Three types of mibile Applications :

1.Native Apps

2. Hybrid Apps

3. Mobile Web Apps

**Native Apps** : apps which are developed in specific operating system supported language( with the same language in which mobile OS is developed).

Native apps which are are installed in mobile

Ex.Apple phones uses IOS as the Operating system ,specific language C language hence the native apps are also developed in C language only.

Advantages :

DisAdv : its bit costlier to develop and maintain.

**Hybdrid** Apps : what ever be the operating system language, Apps are not developed in each language…Apps are developed using the common components (like html5,css , etc) and use some internal plugins (like phonegap,coredava and Appcelarator Titanium) in order to wrap some specific features like camera ,gps etc…

Adv: Easy to develop,less costly and enhancements can be done easily.

Ex.gmail,uber,Instagram and twitter etc.

**Mobile Web Apps** : are the Apps which are loaded through mobile browsers (safari,chome,etc) defautly.

DisAdv: they are slow compared to otherapps and depends on the network connection.

Look and feel is not good compared to other apps

Ex.macys

Mobile testing Types : what kinds of testing we do in mobile testing :

1. Usability testing : touch actions,Orientations-horizontal/vertical-landscape/cascade
2. Compatability Testing : same OS on diff devices and diff mobiles,screen sizes,diff versions of os…
3. Interfacetesting
4. Service Testing
5. Low level resource testing : issues with memory ,local db resouces
6. Performance testing
7. Operational testing
8. Installations testing
9. Security testing

**Different Test Automation Frameworks : Popular tools**

**1.Robotium – by google**

2. UiAutometer- by google

3.Espresso

4. Calabash—supports both android and ios / little difficult

5.Selendroid—previous version of appium

6.Appium—open source framework – supports cross platform automation. we should know the versions its supported.

Official Website for Appium is “ [www.appium.io](http://www.appium.io)” – download appium for windows –inZip the folder—there will be an Installer—Run the .exe file –it creates a desktop icon –double click on it---UI gets opens for the tool – we could see the android icon(for Mac os we can test both android and ios )—

**Appium acts as an Interface between the the program and the mobile device,** where we want to actually execute the automation script. Appium internally implements the client- server architecture where the client is the Script which we want to communicate and the server is the mobile device.

Scriprt------------------Appium--------------------Device

It follows Json wire protocol : http req goes to the server and get back the response in the form of Json.

Appium is basically a server based on Json. Appium wraps internally the vendour provided libraries to performautomation.

Philosophy :

1.

2.

3.

4.

AppiumDesign :

Two ways to Setup Appium Server :

1.Using node npm packages(from command prompt)

2.Using GUI executable file

Android SDK : is the development tool kit for android developers like JDK for java developers.

**To automating first we need the App to automate.**

we need two files : either APK or IPA

Packaged bundles APK : is the packaging file for Android apps

IPA file : is the packaging file from for ios apps…

Normally we either we get these files from dev team ..or we get the source code from git/svn or anything from the dev team..Else we create our own APK file using Android Studio(IDE for android development)…to create IPA file need Xcode (IDE used for ios development)

(Appium Server—APK file ---Device ) : these we need for testing…

1.For testing we can browse for APK files google…and download

2.Once we got the APK file , now either we need a real device or an Emmulator, and we create the device using AVD Manager.

3.Appium Server

Snapshot : to skip the boot process

To use system graphical card instead of emulator graphic code

Copy the file into eclipse

1,UIAutomator : is to test Android devices: its an API to testing framework which is given by Android to test the functionalities of mobile apps.

**UIAutomator viewer**” is the tool to view or inspect element.

Emulator :emulating virtual imagination of device using AVD manager for android devices,and for ios devices using xcode and called simulators.…mimic the device…

Emulators or Simulator are given by vendours.

Emulators : are has bit more features related to hardware related.

Simulators are used by ios ..less complex application than the emulators….higher language applications used simulators ..ex.ios,blackberry

1).To write the scrpits we need the dependancies for mobile automation …in maven project

1.JavaClient dependency from maven repository(take any version)

2.testNg Dependency in order to automate the script

We can add more dependencies based on our req.

2.).we need to write program for appium using desired capabilities.

1.create the object of DeviceCapabilities Class

2.need a Device\_name

3. platform\_name

4.platfor\_version

5. APK file (is thepackaging file for android apps and IPK is the bundle file for the IOS apps)

6. Driver object

WebDriver driver = new AndroidDriver<WebElement>(new URL(“”),devicecapabilityObjectName);

Program :

**package** com.training.MobileAutomation;

**import** java.net.MalformedURLException;

**import** java.net.URL;

**import** org.openqa.selenium.WebDriver;

**import** org.openqa.selenium.remote.DesiredCapabilities;

**import** org.testng.annotations.BeforeClass;

**import** org.testng.annotations.Test;

**import** io.appium.java\_client.android.AndroidDriver;

**import** io.appium.java\_client.remote.AndroidMobileCapabilityType;

**import** io.appium.java\_client.remote.MobileCapabilityType;

**public** **class** App {

WebDriver driver;

@BeforeClass

**public** **void** beforeClass(){

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

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

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

dc.setCapability(MobileCapabilityType.***VERSION***,"5.1.1");

dc.setCapability(MobileCapabilityType.***PLATFORM\_VERSION***,"22");

dc.setCapability(MobileCapabilityType.***APP***,System.*getProperty*("user.dir")+"\\source\\yelp.apk");

dc.setCapability(AndroidMobileCapabilityType.***APP\_ACTIVITY***,"com.yelp.android.ui.activity.RootActivity");

dc.setCapability(AndroidMobileCapabilityType.***APP\_PACKAGE***,"com.yelp.android");

**try** {

driver = **new** AndroidDriver(**new** URL("http://127.0.0.1/wd/hub"),dc);

} **catch** (MalformedURLException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

@Test

**public** **void** testYelp(){

System.***out***.println("Hello :" +driver.getTitle());

}

}

Appium.io---

Adb command :dds,Android debug

Developer.android.com

Command s:

Adb logcat

Adb kill

Adb install,uninstall,devices , -d ,--------------------

Test scripts :

To inspect elements : UIAutomator viewer

Widget,layout

Normal Xpath //input

How do you correlate the elements :

AndroidStudio :to build Apk from source code.Build , sandbox,

Always do basic debugging before rising the defect..check the logs..server logs

Challenges you face during mobile testing?

Mobile automation framework : drivers will change and locator properties will differ (based on native language)

For build : maven or graddle

**How do you get API file IPK file :**

We build it from the source code.

How do you debug?

**How did you test ?** with realone or emulator: lower dev/envi(sit)/hiher envi(real device)/procution

On what desired capabilities ? to establish a session from appium

Appium can be directlt using nodejs.

**How do you locate the elemets?**

Tell about UI automator/uiAutomator viewer?

Are you familier with mobile gridsetup/sauselabs

Grid can be setup by a real machines or nowadays companies use grid based testing.(try free traila)

Q for Research :

Why mobile Testing ?

Diff ways to do Automation?