Appium Automation
Test Plan

ChangeLog

Version	Change Date	By	Description
1.0	2019-04-01	Wilfredo Iglesias	Initial Proposal for review.

1 Introduction & Assumptions

The Test Plan for Appium Automation will consist of a formal basic workflow with the focus being on Android Chrome browser and Android AOL client application.
Our testing process will consist of using Appium and he language will be based on the teams preference and style..

Our methodology will consist of an agile like approach to focus on the search query and results for and user authentication and email composition for the Android AOL Client

This test plan assumes a methodology is in place, as well as the resources for the purpose of this exercise.

1.1 Scope

1.1.1 In Scope

Search query using the string mobile integrations group on google, and verify the first web search result is miwtech.com

On the AOL Client we will install the application, launch it, access the email functionality by authenticating the users id and password, create an email, send it, and verify it is received. Screenshots after each test will be captured and provided

1.1.2 Out of Scope

Areas such as performance, security testing, account/profile creation, will not be performed at this time.

1.2 Quality Objective

- Verify miwtech.com is the first web resulton Chrome browser
- Validate the the installation of AOL client on Android.

- Validate the user is authenticated and can create an email using AOL client for Android.
- Defects are identified, reported, triaged, and fixed or deferred depending on priority and severity level.
- A new build is deployed for any medium to Critical issues.
- A code freeze is implemented for one week prior to launch to perform a regression test.

1.3 Roles and Responsibilities

Detail description of the Roles and responsibilities of different team members like
- QA Analyst - Create test cases, Test plan, perform manual and automated testing against Rest API endpoint.
- Test Manager - Guide team to keep on track, escalate issues to stakeholders, develop test plan and test strategy.
- Configuration Manager - Responsible for build deployment to their respective environments, coordinates with development and QA for emergency hotfixes to be included in build.
- Developers - Development of features that comprise the entire application, fix bugs found, add those fixes to the repo to be streamed into a new build.
- Release Management - Deployment of application into production, coordinates with all team members.

2 Test Methodology

2.1 Overview

3 week sprints are going to be utilized in an Agile fashion.

2.2 Test Levels

Currently for this exercise, the testing will consist of testing search on Google using Chrome browser for Android, and Install, login, and Create and send an email using the AOL Client for Android.

2.3 Bug Triage

- Determine what the solution for each defect, and its impact to the business
- Update and assign defects based on triage.
- Daily a quick 15-30 minute bug triage will occur to highlight outstanding or ongoing issues. By the end of each sprint any outstanding issues will be resolved, or pulled into the following sprint with a high priority to get fixed first.

2.4 Suspension Criteria and Resumption Requirements

Showstoppers such as loss of data in part or complete will qualify as criteria for suspension of that functionality, missing id's that do not result in an error, or missing data that results in a failure to maintain data integrity.

When it is resolved and validated as fixed, then testing can resume.

Adding features into a planned sprint, will result in that feature to be resumed as part of a future sprint, as to avoid loss of quality due to reduced timelines and additional work.

2.5 Test Completeness

- One full cycle of Manual testing has been completed.
- Regression testing has been executed with no new issues found
- All open bugs are fixed or deferred to the next sprint for further review.

3 Test Deliverables

- Test Plan
- Test Cases
- Requirement or User Stories
- Bug Reports
- Test Strategy
- Test Metrics
- Sign Off by QA and stakeholders

4 Resource & Environment Needs

4.1 Testing Tools

- JIRA for capturing User Stories and time estimates for development and QA
- JIRA for Defect reporting and linking
- Appium/Selenium
- Spreadsheet for test case creation, JIRA or similar.

4.2 Test Environment

Following **software's** are required in addition to client-specific software.

- Nodejs
- Appium
- Java SDK
- QA environment up and running
- Windows 10 for setting up environment
- Local admin access to make system configuration changes for automation setup.
- Access to network share
- Github account access with Read/Write permissions
- GitBash or Cmder for creating pull requests, merging, cloning the repo on local env.