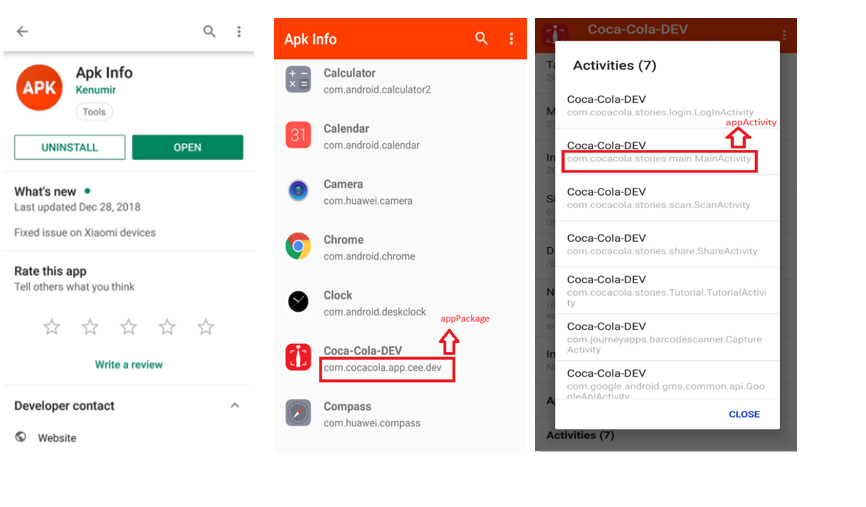
# Preparing Android/iOS devices for automation testing

          After the Automation setup environment is all configured on the machines, some settings must be done also on the devices on which we want to run the automated tests. These configurations are necessary so that the framework knows what app we want to launch, how to launch it, on what device and so on.

         Because the tests are ran on cross-platform, settings must be done both for Android and iOS devices.

**Android**

* + On the device you are trying to run the testes, unable **USB debugging** (follow the steps from the link: <https://www.embarcadero.com/starthere/xe5/mobdevsetup/android/en/enabling_usb_debugging_on_an_android_device.html>)
  + Install the APK app from Google Play which will provide the appPackage and appActivity name of any app which is installed on your mobile device.
  + **Step 1:** Download “APK Info” app from Google Play Store on your android mobile.
  + **Step 2:** Once you have successfully installed APK Info app, open it and check that it lists down all the apps that you have on your phone. Then search for “Coca-Cola”
  + **Step 3:** Check the images below

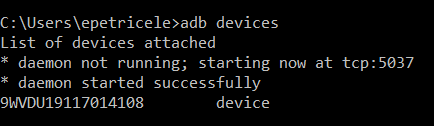


               The appPackage and appActivity will be used in the Json file like in the below example:



                 Device id must be changed with the UDID that mobile device is using:

* + Run in cmd the following command: adb devices



**iOS**

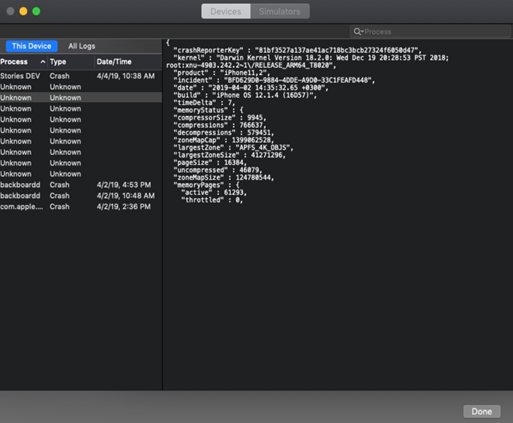
* + On the device you are trying to run the testes, enabling Developer Mode on iPhone:
    - Open the Xcode app on your Mac.
    - You will need to agree to the terms of the software and license agreements when you open Xcode  for the first time. This will install software components and finalize the Xcode installation process
    - Plug your iPhone into your Mac
    - Open the Settings app on your iPhone
    - Scroll down and tap Developer
    - Tick the Enable UI Automation label

Seeing this option in your Settings means you have enabled developer mode on your iPhone. You can now start demoing apps, checking logs and playing with other developer settings on your device.

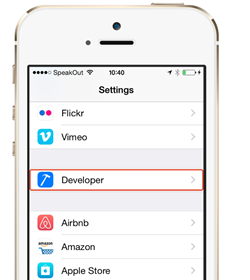
On your mac – >  open Xcode –>   from top bar select Window –> select Devices and Simulators and something like this will appear:



If you click on View Device Logs, logger will appear. Here you can see real time logging of your device and applications.



After you finished with installing Xcode, verifying that Xcode recognized your iPhone and logging works, go to your phone OS settings and verify that developer options appeared. It should look like this:



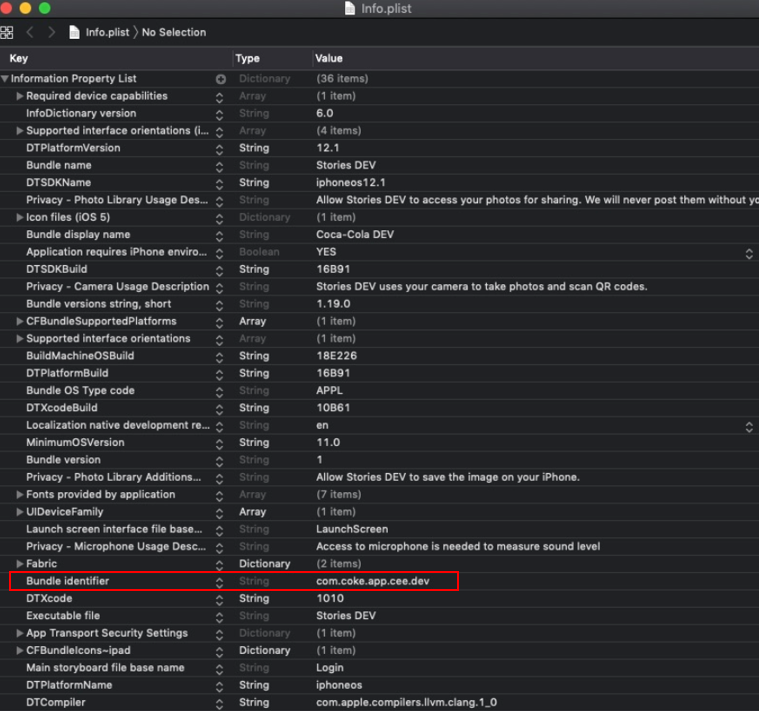
To be able to install applications on your phone, on iOS  ios-deploy commands should be used.

Basically you need ios-deploy installed and after that you only need that device to be connected via USB to your mac and just pass the commands. Command to install ios-deploy:npm install -g ios-deploy

An example of how to install an ipa file to your connected phone:

* + In terminal type Ios-deploy –b path\_to\_your\_application/application.ipa  (and now your app will be installed on your phone).

On iOS devices, the apps can be identified by the bundle identifier. In order to find the bundle id of the app, click right on the ipa file (from mac)→ Open With → Archive Utility → open the Payload folder → right click on the app → Show Package Contents → Info.plist:

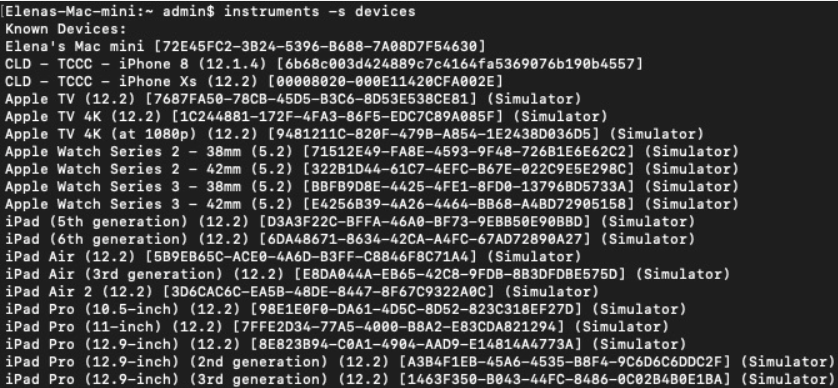


The bundle id must be used in the Json file like in the below example, so that the app can be recognized and launched on the device on which the tests are ran.



Device id can be found by writing the following command in the terminal:

* + instruments -s devices (the Simulators are also displayed in the list)



**OBS:**

In Json files it is very important to provide the location where the node.exe was installed and the appium server. The paths should be changed according to the path they were installed on your machine:

* + 0945a22383a7f563d28786a99aba868f- for Windows
  + c295931f5573c0093ac2af3e1f225d8c- for Mac

Furthermore, appiumServerPort, systemPort and wdaLocalPort should be changed for every device that you are running the testes (if several tests are run in parallel) since sometimes there can be a port conflict if different ports aren't used.

f83b7337b05a0dba667a054111e0df52

d0abf12f13ee2d9875081e7e13689526

Another important aspect here is that the usermail, userpassword, name and prefixEmail should be changed with the credentials for the email address that you are running the tests.