Automation Test Plan
Product Name: Open Weather

Prepared by: Gian Bustos
Date: 12/07/202

Table of Contents

Objective

The objective of this Automated Test Plan is to ensure the effective testing of OpenWeatherMap.org's functionalities in an automated manner. The plan outlines the scope, schedule, and execution details for automated testing.

Project Overview

OpenWeatherMap.org is a web-based platform providing users with real-time weather information and personalized services. The automated testing approach aims to validate the functionality outlined in the FRS and ensure a high level of software quality.

Assumptions

The application is deployed in the testing environment.
Test data is available for various scenarios, including valid and invalid inputs.
The test environment mirrors the production environment in terms of configurations.

Return on Investment (ROI)

The implementation of automated testing will result in increased test coverage, faster feedback cycles, and reduced manual testing efforts. ROI will be measured in terms of reduced defect count, increased release frequency, and improved overall software quality.

Automation Plan

Scope of Automation

The automation scope includes test scenarios related to the following:

Homepage features
User Registration
User login

Test Automation Tools

The following tools are selected for their compatibility with the application's technology stack:

- Carina Framework
- TestNG
- Selenium
- Appium
- Selenium Grid
- Docker

Test Data

A comprehensive set of test data, including positive and negative test cases, will be prepared to validate different scenarios. (Link will be provided soon)

Automation Design

Test Framework

The test automation framework will be the Carina Framework, chosen for its scalability, maintainability, and ease of configuration..

Test Scripts

Test scripts will be developed to cover the functional requirements outlined in the FRS. Test cases will be modular, reusable, and well-documented.

Test Environment Configuration

The test environment will be configured to simulate real-world scenarios, ensuring the reliability of the automated tests.

- Windows 10 - Chrome, Firefox
- Android Mobile OS - Chrome, Mobile App
- iPhone OS - Safari, Mobile App

Automation Execution

Test Execution Schedule

Automated tests will be executed [insert frequency], with a focus on major releases, new feature implementations, and critical bug fixes.

Task	Time Duration
Creating Tes Plan	Start Date - End Date
Test Case Creation	Start date to end date
Test Case Execution	Start date to end date
Summary Reports Submission	Date

Test Deliverables

The following are to be delivered to the client:

Deliverables	Description	Target Completion Date
Test Plan	Details on the scope of the project, test strategy, test schedule, resource requirements, test deliverables.	Date
Functional Test Cases	Test Cases created created for the scope defined	Date
Defect Reports	Detailed description of the defects identified along with screenshots and steps to reproduce on a daily basis.	na
Summary Reports	Summary Reports - Bugs by Bug# Bugs by Functional Area Bugs by Priority	

Continuous Integration

Automated tests will be integrated into the continuous integration pipeline to facilitate quick feedback and early detection of issues.

Defect Management

Defects identified during automated testing will be documented in Jira(Needs to be set up). The severity and priority of defects will be assessed and communicated promptly to the development team.

Test Automation Reporting

Test Execution Reports

Detailed test execution reports, including pass/fail status, logs, and screenshots, will be generated and shared after each test cycle.

Test Coverage Reports

Coverage reports will be generated to track the percentage of functional requirements covered by automated tests.

Automation Maintenance

Regular maintenance will be performed to update test scripts in response to changes in application features, ensuring ongoing relevance and accuracy.

Entry and Exit Criteria

The below are the entry and exit criteria for every phase of Software Testing Life Cycle:

Requirement Analysis:

Entry Criteria:

- Once the testing team receives the Requirements Documents or details about the Project.

Exit Criteria:

- List of Requirements are explored and understood by the Testing team
- Doubts are cleared

Test Planning:

Entry Criteria:

- Testable Requirements derived from the given Requirements Documents or Project Details.
- Doubts are cleared.

Exit Criteria

- Test Plan document is signed-off by the Client.

Test Designing

Entry Criteria:

- Test Plan Document is signed-off by the Client

Exit Criteria:

- Test Scenarios and Test Cases Documents are signed off by the Client.

Test Execution

Entry Criteria

- Test Scenarios and Test Cases Documents are signed-off by the Client
- The application is ready for Testing

Exit Criteria:

- Test Case Reports, Defect Reports are ready

Test Closure

Entry Criteria:

- Test Case Reports, Defect Reports are ready.

Exit Criteria:

- Test Summary Reports.

Suspension and Resumption Criteria

Based on the Client decision, we will suspend and resume the Project.
We will ramp up and ramp down the resources as per Client needs.

Risk and Mitigations

The following is the list of risks possible and the ways to mitigate them:

Risk: Non-availability of a Resource

Mitigation: Backup Resource Planning

Risk: The build URL is not working

Mitigation: Resources will work on other tasks.

Risk: Less time for Testing

Mitigations: Ramp up the resources based on the client's need dynamically.

Approvals

The team will send different types of documents for Client Approval like below:

- Test Plan
- Test Scenarios
- Test Cases
- Reports

Testing will only continue to the next steps once these approvals are done.