



Mobile Automation Test Report  
Signal App

Submitted by:  
**Raghav Tiwari**

Submitted to:  
Hiring Manager @CRED

## Table of Contents

<i>Content</i>	<i>Page No.</i>
Executive Summary	3
Introduction	3
Test Environment	4
Test Cases	5
Test Results	7
Conclusion	9
Appendices	10

# 1. Executive Summary

This test automation framework has been designed and implemented for the Signal Android mobile application using the Appium tool and TestNG framework. The framework provides a modular and scalable architecture that allows for easy integration of test cases, platform independence, and simultaneous execution across multiple devices using Selenium Grid.

The framework leverages object-oriented programming principles to ensure code reusability, maintainability, and readability. It follows the Page Object Model (POM) design pattern, separating test logic from page elements and actions, enhancing test maintainability and reducing code duplication.

*Key features of the framework include:*

**Modular Architecture:** The framework is structured into core components such as BaseTest, TestScripts, and utility classes, promoting code organization and reusability.

**Integration with Appium:** Utilizes the Appium tool for automating interactions with the Signal Android app, ensuring seamless testing of various functionalities.

**TestNG Integration:** TestNG is integrated for test case management, allowing for easy test execution, grouping, and reporting.

**Platform Independence:** The framework supports testing on multiple platforms, including Android, ensuring broad device compatibility.

**Parallel Execution:** Utilizes Selenium Grid for parallel execution of test cases, enabling faster feedback and reducing overall test execution time.

**Version Control:** The project is hosted on a version control system GitHub facilitating collaboration, version management, and code review.

## 2. Introduction

In the realm of software development, the efficacy of mobile applications hinges on the robustness of their testing procedures. Recognizing this imperative, we have meticulously crafted a sophisticated test automation framework tailored specifically for the Signal Android mobile application. This framework represents the culmination of advanced technology,

strategic planning, and unwavering commitment to excellence in software testing.

Our objective with this test automation framework transcends mere compliance with industry standards; rather, it seeks to establish a new paradigm of efficiency, reliability, and scalability in mobile app testing. By leveraging the formidable capabilities of industry-leading tools such as Appium and TestNG, in conjunction with proven software engineering methodologies, we have engineered a solution poised to redefine the landscape of mobile application testing.

In the ensuing sections of this document, we offer a comprehensive overview of our test automation framework, delineating its architecture, functionality, and intrinsic capabilities. We cordially invite you to embark on a journey characterized by innovation and distinction as we unveil how our framework empowers teams to realize unparalleled levels of quality assurance and expedite the deployment of impeccably crafted mobile applications.

### **3. Test Environment**

Our test environment is meticulously designed to replicate real-world conditions, ensuring comprehensive and accurate testing of the Signal Android mobile application. Leveraging state-of-the-art technology and industry-standard tools, we have curated an environment that fosters rigorous testing while maintaining efficiency and scalability.

#### *Key Components:*

**Appium Server (Version 2.5.0):** Appium serves as the cornerstone of our test environment, providing the necessary infrastructure for automating interactions with the Signal Android app across various devices and platforms. With its robust capabilities and extensive support for mobile automation, Appium enables seamless execution of test cases and precise validation of app behaviour.

**Android Studio:** As the primary Integrated Development Environment (IDE) for Android application development, Android Studio plays a pivotal role in our test environment. It facilitates the hosting and deployment of the Signal Android app on virtual or physical devices, enabling efficient testing across diverse Android environments.

TestNG Plugin for Eclipse (Version 7.9.0): TestNG, integrated seamlessly into Eclipse through the TestNG plugin, serves as the framework for organizing and executing test cases. Its versatility and flexibility empower our testing efforts, allowing for parameterization, parallel execution, and comprehensive test reporting.

Maven: Maven serves as the build automation tool for our test framework, streamlining the management of project dependencies, compilation, and execution. By leveraging Maven's standardized project structure and dependency management capabilities, we ensure consistency and reliability across our testing endeavours.

#### *Device and Platform Diversity:*

Our test environment encompasses a diverse array of devices and platforms to simulate real-world usage scenarios comprehensively. Through the utilization of emulators, simulators, and physical devices spanning various Android versions, screen sizes, and hardware configurations, we meticulously validate the compatibility and performance of the Signal Android app across a spectrum of environments.

## **4. Test Cases**

### ***Add Story Test:***

*Objective:* Verify the seamless addition of a story in the Signal Android app.

#### *Steps:*

- Click on the camera button.
- Capture a shot using the capture button.
- Share the story using the send button.
- Confirm the display of the shared story image.

*Expected Result:* The story is successfully added and displayed without errors.

### ***Delete Notification Profile Test:***

*Objective:* Ensure the accurate deletion of notification profiles.

#### *Steps:*

- Click on the three dots button.
- Navigate to the notification profile section.
- Delete the specified notification profile.

*Expected Result:* The selected notification profile is deleted effectively.

### **Delete Story Test:**

*Objective:* Validate the deletion functionality of user stories.

*Steps:*

- Access the stories section.
- Select the desired story for deletion.
- Delete the story using the provided option.

*Expected Result:* The chosen story is removed from the user's account as intended.

### **Notification Profile Test:**

*Objective:* Ensure the successful creation and management of notification profiles.

*Steps:*

- Access the notification profile settings.
- Create a new notification profile.
- Update the notification profile settings.

*Expected Result:* The new notification profile is created and updated without issues.

### **Referral Test:**

*Objective:* Validate the referral functionality for inviting friends.

*Steps:*

- Access the user's profile section.
- Navigate to the invitation options.
- Share the referral link with friends.

*Expected Result:* The referral link is shared successfully, enabling user recruitment.

### ***Update Settings Test:***

*Objective:* Ensure the seamless updating of user settings.

*Steps:*

- Access the user's account settings.
- Update appearance and theme settings.
- Modify chat colour preferences.

*Expected Result:* The user's settings are updated accurately without any glitches.

These test cases cover a comprehensive range of functionalities within the Signal Android app, ensuring its robustness and reliability.

## **5. Test Results**

Here's a detailed overview of the test results for each automated test script:

### ***Add Story Test Results***

*Test Case:* Add Story Test

*Execution Status:* Passed

*Description:* The test meticulously verified the seamless functionality of adding a story in the Signal app. By capturing a photo, sending it as a story, and validating its visibility in the story feed, the test exemplified the app's flawless performance in this critical feature.

*Observations:* Each step executed flawlessly, showcasing the app's capability to handle story creation and management without any glitches.

*Pass/Fail Criteria:* Passed

### ***Delete Notification Profile Test Results***

*Test Case:* Delete Notification Profile Test

*Execution Status:* Passed

*Description:* This test exemplified our commitment to ensuring an intuitive user experience by thoroughly testing the deletion of notification profiles in the Signal app. By navigating through the UI and successfully deleting the specified profile, the test underscored our dedication to delivering a seamless user interface.

*Observations:* The test's flawless execution highlighted the app's robustness in handling profile management tasks without any issues.

*Pass/Fail Criteria:* Passed

### **Delete Story Test Results**

*Test Case:* Delete Story Test

*Execution Status:* Passed

*Description:* Our test suite rigorously validated the functionality to delete stories in the Signal app, emphasizing our dedication to ensuring data privacy and user control. By efficiently navigating through the story feed and successfully deleting the specified story, the test reaffirmed our commitment to user-centric design principles.

*Observations:* The test's flawless execution underscored the app's reliability in managing and deleting user-generated content without any hiccups.

*Pass/Fail Criteria:* Passed

### **Notification Profile Test Results**

*Test Case:* Notification Profile Test

*Execution Status:* Passed

*Description:* This test add a notification profiles in the Signal app. By creating and configuring a new profile named "Work," the test showcased our attention to detail in ensuring customizable user experiences.

*Observations:* The flawless execution of each step demonstrated our capability to deliver tailored notification settings, enhancing user productivity and engagement.

*Pass/Fail Criteria:* Passed



## ***Referral Test Results***

*Test Case:* Referral Test

*Execution Status:* Passed

*Description:* Our comprehensive testing approach meticulously validated the referral functionality in the Signal app. By seamlessly navigating to the profile settings and successfully sending invites, the test exemplified our dedication to expanding the Signal community.

*Observations:* The test's flawless execution underscored our capability to seamlessly integrate referral mechanisms, empowering users to effortlessly invite friends and colleagues to join the Signal platform.

*Pass/Fail Criteria:* Passed

These test results underscore our unwavering dedication to quality assurance and user satisfaction, positioning the Signal app as a reliable, user-centric communication platform.

## **6. Conclusion**

The test automation framework developed for the Signal Android app demonstrates a meticulous approach towards ensuring the reliability, functionality, and performance of the application. By leveraging the latest technologies such as Appium for mobile automation and TestNG for test management, coupled with a modular architecture and object-oriented design principles, the framework offers scalability, maintainability, and ease of execution.

Through rigorous testing and validation of critical functionalities such as adding stories, managing notification profiles, and updating user settings, the framework aims to enhance the user experience and ensure the seamless operation of the Signal app across different devices and environments.

With a robust test environment, comprehensive test cases, and adherence to industry best practices, the framework serves as a vital tool for quality assurance, enabling continuous improvement and refinement of the Signal Android app to meet the evolving needs and expectations of its users.

## 7. Appendices

### *Appendix A: Test Case Details*

Test Case Name: Add Story Test

Test Scenario: Verify the functionality to add a story in the Signal app.

Preconditions: User is logged into the Signal app and has access to the camera.

Test Steps:

Click on the camera button.

Capture a photo.

Share the photo as a story.

Expected Result: The photo should be successfully added as a story, and it should be visible in the user's stories.

Test Case Name: Delete Notification Profile Test

Test Scenario: Verify the functionality to delete a notification profile in the Signal app.

Preconditions: User is logged into the Signal app and has created notification profiles.

Test Steps:

Navigate to the notification profiles settings.

Select a notification profile to delete.

Confirm the deletion of the notification profile.

Expected Result: The selected notification profile should be deleted successfully.

Test Case Name: Delete Story Test

Test Scenario: Verify the functionality to delete a story in the Signal app.

Preconditions: User is logged into the Signal app and has posted stories.

Test Steps:

Navigate to the user's stories section.

Select a story to delete.

Confirm the deletion of the story.

Expected Result: The selected story should be deleted successfully.

Test Case Name: Notification Profile Test

Test Scenario: Verify the functionality to create a new notification profile in the Signal app.

Preconditions: User is logged into the Signal app.

Test Steps:

Navigate to the notification profiles settings.

Create a new notification profile with specified settings.

Save the notification profile.

Expected Result: The new notification profile should be created and saved successfully.

Test Case Name: Referral Test

Test Scenario: Verify the functionality to invite friends via referral in the Signal app.

Preconditions: User is logged into the Signal app.

Test Steps:

Navigate to the user's profile section.

Click on the option to invite friends.

Share the referral link via available options.

Expected Result: The referral link should be successfully shared with friends.

## *Appendix D: Framework Codebase*

The source code of the test automation framework, including test scripts and related files, can be accessed via the GitHub repository at the following link:

[SignalApp Testing Framework](#)

This repository contains the Maven project structure with the necessary dependencies, configuration files, and utility classes for automating tests on the Signal app.