DETOX SETUP GUIDE

Detox is to automate user input in the app. This helps to create automated test cases.

E.g. Automate create booking, automate log in, etc.

Before doing this guide, ensure that your app is running successfully.

For the official setup guide, check [*https://github.com/wix/Detox/blob/master/docs/Introduction.GettingStarted.md*](https://github.com/wix/Detox/blob/master/docs/Introduction.GettingStarted.md)

# Contents:

* SETUP
* WRITING DETOX TEST CASES
* ADDITIONAL INFORMATION

# SETUP:

1) Install Detox CLI globally. Run npm install -g detox-cli in a terminal or command prompt

2) Run npm install in a terminal or command prompt to install all the packages in package.json

This should install Detox to the project. If not, you can manually install Detox by running npm install detox --save-dev

Similarly, npm install should also install Jest and Jest-Circus. If not, you can manually install them by running npm install jest jest-circus --save-dev

3) Once the necessary packages (above) has been installed. In your project folder, run detox init -r jest

This will generate the e2e folder, files inside the folder and a detoxrc.json file.

NOTE: If these files/folder has been generated or is inside this repo, you can skip this step. Check out the list below to verify the files/folder that is generated.

List of files/folder that will be generated:

* An e2e/ folder in your project root
* An e2e/config.json file
* An e2e/encironment.js file
* An e2e/firstTest.e2e.js file
* A .detoxrc.json file in your project root

4) Setup emulator and/or device (iOS/Android). By default, it should be set up with the correct settings (default settings).

However, check out these links to verify your setup.

Android: [*https://github.com/wix/Detox/blob/master/docs/Introduction.AndroidDevEnv.md*](https://github.com/wix/Detox/blob/master/docs/Introduction.AndroidDevEnv.md)

iOS: [*https://github.com/wix/Detox/blob/master/docs/Introduction.IosDevEnv.md*](https://github.com/wix/Detox/blob/master/docs/Introduction.IosDevEnv.md)

For Android, verify that:

* Java 1.8 is installed
* Android SDK’s path has been added to your environment variables
* Your Android emulator/device is up and running
* Quickboot is enabled (if using emulator)

*Check out the Android setup link above if any of these points are not set up*

For iOS, verify that:

* Latest version of Homebrew is installed
* Latest version of applesimutils is installed using Homebrew

*Check out the iOS setup link above if any of these points are not set up*

6) Configuring Detox to your device.

## Android:

For the official guide, check <https://github.com/wix/Detox/blob/master/docs/Introduction.Android.md>

1. Inside .detoxrc.json, configure the “android” entry to your device. Find the path of your app-debug.apk and paste it into the “binaryPath” field. Find the name of your emulator and paste it into the “avdName” field.

If you are using an emulator, “type” should be android.emulator or if you are using an actual device, “type” should be android.attached

Your configuration should look something like this:

"android": {

"type": "android.emulator",

"binaryPath": "android/app/build/outputs/apk/debug/app-debug.apk",

“build": "cd android && gradlew assembleDebug assembleAndroidTest -DtestBuildType=debug && cd ..",

"device": {

"avdName": "Pixel\_3a\_API\_28"

}

}

*NOTE: As per the instructions, edit the fields to match your machine.*

1. Go to your root build.gradle file (android/build.gradle) and add google() and detox as repository lookup points

allprojects {

    repositories {

        ...

        maven {

            // All of Detox' artifacts are provided via the npm module

            url "$rootDir/../node\_modules/detox/Detox-android"

        }

        google()

        ...

    }

}

1. Also in your root build.gragle file (android/build.gradle), add Kotlin.

buildscript {

    ext {

        ...

        minSdkVersion = 18

        kotlinVersion = '1.3.41'

    }

dependencies {

...

classpath "org.jetbrains.kotlin:kotlin-gradle-plugin:$kotlinVersion"

}

}

1. Go to your app’s build.gradle file (android/app/buid.gradle) and add in the Detox configuration

Add this to the dependencies section:

dependencies {

    ...

    // Add Detox for app actions automation

    androidTestImplementation('com.wix:detox:+')

}

Add this to the defaultConfig subsection:

android {

...

    defaultConfig {

...

        // Config for Detox (App actions automation)

        testBuildType System.getProperty('testBuildType', 'debug'

        testInstrumentationRunner 'androidx.test.runner.AndroidJUnitRunner'

    }

}

## iOS:

For the official guide, check <https://github.com/wix/Detox/blob/master/docs/Introduction.Ios.md>

1. Inside .detoxrc.json, configure the “ios” entry to your device. Find the path of your app and add it into the “binaryPath” field. Find your device type and add it to the “type” field.

Your configuration should look something like this:

    "ios": {

      "type": "ios.simulator",

      "binaryPath": "ios/build/Build/Products/Debug-iphonesimulator/LionPortal.app",

      "build": "xcodebuild -workspace ios/LionPortal.xcworkspace -scheme LionPortal -configuration Debug -sdk iphonesimulator -derivedDataPath ios/build",

      "device": {

        "type": "iPhone 11"

      }

    }

*NOTE: As per the instructions, edit the fields to match your machine.*

7) Run the app normally. npx react-native run-android or iOS equivalent.

Wait for app to completely build/run. Once app has loaded, leave it. Console will be useful for console.log messages.

8) In a different terminal, build the app with Detox. Run the detox build --configuration android or detox build --configuration ios command.

9) Run your tests with detox test --configuration android for android. For iOS, detox test --configuration ios

(If it returns an error of apk could not be found, run the detox build command (Step 4))

To specify a specific file to run, run detox test <file name> --configuration android/ios

For example, detox test login.e2e.js --configuration android/ios

# Writing Detox Tests

Write your tests in the e2e folder as Detox will only check for tests in this e2e folder.

(I have made login.e2e.js & unit-booking.e2e.js as an example)

## Explanation of examples:

Inside login.e2e.js, I have made 2 test cases: *Log in as bluefrog (PU)* and *Log in as lionadmin (A)*.

describe('Login', () => {

  it("Log in as bluefrog (PU)", async () => {

...

  })

  it("Log in as lionadmin (A)", async () => {

...

})

});

The concept of Detox is simple:

* ACTIONS: You need to point at an element as enter the action you want
* MATCHERS: You need to tell Detox what to expect in order to determine if the test is a PASS or FAIL

We will use *Log in as bluefrog (PU)* at login.e2e.js as an example.

In Line 10, the code await element(by.id('UsernameInput')).typeText('bluefrog'); means that you are telling Detox to look for the element with the id “UsernameInput” and telling Detox to type “bluefrog”.

In Line 12, the code await element(by.text('Log in')).tap(); means that you are telling Detox to look for the text “Log in” and telling Detox to tap it.

In Line 15, the code await expect(element(by.text('Home'))).toBeVisible(); means that you are telling Detox to check if the text “Home” is visible (“Home” will be visible in the bottom tab navigator when the user has successfully logged in). If it is, this will be considered as a PASS and it will continue to the next line. If not visible, this will be considered as a FAIL and the test will throw an error.

As you can see, when running this test, Detox will do these ACTIONS for you without the need of your manual input.

For help in writing tests, check:

Matchers -> <https://github.com/wix/Detox/blob/master/docs/APIRef.Matchers.md>

Actions -> <https://github.com/wix/Detox/blob/master/docs/APIRef.ActionsOnElement.md>

*These two links contain the full list and explanation of MATCHERS and ACTIONS*

With this, you are ready to use Detox!

For troubleshooting and/or if I have incorrectly set it up, check <https://github.com/wix/Detox/blob/master/docs/Introduction.GettingStarted.md>

# ADDITIONAL INFORMATION

1) For Android, Detox will remove animations in the device settings by default.

This is based on Google's recommendation to avoid flakiness. See https://developer.android.com/training/testing/espresso/setup#set-up-environment

Hence, the loading screen spinning animation will not spin. It will just be a static image.

You can enable back animations in the device settings, but every time a Detox test is run, animations will be disabled.