

Test Automation Plan

I will start by defining the objectives and scope of my automation testing for the for Mobile web application and mobile native application:

Key considerations for the automation testing for Mobile web application and mobile native application:

- ✓ Test mobile web and native app on real devices along with simulators and emulators like HTC, Samsung, Apple and Nokia.
- ✓ Test mobile web and native app on various OS versions like Android, Symbian, Windows, Blackberry and IOS
- ✓ Test mobile web and native app on different OS versions of operation system like iOS 12.x, iOS 15.x, BB5.x, BB6.x, Android- 4.2, 4.3, 4.4, etc.
- ✓ Test mobile web and native app on multiple networks like GSM and CDMA.
- ✓ Test mobile web and native app for multiple locations
- ✓ Test mobile web and native app for varying screen dimensions and other specifications

To address all the above technical aspects, the following types of testing will be performed on mobile web application and mobile native application to ensure that we are also testing on all devices relevant to our customers:

Usability testing: To make sure that the mobile app is easy to use and provides a satisfactory user experience to the customers

Compatibility testing: Testing of the application in different mobiles devices, browsers, screen sizes and OS versions according to the requirements.

Interface testing: Testing of menu options, buttons, bookmarks, history, settings, and navigation flow of the application.

Services testing: Testing the services of the application online and offline.

Low-level resource testing: Testing of memory usage, auto-deletion of temporary files, local database growing issues known as low-level resource testing.

Performance testing: Testing the performance of the application by changing the connection from 2G, 3G to WIFI, sharing the documents, battery consumption, etc.

Operational testing: Testing of backups and recovery plan if a battery goes down, or data loss while upgrading the application from a store.

Installation tests: Validation of the application by installing /uninstalling it on the devices.

Security Testing: Testing an application to validate if the information system protects data or not.

NB// We also make use of the data from the system, like user access information about the type of devices, OS and networks they use so that we can cover those users during our test. For example, we can check information like what's the highest, middle and lowest version of OS do our user devices use, then we can apply boundary value analysis in determining which OS versions to test with for OS Version and compatibility testing.

Automation Tool Selection

To choose the tools to use for the test automation of mobile web application and mobile native application, I would consider the following:

- ✓ Environment Support
- ✓ Ease of use
- ✓ Testing of Database
- ✓ Object identification
- ✓ Image Testing
- ✓ Error Recovery Testing
- ✓ Object Mapping
- ✓ Scripting Language Used
- ✓ Support for various types of test – including functional, test management, mobile, etc...
- ✓ Support for multiple testing frameworks
- ✓ Easy to debug the automation software scripts
- ✓ Ability to recognize objects in any environment
- ✓ Extensive test reports and results
- ✓ Minimize training cost of selected tools

Automation Tools Preferred

Below are the two test automation tools that I would prefer to use after considering the considerations above:

Appium:

I would prefer to use Appium which is an open-source automated testing tool for testing mobile applications. This tool also supports the automation of native, hybrid, and mobile web applications built for iOS and Android. In our case, we will be automating testing mobile web application and mobile native application which is supported by Appium. This tool uses vendor-provided automation frameworks and is based on server architecture.

Appium is especially favoured for being a flexible, cross-platform framework that I can use to create test scripts applicable to multiple platforms (Windows, iOS, and Android) – using the same API. Essentially, Appium users can reuse their source code for Android as well as iOS, thus reducing time and effort going into building tests.

Apache JMeter:

To test the performance for example, load and stress test of the mobile web app and mobile native app, a performance test tool JMeter will be used with different loads on the application.

2. Regression Testing

In order to feel confident that a new deployment doesn't introduce any bugs, I would conduct regression testing which is the selective retesting of a system or component to verify that modifications have not caused unintended effects and that the system or component still works as specified in the requirements.