Katalon Studio For Automated Testing

Sesi 10

MOBILE TESTING

How To Do Android Setting on Windows

Using **Katalon Studio**, a mobile tester can design automation tests for both Android and iOS apps on physical devices, cloud services or emulators.

Unlike Web application testing, to test a mobile app, you need to install Node.js and Appium in addition to Katalon Studio.

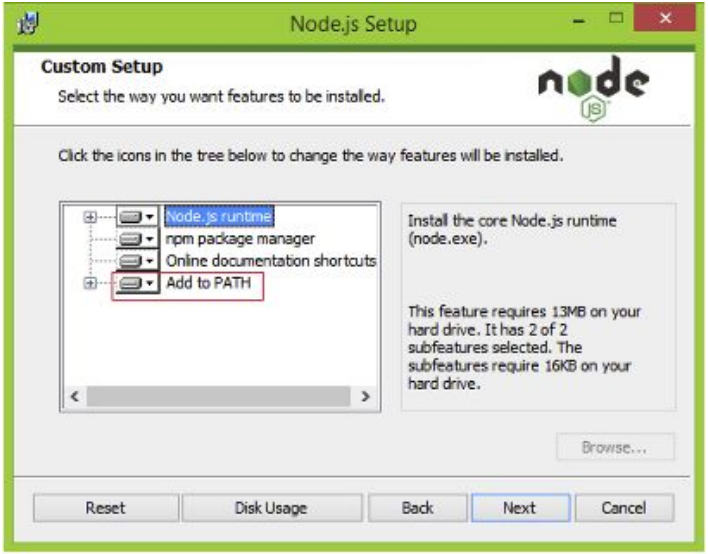
How To Do Android Setting on Windows

Install Node.js

- Download and install Node.js using the Windows installer on: https://nodejs.org/en/download

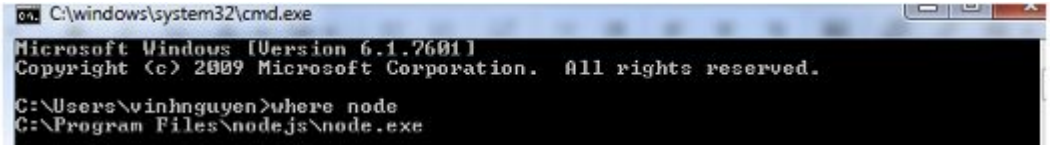
  ***Note**: Make sure you install Node.js into a location where you have full **Read** and **Write** permissions.

- When installing Node.js, make sure that the Add to PATH option is checked.



How To Do Android Setting on Windows

- Verify if Node.js is installed correctly. Open Windows command line and type the command: where nodeto see if the system can find Node.js.



Install Appium

- Open Command Prompt of Windows and input following command to install Appium:

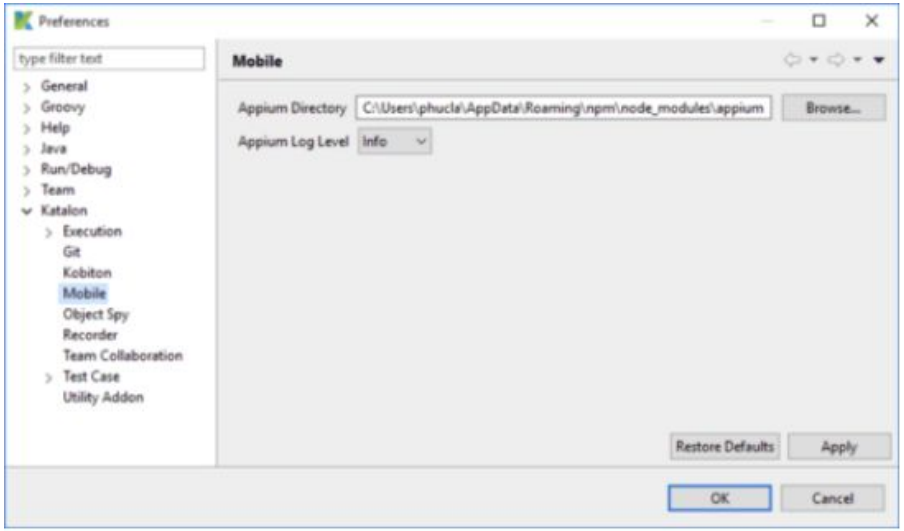
```
npm install -g appium
```

- Open Katalon Studio and go to the Windows > Katalon Studio Preferences > Katalon > Mobile settings. Set Appium Directory to the installed folder.

*By default, it's usually installed at C:\Users\{your login account}\AppData\Roaming\npm\node_modules\appium.

How To Do Android Setting on Windows

- You can also specify the level of log details to be displayed for Mobile execution using the Appium Log Level as well. There are two options: with Info, you can get the log at simple level while Debug is used in case you want more details logs.



How To Do Android Setting on Windows

Setup Devices

- Install USB Driver for your phone, you can download it from the device manufacturer site. (If your Android phone is one of Google Nexus series, please skip this step because Android SDK already includes the driver)
- Turn on the phone developer mode (go to Settings -> Developer options).
- Connect your Android device to the computer using USB cable. Just confirm if prompted for accepting/trusting the phone.
- Check whether the device is ready for automation by using ADB: Open command prompt, type command “adb devices” (without quotes). System should display a list of devices which have been connected. If this command is not recognized, then download and install the adb driver

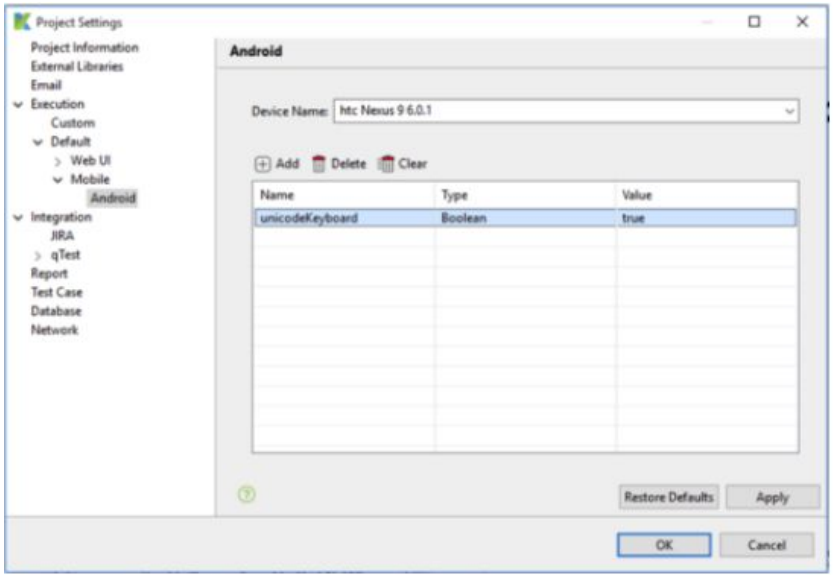
Notes: Katalon Studio also supports mobile emulators. You can start emulators and launch your virtual devices. As long as they can be recognized by using the adb command, it should be fine to execute mobile tests with them.

How To Do Android Setting on Windows

You can modify extra Desired Capabilities when executing automation test in Katalon Studio.

To define Desired Capabilities for local execution using Chrome, Firefox, IE, Safari or Edge, please access Project > Settings > Execution > Default > Mobile > Android (iOS option is available in macOS)

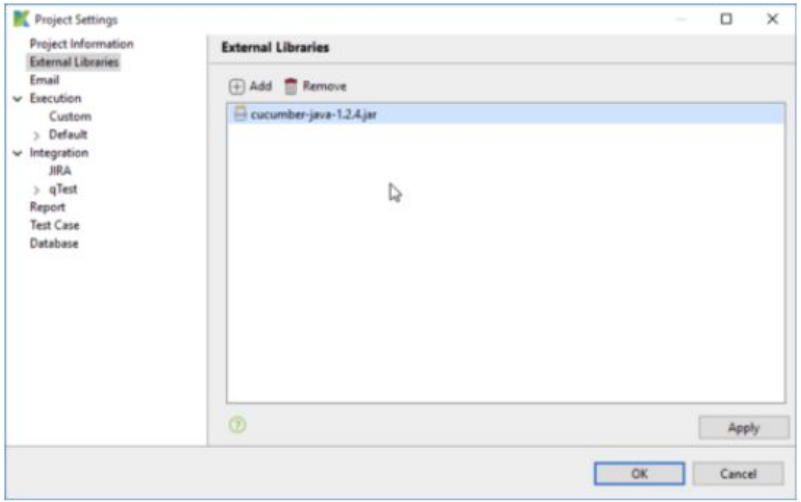
The example below show the desired capabilities settings for Android to enable Unicode input.



How To Do Android Setting on Windows

External Library

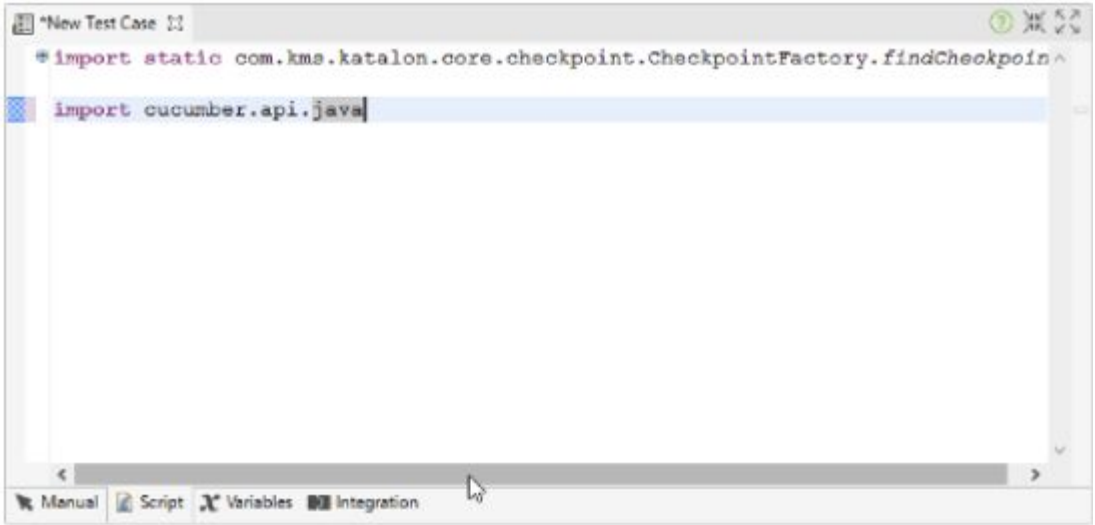
With this setting, you can add or remove your external libraries from Project > Settings > External Libraries.



- Add: Click to add your external libraries. The selected libraries will be copied to the Drivers folder in the Katalon project.
- Remove: Click to remove existing libraries. They will be removed from the Drivers folder in the Katalon project.

How To Do Android Setting on Windows

The added libraries can be imported and utilized in the Script View of test cases:



How To Do Android Setting on macOS

1. Install Homebrew
2. Install Node.js
3. Install Appium
4. Connect Appium to Katalon Studio
5. Configuration Android Device for automation
6. Run test on mobile

How To Do Android Setting on macOS

- Step 1: Install Homebrew
  https://brew.sh/
  /usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Hom...)"
  CheckI Homebrew installed
  brew -v OR brew – -version
  which brew
- Step 2: Install Node.js
  brew install node
  brew install npm (optional)
  node -v
  npm -v
  which node
  which npm
- Step 3: Install Appium
  npm install -g appium
  npm install -g appium@1.7.2
  appium -v
  which appium
  /usr/local/lib/node_modules/appium

How To Do Android Setting on macOS

- Step 4: Open Katalon Studio and connect to Appium
- Step 5: Configure android mobile for automation
  Connect mobile with your system with usb cable
- Step 6: Goto Katalon Studio and Create/Run mobile tests
- Step 7: Run and validate

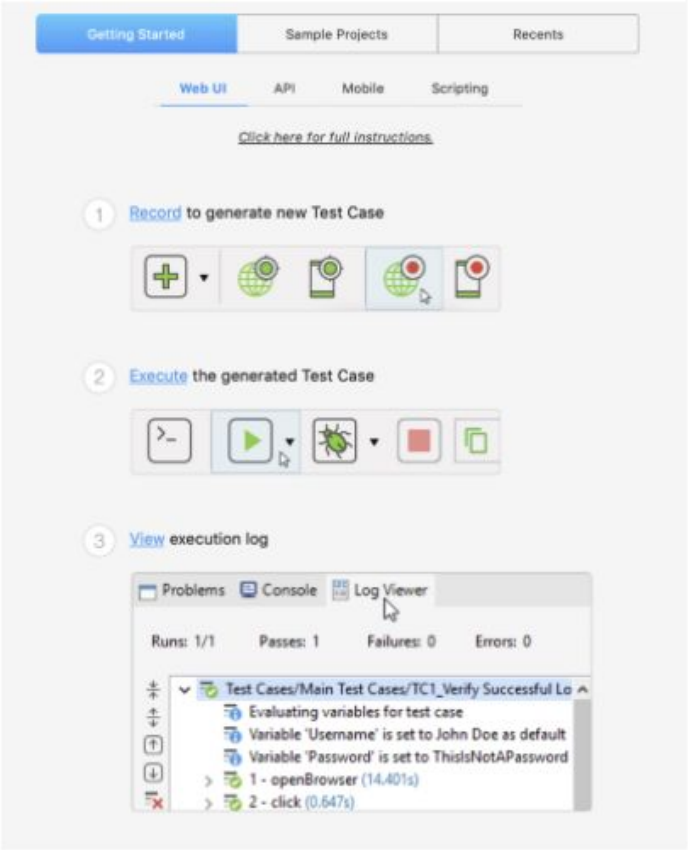
ANDROID CASE(S)

Langkah pertama adalah record untuk *generate test case.*

Ada dua pilihan untuk *generate test case* yaitu,
A. *object spy*
berfungsi untuk menyimpan object2 view componen untuk selanjutnya di save dan dibuat test case nya
B. *recorder.*
dapat langsung menyimpan object dan test case langsung sembari kita test manual.

Untuk mulai *record* pilih icon *record mobile* berwarna merah pada *toolbar.* Katalon akan mulai otomatis scan device yang ada termasuk AVD.

di sini menggunakan *AVD (Android Virtual Device)* dari android studio.
Kita akan menggunakan AVD Android 9.0 Pie.
Setelah AVD running, buka Katalon Studio.



ANDROID CASE(S)

Recording your first test with Katalon Studio Mobile Recorder, an Example

Sending a message

1. Launch the App under test (**ApiDemos.apk**)
2. Tap on “OS” text
3. Tap on “SMS Messaging” text
4. Enter Recipient, Message Body and click on **Send**

ANDROID CASE(S)

Step 1: Launch Katalon Studio, go to **File > New Sample Project > Sample Mobile Testing Project**

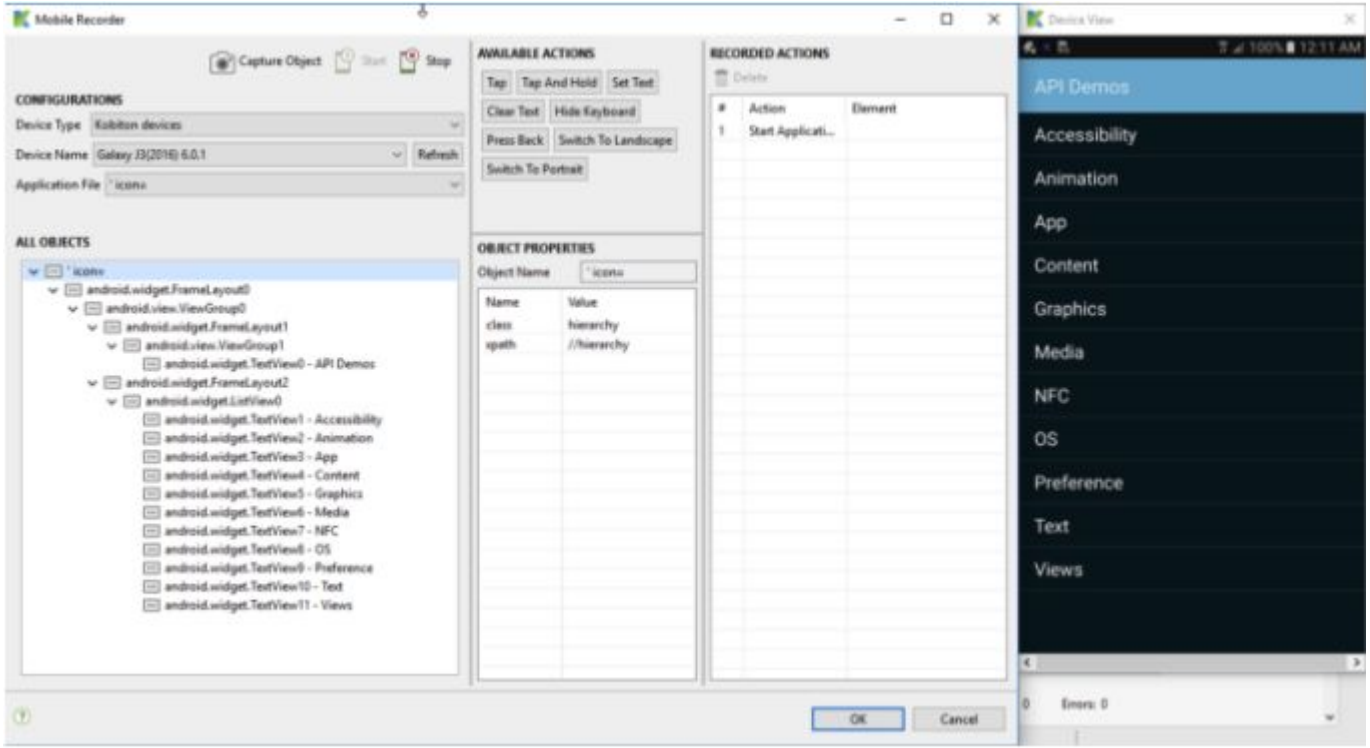
Step 2: Click on **Record Mobile** from the main toolbar.

The **Record** dialog will be displayed.

Step 3: Select device under **Device Type =>** Browse your **Application File** and proceed.

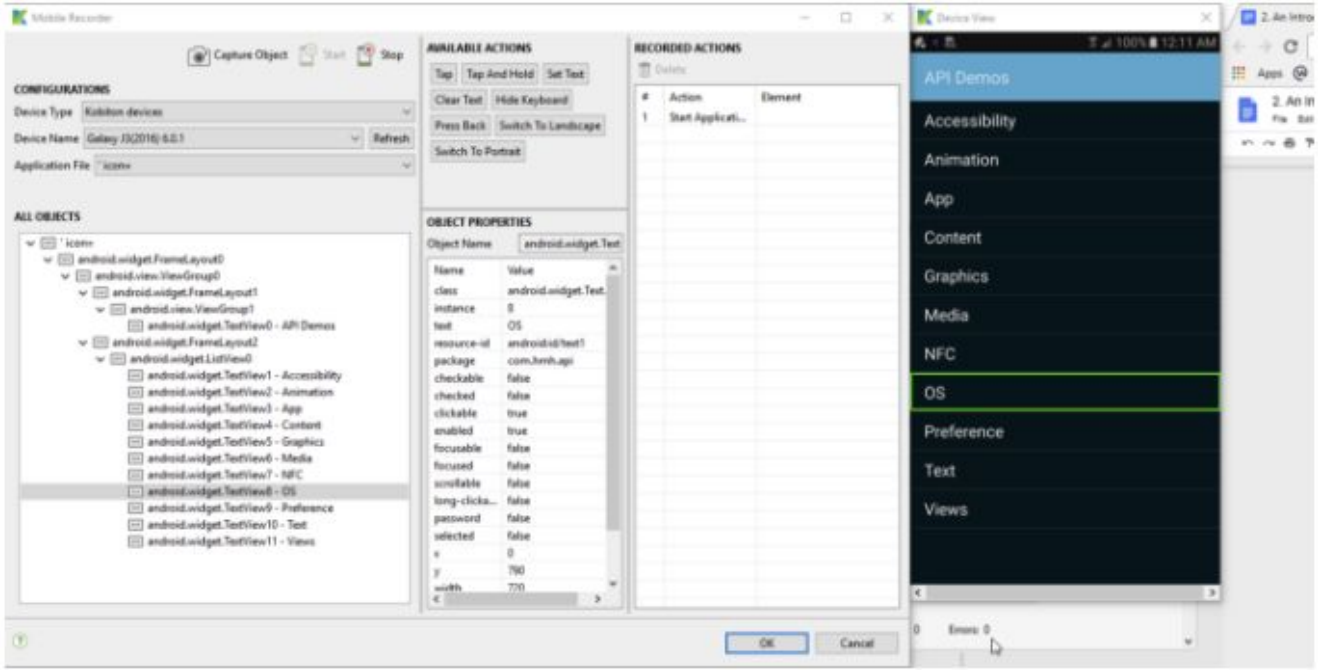
ANDROID CASE(S)

Step 4: Click on the **Start** button to begin recording your test case. Wait until your app is launched.



ANDROID CASE(S)

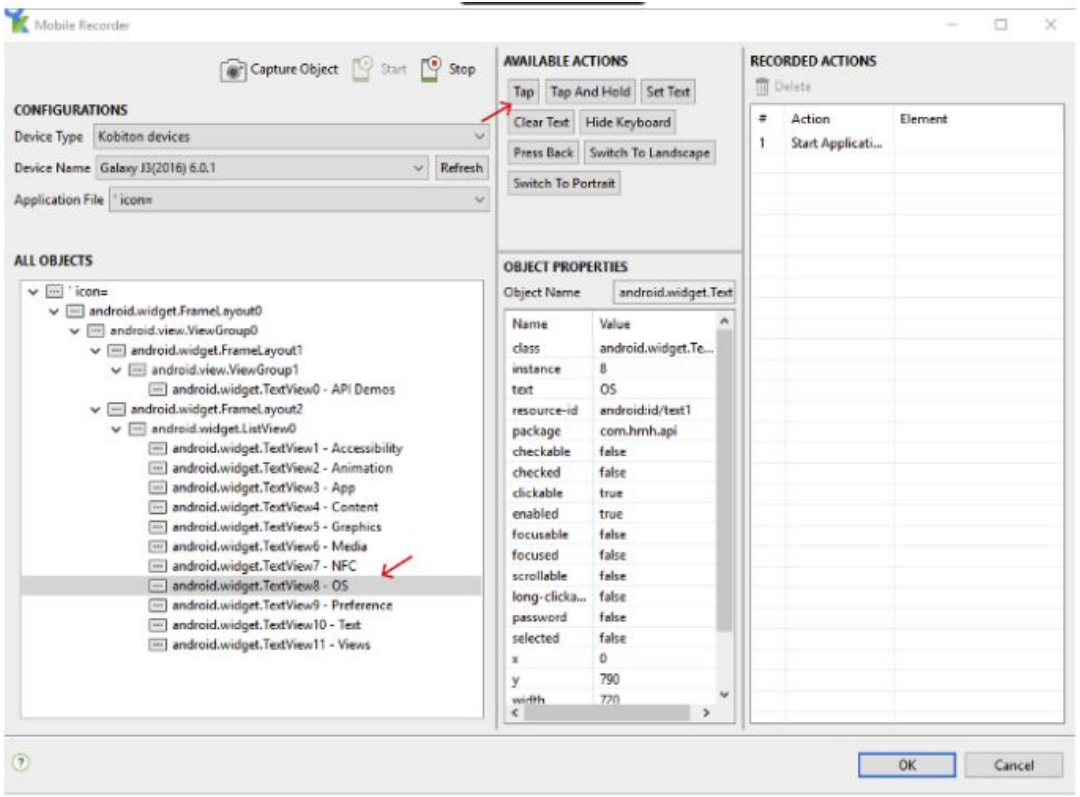
Step 5: Click on the **“OS”** button on your application. Katalon Mobile Recorder will find the object among the **All Objects** list and highlight your selection in green.



ANDROID CASE(S)

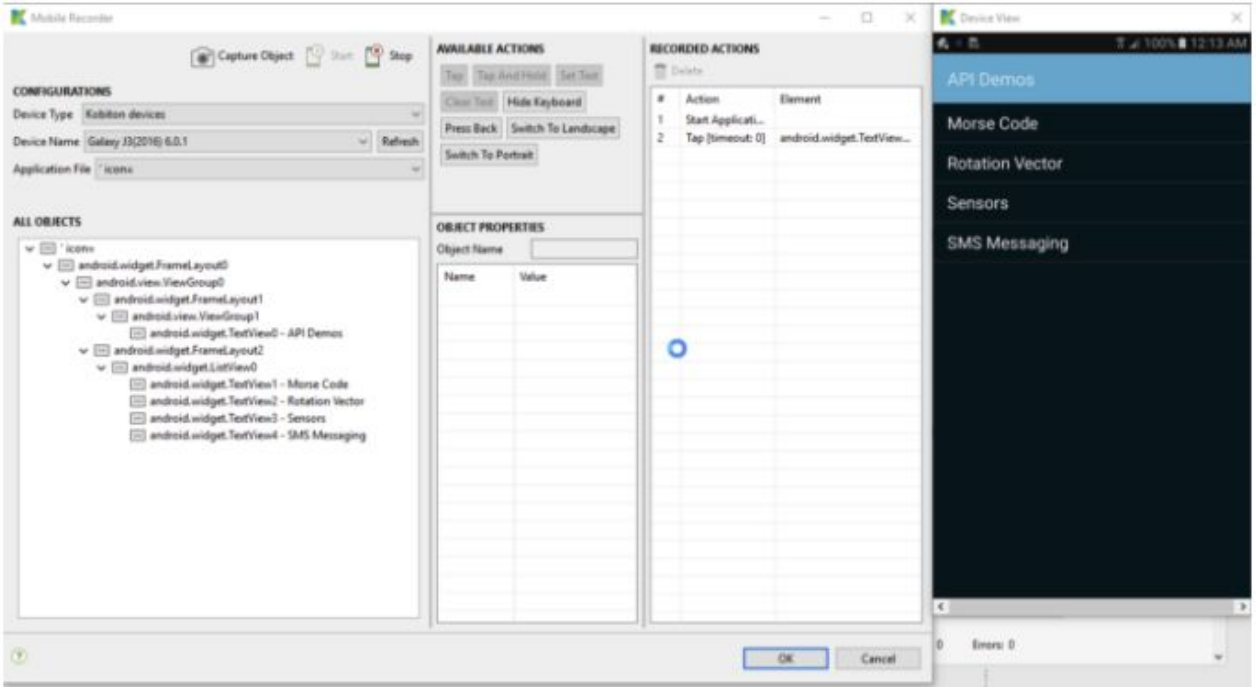
- Step 6: Once the object is selected, the actions in Record dialog box will be enabled.

  Click on **“Tap”** action.



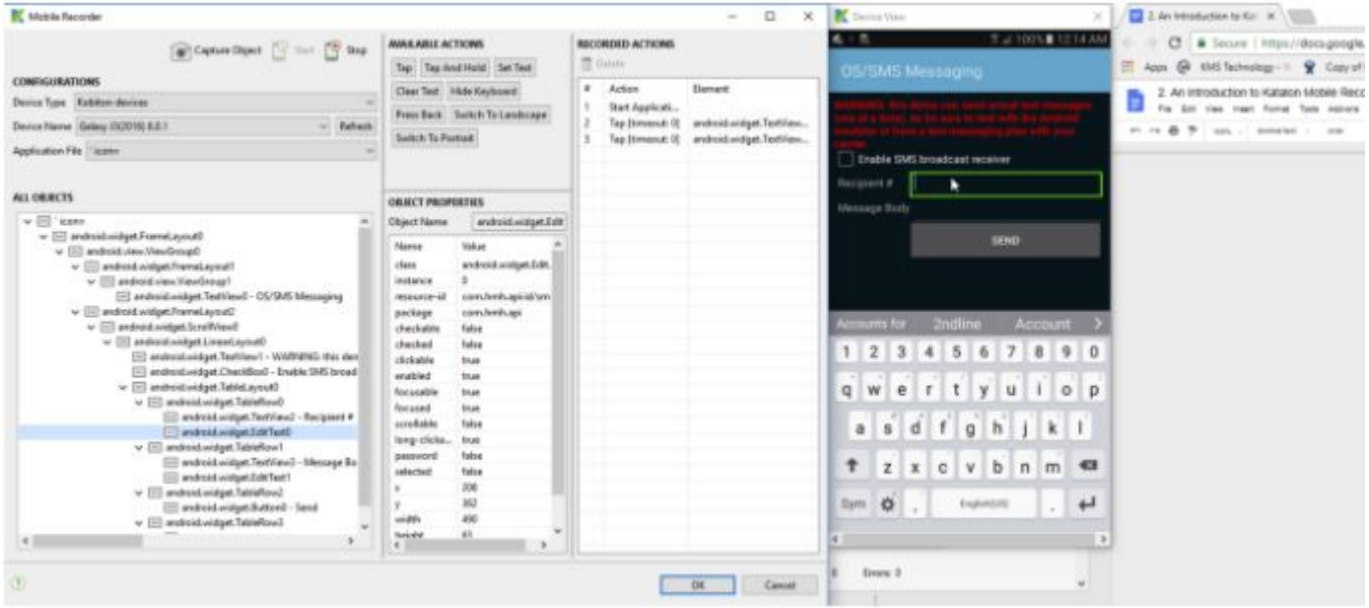
ANDROID CASE(S)

Step 7: Once **Tap** action was performed, emulator will be rendered with elements which are visible, you will see a progress information dialog. You will see the **Tap** in the list of **Recorded Actions**.

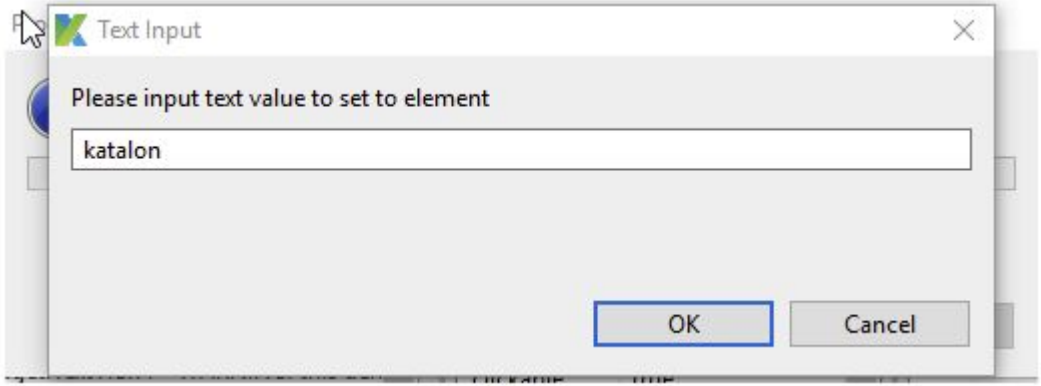


ANDROID CASE(S)

Step 8: To continue, click on “SMS Messaging” button on the new app interface. Similarly, click on “Tap” action to add it in Recorded Actions.
Step 9: Select the “Recipient” object in the actions in Record Dialog box or Mobile Screen. Click on “Set Text” action in a new popup where user needs to enter the input (“katalon”) and click on OK.

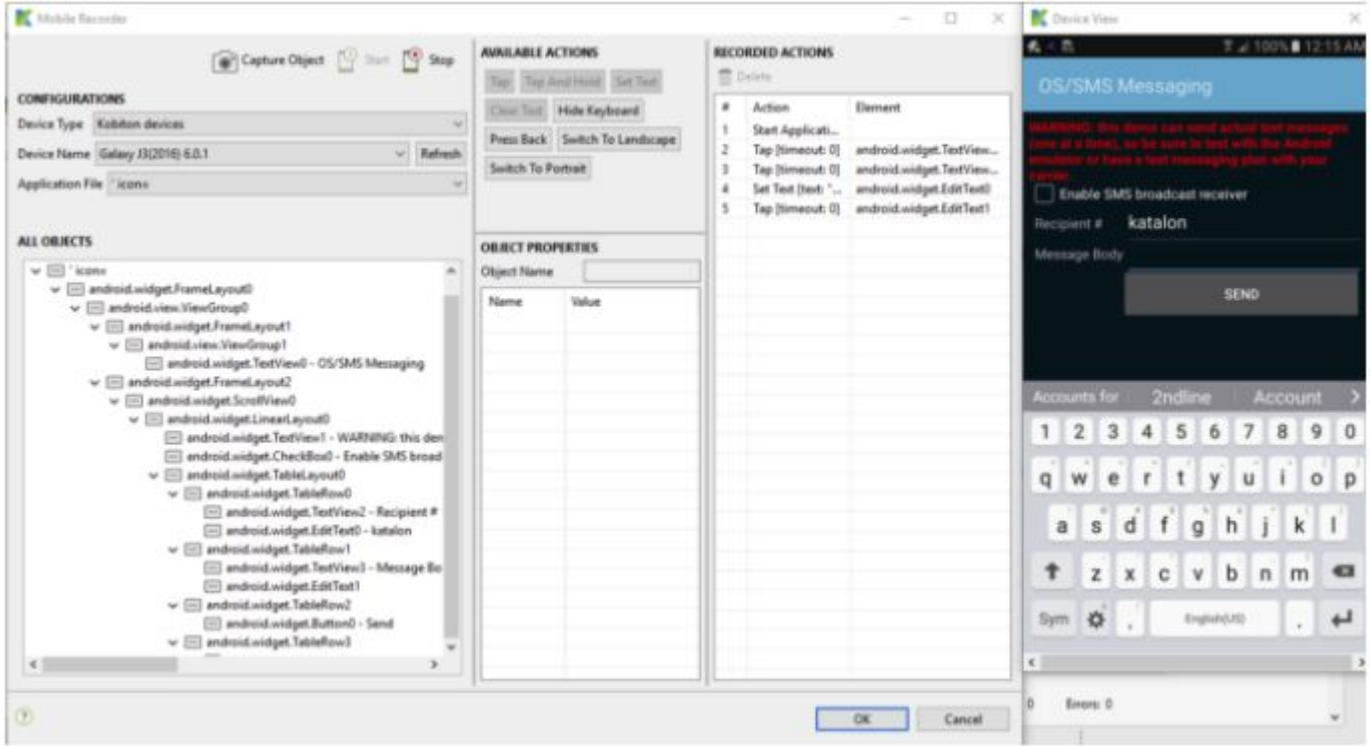


ANDROID CASE(S)



Set Text action will be added in **Recorded Actions.**

ANDROID CASE(S)

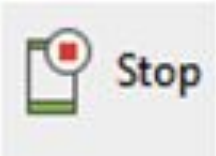


Step 10: Select the “Message Body” object in the actions in Record Dialog box or Mobile Screen. Click on “Set Text” action to input “Mobile Automation” as text. Set Text action will be added in Recorded Actions.

ANDROID CASE(S)

Warning: The Katalon Sample App will not allow you to send the message through the Send button. Thus, our example ends here.

Step 11: After finishing with your desired list of actions, you can stop the recording by clicking on **Stop** button at the top or **OK**button at the bottom.

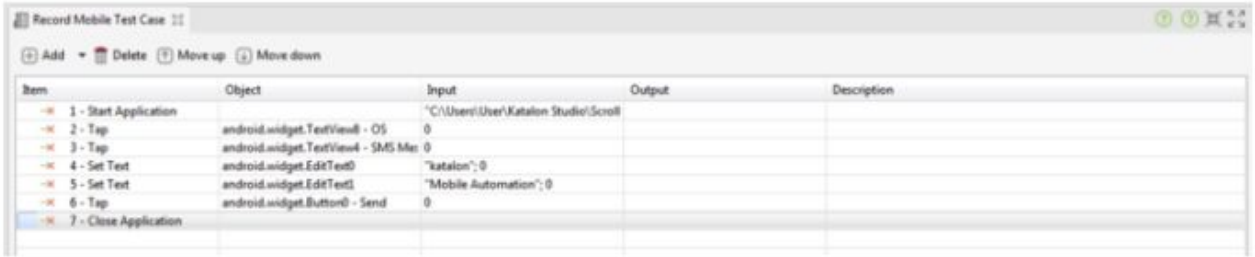


ANDROID CASE(S)

Step 12: A dialog box will be displayed for storing recorded objects into the object repository.

Step 13: If you want to add a new folder, click on **New Folder.** A popup dialog will be displayed, enter the name of the folder and proceed.

Step 14: Select your newly created test case among the **Test Explorers** drop-down. Recorded objects and actions are saved in the test case as shown below.



ANDROID CASE(S)

Step 15: You can run this test case by clicking on the **Run** button. Because the default browser is Firefox, click on the drop-down to see other selections and choose your desired device. Katalon Studio will execute the recorded steps accordingly.

ANDROID CASE(S)

For your reference, Katalon Studio generates a test script for the recorded actions,
you can view it by switching to **Script mode**.

```
import static
com.kms.katalon.core.testobject.ObjectRepository.findTestObject

import com.kms.katalon.core.mobile.keyword.MobileBuiltInKeywords as
Mobile



'Start the Application'

Mobile.startApplication('.\\Data Files\\ApiDemos.apk', true)



'Tap on text \"OS\"'

Mobile.tap(findTestObject('Record Mobile Test
Case/android.widget.TextView8 - OS'), 60)



'Tap on text \"SMS Messaging\"'

Mobile.tap(findTestObject('Record Mobile Test
Case/android.widget.TextView4 - SMS Messaging'), 60)



'Enter text as \"Katalo\" in Recipient text box'

Mobile.setText(findTestObject('Record Mobile Test
Case/android.widget.EditText0'), 'katalon', 60)



'Enter text as \"Mobile Automation\" in Message Body text box'

Mobile.setText(findTestObject('Record Mobile Test
Case/android.widget.EditText1'), 'Mobile Automation', 60)



'Tap on Send Button'

Mobile.tap(findTestObject('Record Mobile Test
Case/android.widget.Button0 - Send'), 60)



'Close the Application'

Mobile.closeApplication()
```

ANDROID CASE(S)

Testing mobile apps using Katalon Studio and Kobiton’s cloud-based device farm

Create mobile automation test cases using Katalon Studio's recording utility

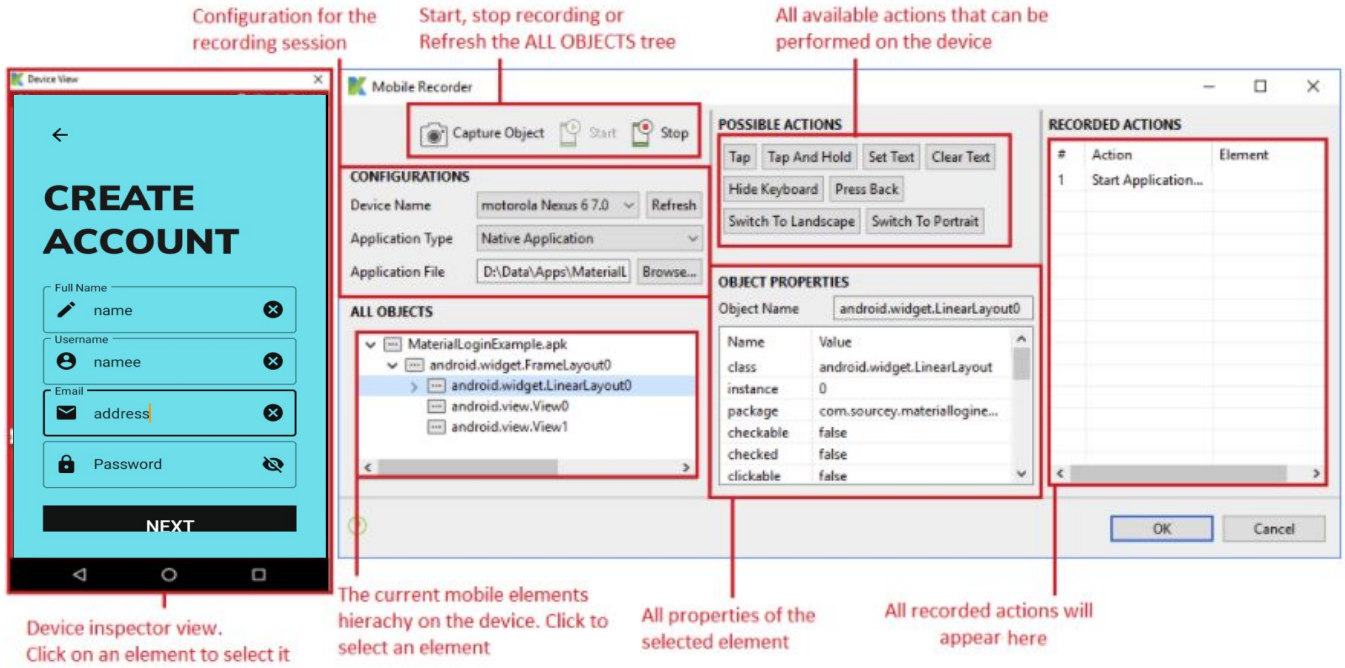
Before starting, make sure that you have:

- Downloaded the target app under test – Material Login Demo.
  This app provides a set of simple authentication features that will be automated as demonstrated below.
- Configured your Katalon Studio instance for mobile testing.

ANDROID CASE(S)

1. Let's open the sample project (already included in Katalon Studio) and create a new test case.

   Click on the Mobile Recorder button to open its dialog as shown below



ANDROID CASE(S)

Section	Explanation
Device View	Device inspector view. Preview of device UI will be displayed here. Click on any elements to select.
Configurations	Select setting for recording session
All Objects	The current mobile elements hierarchy on the device. Click to select an element and view its properties in Object Properties.
Possible Actions	All available actions that can be performed on the device
Object Properties	All properties of the selected element in All Objects.
Recorded Actions	All recorded actions appear here

ANDROID CASE(S)

2. With the dialog opened, follow the following steps to configure your recording session:

- Plug in your device or start your emulator then select it from the *Device Name* combo box.
- Click *Browse...* to select the app file from your computer (.apk for Android, .ipa for iOS apps)
- Click *Start* and wait for the app to launch on your device.

Once the configuration is completed, the app appears as shown in the screenshot above. (Note: starting the app is registered as the first recorded action.)

3. With your app ready on your mobile device, start recording the test case.

- Tap on *"No account yet? Create one"* to navigate to the *Create Account* Select the *android.widget.TextView0* object in the all objects tree. (Or you can tap the text on the Device view to
- Then click the *Tap* button to record the action. This action simulates a tap

ANDROID CASE(S)

You will be directed to the *Create Account* screen.

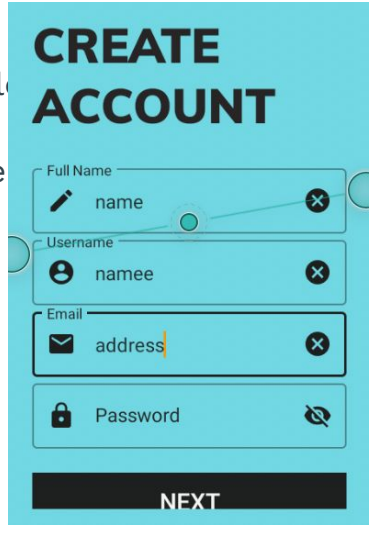
- Fill in the information to create an account.

Select android.widget.EditText0 (or tap on the Name text box on the Device view to fill in name). Then click Set Text to perform the set text action on the Name text.

Enter "name" for the text input as below. Click OK.

The text is set to the device, and the action is recorded in the Recorded actions tabl

- Perform the same steps with the *Address* field, and enter "address" for this fie



ANDROID CASE(S)

- To enter the value for the *Email* field, we will need to hide the on-screen keyboard to make the field visible by pressing the *Hide Keyboard* action button. Select the *Email* field to perform the set text action using "email@domain.com".
- Follow the same steps for the *Mobile Number*, *Password* and *Re-enter Password* text fields.
- Finally, click on *Tap* to tap on the *Create Account*. You will be redirected to the main screen.
- Click Stop on the Mobile Recorder dialog to stop the recording.

RECORDED ACTIONS

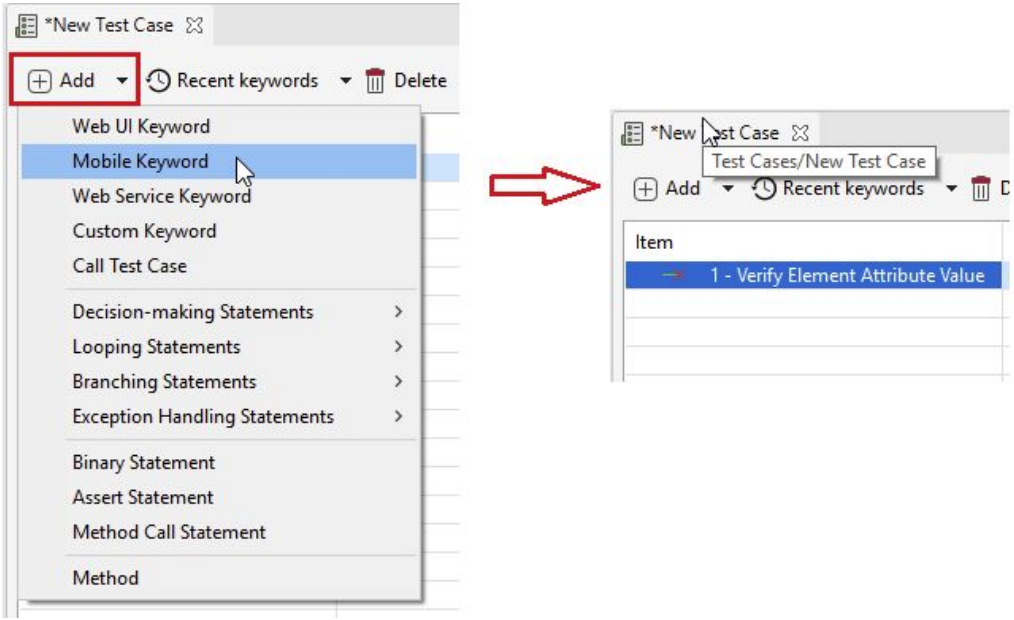
#	Action	Element
1	Start Application [appFile: "D:\Data\...	
2	Tap [timeout: 0]	android.widget.TextView0 - No account yet? Create one
3	Set Text [text: "name", timeout: 0]	android.widget.EditText0
4	Set Text [text: "address", timeout: 0]	android.widget.EditText1
5	Hide Keyboard	
6	Set Text [text: "email@domain.com",...	android.widget.EditText2
7	Hide Keyboard	
8	Set Text [text: "099949197", timeout: 0]	android.widget.EditText3
9	Hide Keyboard	
10	Set Text [text: "password", timeout: 0]	android.widget.EditText4
11	Hide Keyboard	
12	Set Text [text: "password", timeout: 0]	android.widget.EditText5
13	Hide Keyboard	
14	Tap [timeout: 0]	android.widget.Button0 - CREATE ACCOUNT
15	Close Application	

Click *OK* to complete the recording and then choose a folder to store your captured objects. Click *OK* again for Katalon Studio to generate test scripts.

ANDROID CASE(S)

4. View your test case in the Manual mode or replay it to validate all recorded steps. Let's now add validation points for this test case to verify whether the recorded inputs can set to the text fields successfully.

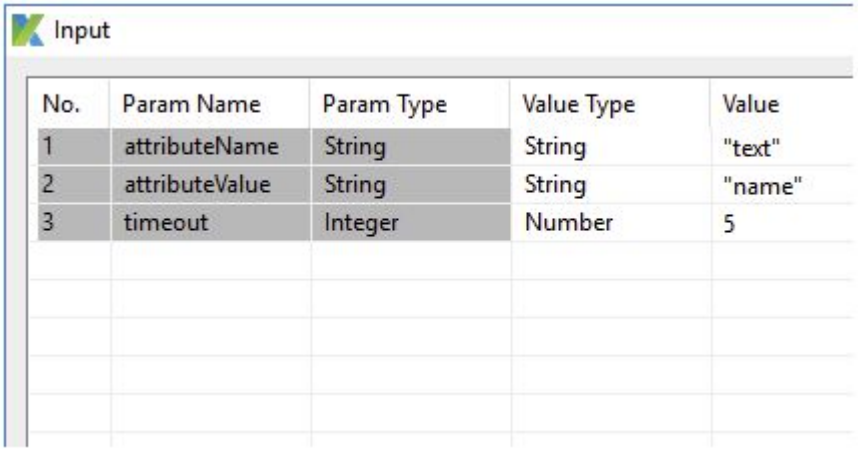
- Select the first **_Set Text _**from the *Add* toolbar or by clicking on *Add* from the context menu, add a Mobile Keyword step and choose the keyword *Verify Element Attribute Value.*



- Select the test object representing the *Name* text field, which is *widget.EditText0*, as an object to verify.

ANDROID CASE(S)

- Double click on the input cell for the test step and set the following values.



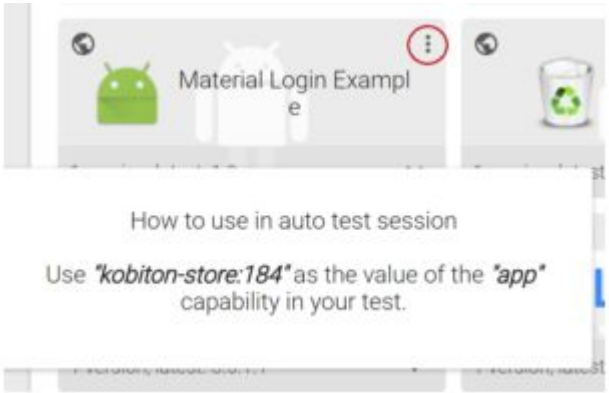
- Perform the same procedure for other *Set Text* actions.

After adding the validation test steps, try to run the test case again. This time the execution takes longer for Katalon Studio to perform the validation steps, but your test case should pass successfully since the input value is identical to the validation value. You can test with different values or connect to a data source.

ANDROID CASE(S)

Run your mobile automation test case on Kobiton's cloud devices

1. Upload the Material Login Demo app to Kobiton App Repository. From the Repository view, select the more actions button and select the *Automation snippet.* Copy the app id (the one in bold, for example, *Kobiton-store:184* as shown below) and save it.

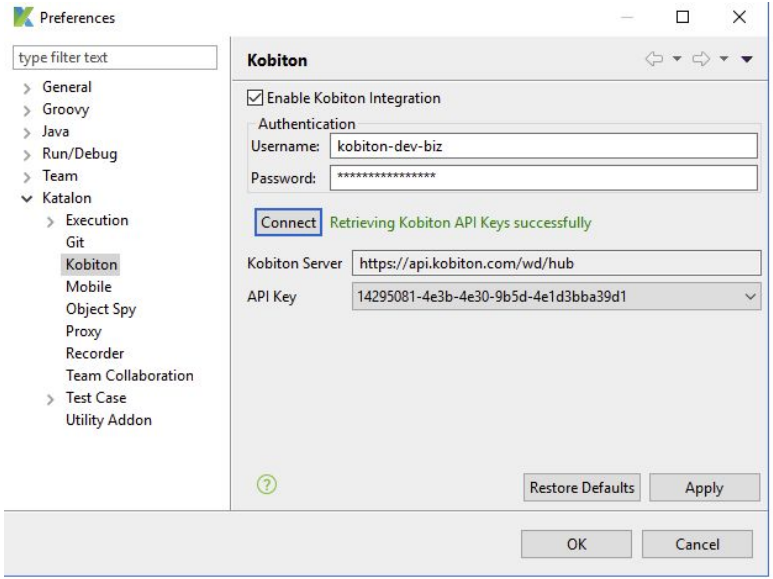


ANDROID CASE(S)

2. Navigate to the Kobiton Devices page. Choose the devices on which you want to run your test and mark those devices as a favorite by clicking on the star icon.

3. Let's enable Kobiton integration.

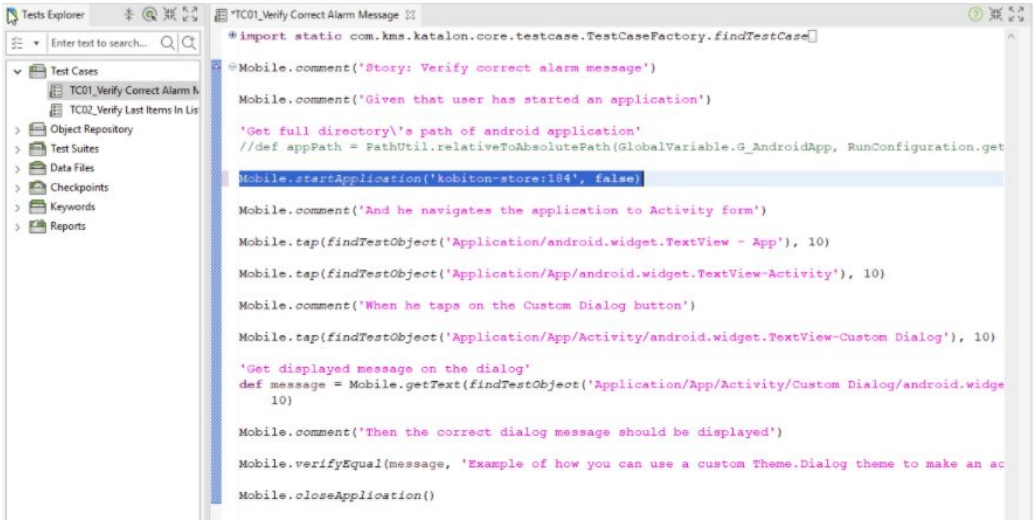
- Open Katalon Studio Preferences (*Windows -> Katalon Studio Preferences*) then select *Kobiton*.



ANDROID CASE(S)

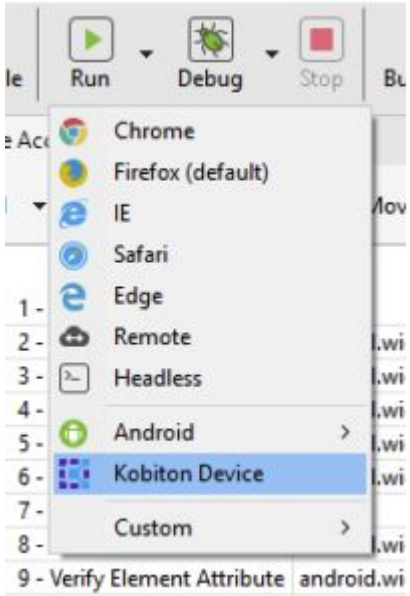
- Enter your Kobiton username and password. Click *Connect* to test the connection and authentication. With a valid account, Katalon Studio automatically retrieves the API keys from Kobiton.
- Click *Apply* or *OK* to save the settings.

4. Open the test case created from the previous section, replace the input application path for the first step "*Start Application*" with the Kobiton app id saved in Step 1.



ANDROID CASE(S)

5. A new menu option appears under the *Execution* toolbar as below. Click on Kobiton Device to open the Kobiton Favorite Device dialog.



6. Select your target device for execution from the Kobiton Favorite Device list. Click OK to start execution with that device. It may take some time to initialize the connection to Kobiton prior to the execution. You should be able to login to Kobiton sessions to view the result once the test execution completed.

How To Do Mobile iOS Setting on macOS

1. Setup Prerequisites

Katalon Studio requires the latest installation of Appium and Node.js. Please setup as the following steps:

- Install Homebrew from Terminal:

```
/usr/bin/ruby -e "$(curl -fsSL
https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

- Install Carthage with Homebrew:

```
brew install carthage
```

- Install node and npm with Homebrew:

```
brew install node
```

```
brew install npm
```

Note: To make sure Node.js is installed correctly, you run the command which node on Terminal.

How To Do Mobile iOS Setting on macOS

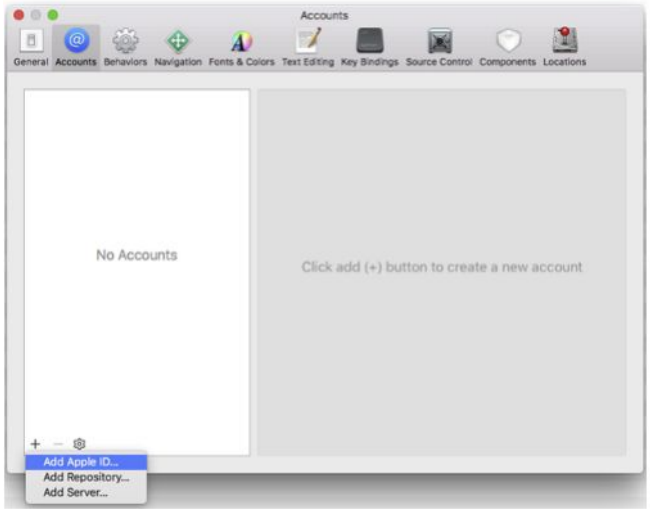
- Install Appium with npm

```
npm install -g appium
```

- Set Appium Directory to “/usr/local/lib/node_modules/appium” in Katalon Preferences (from Katalon menu, go to Preferences > Katalon > Mobile)

You will need to install and configure Xcode in case of testing on iOS devices. Please set up Xcode as follows:

- Open Xcode > Preferences > Accounts: Add developer’s Apple ID



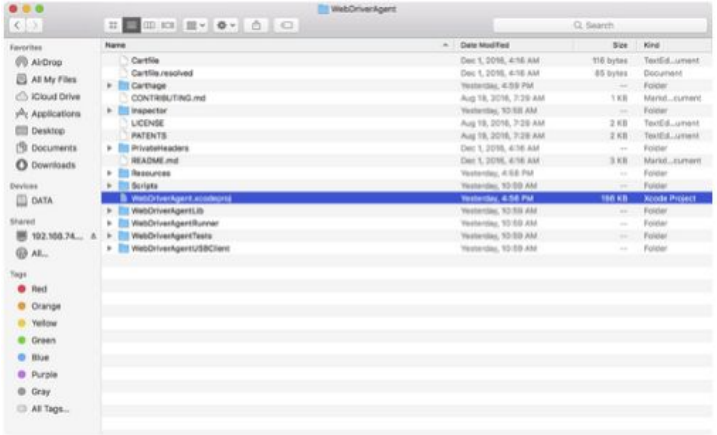
How To Do Mobile iOS Setting on macOS

- Open Terminal at WebDriverAgent in Appium directory folder: <...>/appium/node_modules/appium-xcuitest-driver/WebDriverAgent and enter following command to initialize WebDriverAgent project:

```
mkdir -p Resources/WebDriverAgent.bundle
```

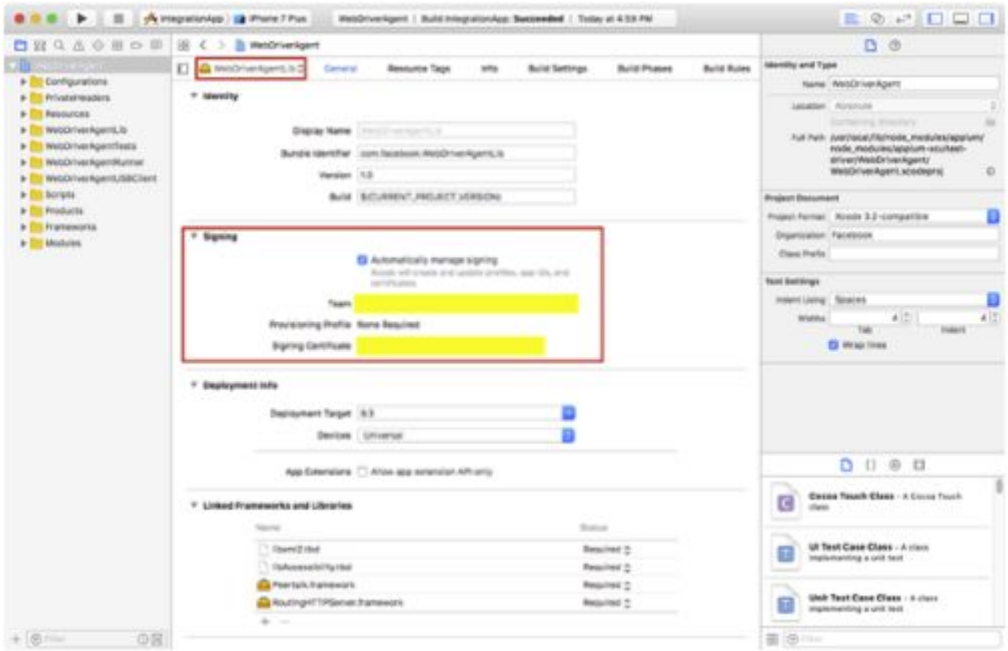
```
sh ./Scripts/bootstrap.sh -d
```

- Open project WebDriverAgent.xcodeproj under WebDriverAgent in Xcode.



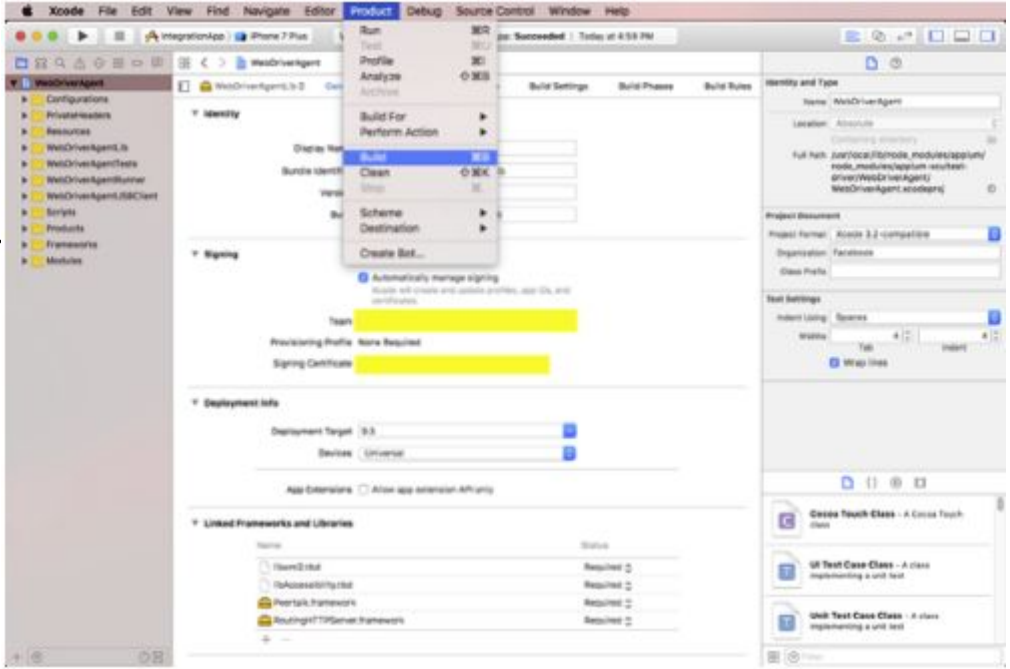
How To Do Mobile iOS Setting on macOS

- Select WebDriverAgentLib, and in the Signing section, check Automatically manage signing and select a team.



How To Do Mobile iOS Setting on macOS

- Then on Xcode's menu bar select Product > Build



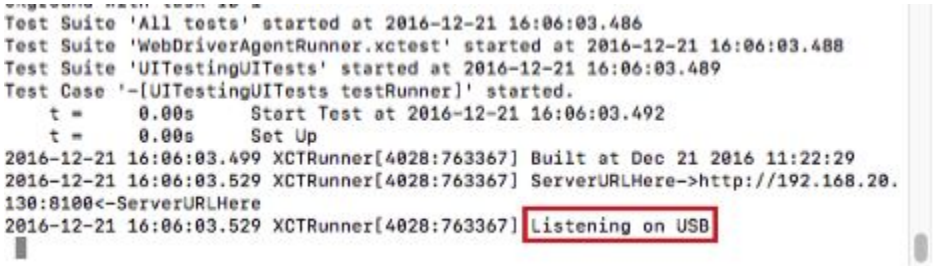
- Repeat the last two steps for WebDriverAgentRunner
- Build WebDriverAgent to verify whether the steps above work

```
xcodebuild -project WebDriverAgent.xcodeproj -scheme
WebDriverAgentRunner -destination 'id=<udid>' test
```

How To Do Mobile iOS Setting on macOS

- If the following dialog is shown, select Always Allow

- You should be able to see Listening on USB in the build output as below



How To Do Mobile iOS Setting on macOS

2. Setup Devices

For Android devices:

- Turn on your Android device's developer mode (go to **Settings > Developer option**s).
- Connect the device to your computer via a USB cable. Just confirm if prompted for accepting/trusting the device.

For iOS devices:

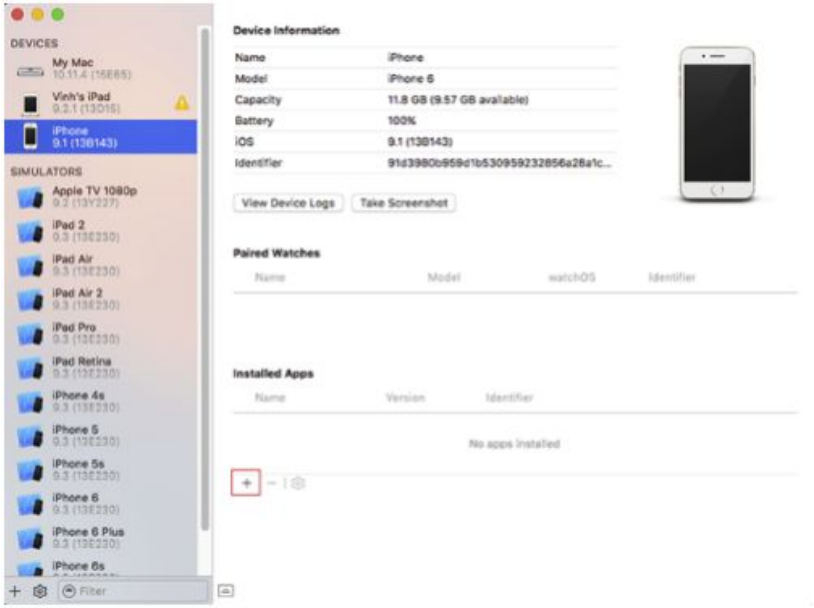
- Connect your iOS Devices to your computer via a USB cable. Just confirm if prompted for accepting/trusting the phone.
- For iOS 8 and higher, you must enable the service UI automation on the device (connect the iOS device to Xcode) as follows:
    1. plug in the iOS device
    2. open Xcode on Mac
    3. go to Settings on the device > Developer > turn ON UIAutomation

- For iOS 6 and higher, on your iOS device, go to Settings > Safari > Advanced and enable Web Inspector

How To Do Mobile iOS Setting on macOS

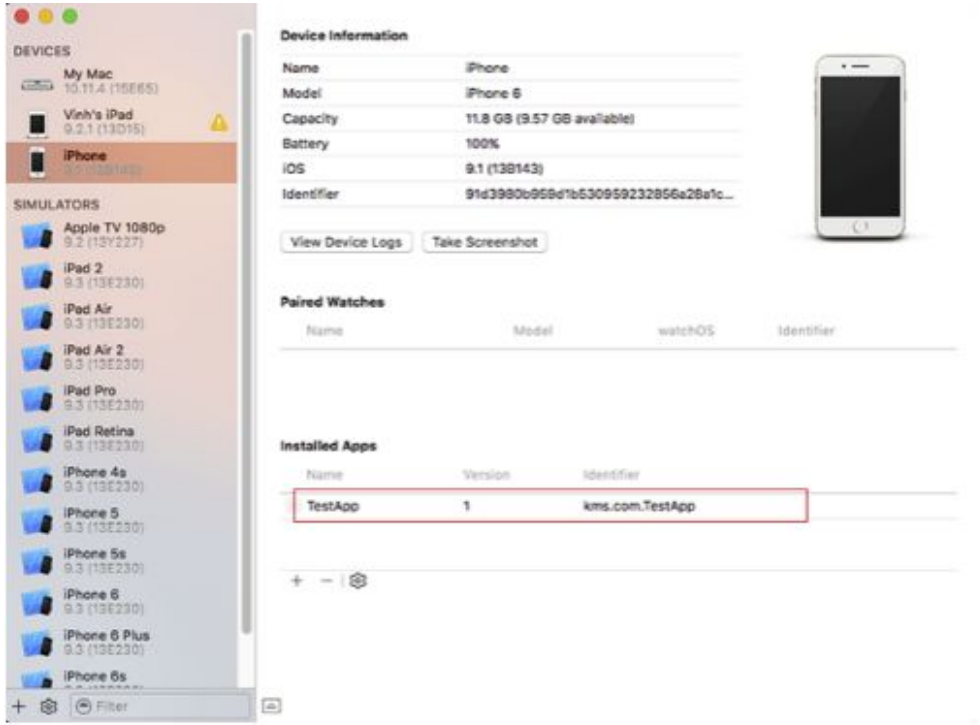
To test an iOS native application file (**.ipa** or **.app** file), make sure the file is already built and signed properly to deploy on the device. Follow these steps to check if an application file is already built and signed correctly:

1. Open **Xcode** and navigate to **Window/Devices**
2. Choose your device from the Devices list
3. Press the “+” button and choose your applicatio



How To Do Mobile iOS Setting on macOS

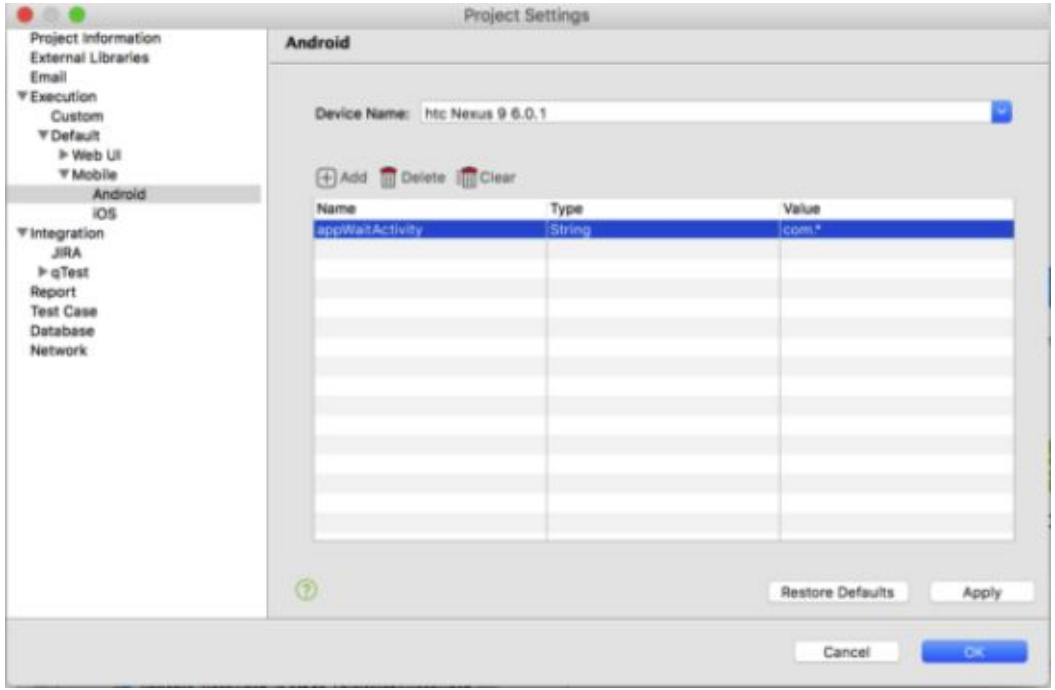
4. If installed successfully, the application will appear in the Installed Apps section as shown below



How To Do Mobile iOS Setting on macOS

To define Desired Capabilities for local execution using Chrome, Firefox, IE, Safari or Edge, please access **Project > Settings > Execution > Default > Mobile > iOS (or Android)**.

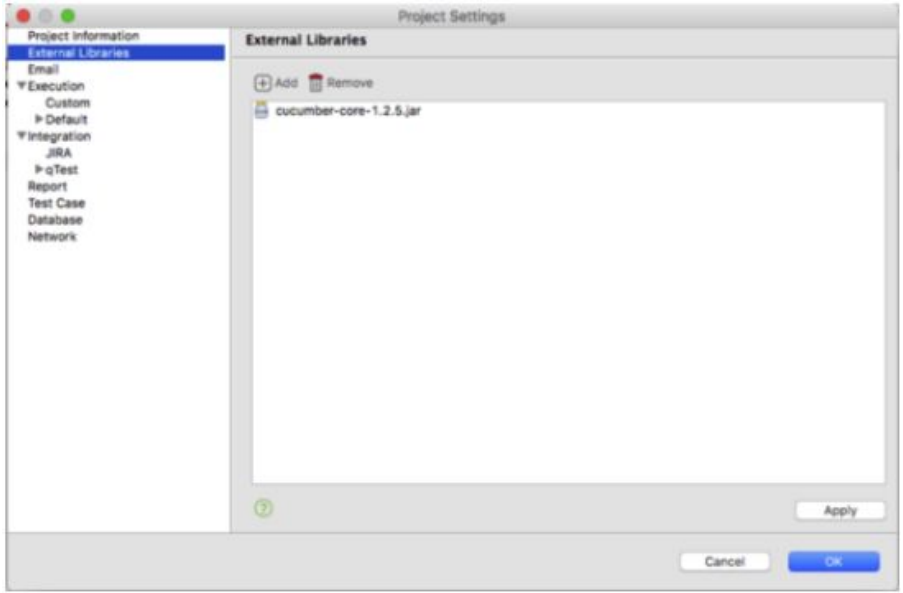
The example below shows Desired Capabilities settings for Android to enable Unicode inputs.



How To Do Mobile iOS Setting on macOS

4. External Library

You may use external libraries in your test project.
Go to **Project > Settings > External Libraries** to add new or remove existing libraries for Katalon Studio. External libraries are stored in the Katalon Studio project's **Drivers** folder.



- Add: Click to add your external libraries. The selected libraries will be copied to the Katalon project'sDrivers folder.
- Remove: Click to remove existing libraries. They will be removed from the Katalon project's Drivers folder.

How To Do Mobile iOS Setting on macOS

The added libraries can be imported and referenced in the Script View of Katalon Studio: