**Appium**

Appium is mobile web, native and hybrid software application test automation tool developed and supported by Sauce Labs.

<https://www.youtube.com/watch?v=yTW7hZZfTYo&list=PLUDwpEzHYYLsx_2JFNBMITjHqTnuszhb_>

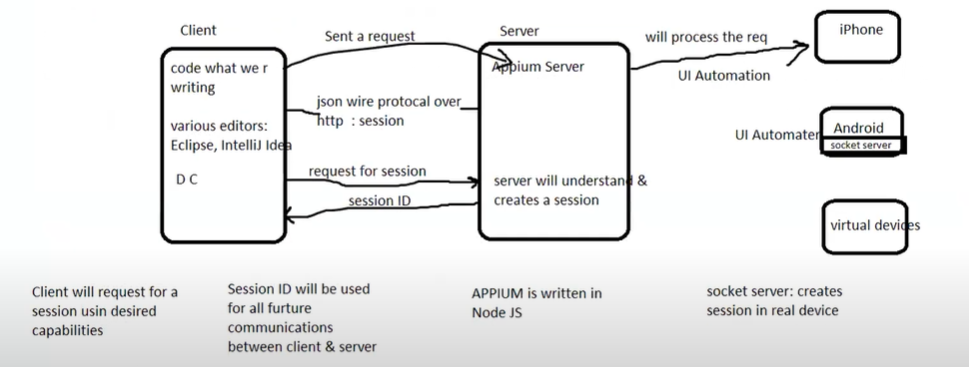
Advantages 🡪

* Appium can be used on Android & iOS platforms.
* It supports all web, native and hybrid apps.
* Allows to communicate with other apps like WhatsApp.
* Support for build in apps like alarm, calendar etc
* Any WebDriver compatible language is supported like Java, Ruby, C# etc

Limitations 🡪

* Android API level must be greater than 17 (means Android version 4.1).
* Script execution is very slow on iOS & android virtual devices.
* No parallel execution directly, but we can handle it using sauce labs.
* No support for Toast messages.

Appium Architecture 🡪



* Appium is basically a server written using Node JS.
* We write our code in client using any editor like eclipse.
* ‘json wire protocol over http’ is used to communicate between client and server.
* Using json wire protocol, client sends request to server and server will create a session & session ID for this client.
* WebDriver will request the session and server will give session ID to client and thus connection gets established. So whatever actions we perform on client, same actions will be done by the server on particular devices like iPhone, Android phone, virtual devices.
* iPhone uses ‘UI Automation’ internally to perform actions.
* Android phone uses ‘UI Automater’ internally to perform actions.

Pre- requisites of Appium 🡪

* Java
* Eclipse
* Maven
* TestNG
* Selenium Standalone Server
* Android Studio (it is an IDE but also provides SDK and virtual users)
* Appium Server / Appium Desktop
* Appium Client Library (available in Jar files)

For developing android application we need SDK (like for Java application we need JDK).

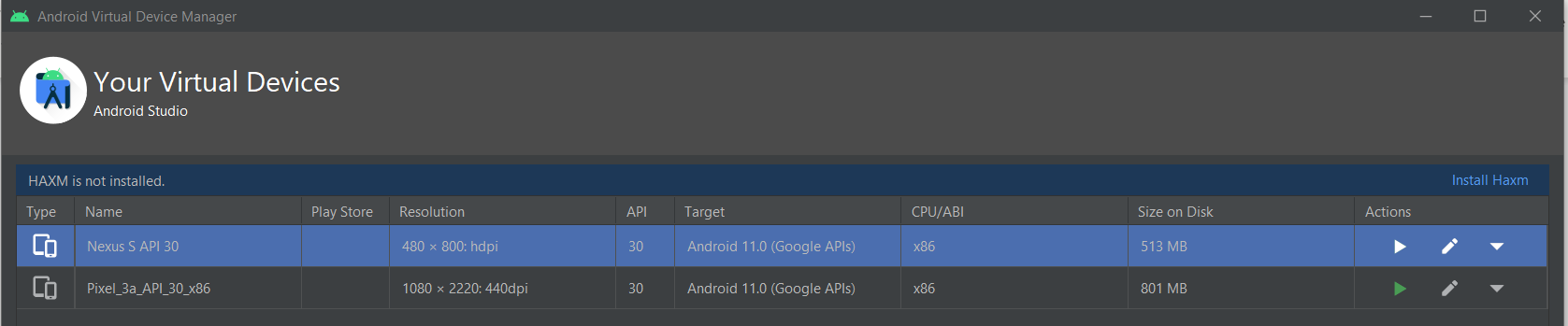
APK (Android Application Package) is similar to .exe file which are used to install program on windows. APK files are explicitly designed to run on Android devices.

Mobile testing can be done on three different devices namely,

* Real device
* Emulator (Android virtual device)
* Simulator (iOS virtual device)

**Launching virtual device 🡪**

From Android studio, virtual device is created using ADV Manager.



Click on Run option under Actions column to launch virtual mobile device.

Now launch the Appium server.

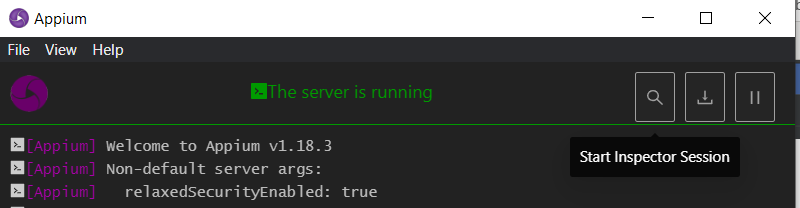
Open cmd at C:\Users\VJ\AppData\Local\Android\Sdk\tools\bin and type ‘adb.exe devices’, it will list the connected devices.

**Different ways of installing APK on devices (real / virtual) 🡪**

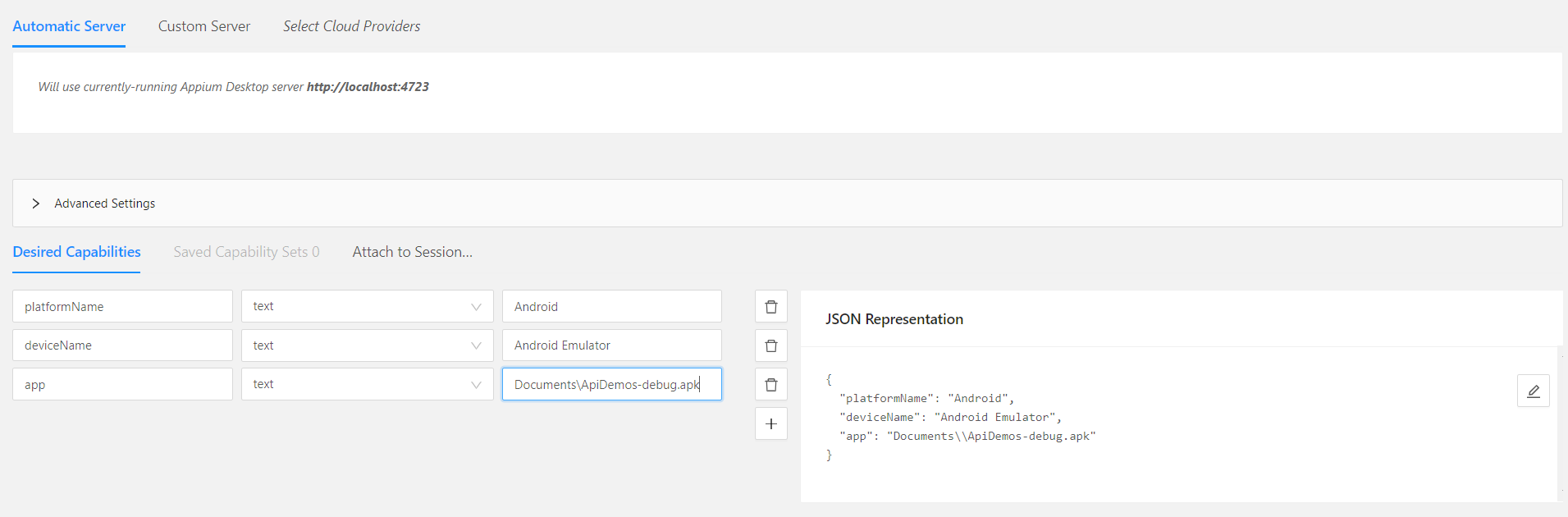
* Using Appium Server Desktop (Inspection session)
* Using adb.exe
* Using program

**Installing APK on virtual device manually🡪**

In Appium server, click on ‘Start Inspector Session’.



Provide the desired capabilities.



Note – Desired capabilities are documented at <http://appium.io/docs/en/writing-running-appium/caps/index.html>

Click on Start Session, it will get installed on device.

**Elements can be identified using,**

* Appium server inspector
* SDK uiautomator (available in installed SDK folder)

**Installing APK on virtual device using JAVA Program**🡪

To configure Eclipse for Appium, we need to download following dependencies 🡪

* Appium Client Library
* Selenium stand alone server
* Commons\_lang3

Complete project is at C:\Users\VJ\Desktop\Switch\Mobile\_Testing\Project\Appium\_Test

Before this make sure Appium server and virtual device are up.

**public** **class** Install\_App {

**public** **static** **void** main(String[] args) **throws** MalformedURLException {

DesiredCapabilities dc = **new** DesiredCapabilities();

dc.setCapability(MobileCapabilityType.***AUTOMATION\_NAME***, "Appium");

dc.setCapability(MobileCapabilityType.***PLATFORM\_NAME***, "Android");

dc.setCapability(MobileCapabilityType.***PLATFORM\_VERSION***, "11.0");

dc.setCapability(MobileCapabilityType.***DEVICE\_NAME***, "Android Emulator");

dc.setCapability(MobileCapabilityType.***APP***, "C:\\Users\\VJ\\Desktop\\Switch\\Mobile\_Testing\\ApiDemos-debug.apk");

URL url = **new** URL("http://127.0.0.1:4723/wd/hub");

AndroidDriver<WebElement> driver = **new** AndroidDriver<WebElement> (url, dc);

}

}

Appium Inspector(Android & iOS) or UiAutomatorViewer(Android) is used to locate element on mobile.