**OpenWeather**

**Testing Strategy**

1. **Overview**

OpenWeather is a web application. User can get weather data from this application. A strategy for the testing portion of an application describes the general approach and objectives of the test activities. It includes which stages of testing (unit, integration and system) are to be addressed and which kinds of testing are to be performed.

1. **Scope**

Features to be tested

* + Search weather
  + Links on Navigation Bar (Guide, API, Pricing, Maps, FAQ, Partners, Blog, Marketplace, Sign in, Support)
  + Authentication (Login User /Non-login user)

1. **Test approach:**

3.1. Testing Process

+ Understand Application

+ Identify the Acceptance Criteria

+ Create test tasks

+ Design and Create Test Cases

+ Develop Test Scripts

+ Execute the Test Cases (Manual/Automation)

+ Submit bugs

+ Retest the bug fixes

3.2. Test type

+ Functional testing

* + Verify the entire application works correctly.
  + Verify there is no broken links
  + Verify that the Search feature works correctly with existing value/non-existence value/special characters.

+ UI testing:

* + Font
  + Color
  + Typography
  + UI components: Input validation to decide it should be disabled / enabled /color, font size changed…
  + Tool tip
  + Mouse actions
  + Key actions
  + Responsive on mobile devices.

+ Compatibility testing:

* + Verify if it is compatible with all possible Operating Systems, Browser (Chrome, MS Edge, FF), etc.
  + Verify if it is compatible with all Mobile devices.

+ Security testing

* + Authentication: Try to access to site or modules without credentials
  + Authorization: Try to access a feature not available for the logged in user role
  + Prevention of XSS, SQL Injection,…

+ Performance testing:

* + Speed - Determines whether the application responds quickly
  + Scalability - Determines maximum user load the software application can handle.
  + Stability - Determines if the application is stable under varying loads

+ Regression Testing: Perform regression test if there is any new change in the application.

3.3. Testing techniques

* + - Manual Test
    - Automation Test

|  |  |
| --- | --- |
| What | - Component test  - Integration test  - End to end test |
| Why | - Ensure all AC are met |
| When | - After finish writing manual test cases  - Every new builds |
| How | - Tool: Cypress, Java with Selenium,… |

1. **Test Environment**
   * DEV environment
   * QA environment
   * UAT environment
   * Production environment

**+ Test Environment Setup Activities**

* Understand the required architecture, environment set-up and prepare hardware and software requirement list for the Test Environment.
* Setup test Environment and test data
* Perform smoke test on the build

1. **Deliverable Documents**

5.1. Deliverables of Test Case Development

* Test cases/scripts
* Test data

5.2. Deliverables of Test Environment Setup

* Environment ready with test data set up
* Smoke Test Results.

5.3. Deliverables of Test Execution

* Test cases updated with results
* Defect reports