**Appium and Protractor end-to-end testing mobile application**

We will be using [Appium](http://appium.io/) with protractor to perform our E2E testing. Appium is an open source test automation framework for use with native, hybrid and mobile web apps. It drives iOS and Android apps using the WebDriver protocol.

The E2E tests it selves will be executed by [Protractor](https://angular.github.io/protractor/#/). You might already be familiar with Protractor. This is an end-to-end testing framework for AngularJS applications.

**Android sdk & emulator configuration**

1. Java SDK (minimum 1.6) must be installed and JAVA\_HOME configured.
2. Latest [Android-Sdk](http://developer.android.com/sdk/index.html) must be installed and ANDROID\_HOME set.

This makes it perfectly suitable for testing Ionic applications. The following setup will be used for testing on real devices and emulator.

**Protractor – Appium configuration**

Once we have the basic app is ready, the first step will be installing Appium and its dependencies through NPM:

npm install -g appium

npm install -g wd

We also need to install protractor:

npm install -g protractor

Now that we have installed all the dependencies, the next step is to prepare the Protractor config file. This will be similar to a normal config file, but some changes need to be made to allow us to test the webview of an android device. The full protractor config file will look like below

exports.**config** = {  
 **seleniumAddress**: **'http://localhost:4723/wd/hub'**,  
  
 **specs**: [**'./protractor/\*\_spec.js'**],  
  
 **capabilities**: {  
 **platformName**: **'android'**,  
 **platformVersion**: **'4.2'**,  
 **deviceName**: **'MC40C'**,  
 **browserName**: **""**,  
 **autoWebview**: **true**,  
 *//CHANGE THIS TO YOUR ABSOLUTE PATH* **app**: **'D:/protractor/ionic-appium-protractor-example/platforms/android/build/outputs/apk/android-debug.apk'** *// app: 'D:/dc\_MyProductivity\_debug.apk'  
 //newCommandTimeout: 60* },  
 **baseUrl**: **'http://10.0.2.2:8000'**,  
  
 *// configuring wd in onPrepare  
 // wdBridge helps to bridge wd driver with other selenium clients  
 // See https://github.com/sebv/wd-bridge/blob/master/README.md* onPrepare: **function** () {  
 **var wd** = *require*(**'wd'**),  
 protractor = *require*(**'protractor'**),  
 wdBridge = *require*(**'wd-bridge'**)(protractor, **wd**);  
 wdBridge.initFromProtractor(exports.**config**);  
  
 }  
};

The next step will be to generate the APK as seen in the protractor config file.   
This is done by the following command:

ionic build android

Now the last step will be to run Appium, this should always be started before running Protractor,   
we start it with following command:

appium

Now with everything setup we will be able to perform tests on an actual android device!

We can even create a gulp task to run the end to end testing with protractor

the below gulp task take care of running the e2e testing of a Cordova application

gulp.**task**(**'e2e'**, **function**() {  
 *//manual uninstall, the apk will else not reflect the new changes* exec(**'adb uninstall com.ionicframework.ionicappiumprotractorexample806497'**, **function**(err, stdout, stderr) {  
 **console**.log(stdout);  
 **console**.log(stderr);  
 *//generate a new android-debug.apk* exec(**'ionic build android'**, **function**(err, stdout, stderr) {  
 **console**.log(stdout);  
 **console**.log(stderr);  
 *//finally run protractor tests* exec(**'protractor protractor-config.js'**, **function**(err, stdout, stderr) {  
 **console**.log(stdout);  
 **console**.log(stderr);  
 });  
 });  
 });  
});

**Sample App cloned here**