

## Technical Assignment - Mobile Automation

For this technical assignment, use the sample app available in the following repository:

<https://github.com/saucelabs/sample-app-mobile>

Apps Download location - <https://github.com/saucelabs/sample-app-mobile/releases/>

For iOS Simulator App --> <https://github.com/saucelabs/sample-appmobile/releases/download/2.7.1/iOS.Simulator.SauceLabs.Mobile.Sample.app.2.7.1.zip>

For Android App --> <https://github.com/saucelabs/sample-appmobile/releases/download/2.7.1/Android.SauceLabs.Mobile.Sample.app.2.7.1.apk>

Pick an app for the operating system for which you're doing your test. Your task is to test the 'Checkout' functionality:

1. Design test scenarios to exercise Add and Delete few products in the Checkout flow (don't forget negative tests!)
2. Automate test scenarios using a test framework of your choice
3. Report any bugs you've found

It's up to you how you do this, but we'll want to re-run your UI automation tests to verify the results, and for that we expect you to have chosen a suitable native framework.

### #Building the sample app

Refer to [README.md] (<https://github.com/saucelabs/sample-app-mobile/README.md>) for detailed instructions.

### #Requirements

- \* Submit your completed solution either as link to a git repository (using a public repo service such as GitHub or Bitbucket), or as a zip file (making sure you include the .git directory). Commit early and often rather than in a single large commit; trial and error is acceptable and even encouraged as it really helps us capture your thinking.
- \* Your solution should contain instructions sufficient to allow a technical user to run and replicate your results (ideally like a README.md).
- \* Produce a report highlighting any bugs in the application which you have found.

### ## Notes and suggestions:

- \* Solutions will be judged favorably if they demonstrate an ability to embed your tests in a framework suitable for a modern CI/CD pipeline. For instance, providing a DockerFile with a suitable return code that could be executed in a pipeline is a benefit.

=====