutorials/configure-appium-to-run-on-real-android-device-over-wifi>

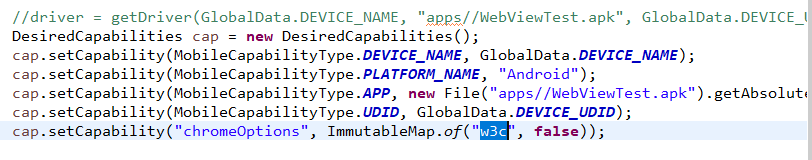
appium --chromedriver-executable <<location>>

appium --chromedriver-executable D:\synechron\_appium\_22\_08\Eclipse\_WS\MobileAutomation\drivers\chromedriver.exe

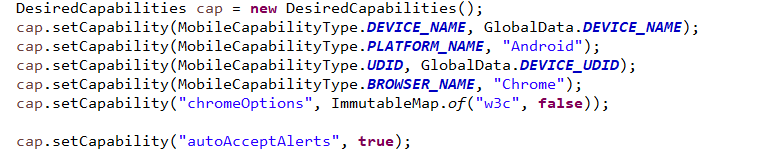
Executing tests on Hybrid app

* start the server with chromedriver enabled

Mandatory capability - cap.setCapability("chromeOptions", ImmutableMap.*of*("w3c", **false**));



Browser Automation



Maven commands to execute tests

* Update pom with surefire plugin
  + <https://maven.apache.org/surefire/maven-surefire-plugin/examples/testng.html>
* Update pom with compiler plugin
  + <https://maven.apache.org/plugins/maven-compiler-plugin/examples/set-compiler-source-and-target.html>
* Update JRE to use from JDK

Executing tests from Batch file

