Table of Contents

[1 Introduction 2](#_Toc159158328)

[1.1 Purpose 2](#_Toc159158329)

[1.2 Project Overview 2](#_Toc159158330)

[2 Scope 2](#_Toc159158331)

[2.1 In-Scope 2](#_Toc159158332)

[2.2 Out-of-Scope 2](#_Toc159158333)

[3 Testing Strategy 2](#_Toc159158334)

[3.1 Test Objectives 2](#_Toc159158335)

[3.2 Test Assumptions 3](#_Toc159158336)

[3.3 Data Approach 3](#_Toc159158337)

[3.4 Level of Testing 3](#_Toc159158338)

[3.5 Unit Testing*.* 3](#_Toc159158339)

[3.6 Functional Testing 4](#_Toc159158340)

[3.7 Non-Functional Testing 4](#_Toc159158341)

[3.8 User Acceptance Testing 5](#_Toc159158342)

[3.9 Regression Testing 6](#_Toc159158343)

[4 Execution Strategy 6](#_Toc159158344)

[4.1 Entry Criteria 6](#_Toc159158345)

[4.2 Exit criteria 6](#_Toc159158346)

[4.3 Validation and Defect Management 7](#_Toc159158347)

[5 Environment Requirements 7](#_Toc159158348)

[5.1 Test Environments 7](#_Toc159158349)

[6 Dependencies 8](#_Toc159158350)

# Introduction

## Purpose

The purpose of this test plan is to outline the testing approach, strategy, and execution plan for the React app that loads GIF photos from the Giphy API. It will cover testing for loading trending GIFs, searching GIFs, and the feedback form feature.

## Project Overview

The project involves developing a React app that fetches GIF photos from the Giphy API. The app features loading trending GIFs, searching GIFs, and a feedback form.

# Scope

## In-Scope

* Testing all functions/features related to loading trending GIFs.
* Testing all functions/features related to searching GIFs.
* Testing the feedback form feature, including form validation and local storage functionality.

## Out-of-Scope

* Integration testing with external services beyond Giphy API.
* Performance testing under heavy load conditions.

# Testing Strategy

## Test Objectives

* Validate the functionality of loading trending GIFs.
* Ensure the search functionality retrieves GIFs accurately.
* Verify proper form validation and local storage functionality in the feedback form

## Test Assumptions

* The Giphy API is stable and returns expected responses.
* The internet connection is reliable for API calls.

## Data Approach

* Use test data sets covering various scenarios for loading trending GIFs and searching GIFs.
* Test feedback form with different combinations of input data.

## Level of Testing

|  |
| --- |
| **Test Type** |
| Unit Testing |
| Functional Testing |
| User Acceptance Testing |
| Regression Testing |

## Unit Testing*.*

* Test individual components responsible for loading trending GIFs, searching GIFs, and the feedback form.
* Validate functions for fetching data from the Giphy API.

|  |  |  |
| --- | --- | --- |
| **Tester’s Name** | **Department/ Area** | **Role** |
| Mark | Development | Developer |
|  |  | Test Lead |
|  |  | Test Analyst |

## Functional Testing

**Trending GIFs**:

* Verify initial 15 GIFs are loaded correctly (image, title).
* Simulate scrolling and ensure more GIFs are loaded (up to 15 per request).
* Test pagination behavior, ensuring it stops when the last page is reached.

**Search GIFs:**

* Verify search input works and triggers API calls on submit.
* Test various search terms with expected results (image, title).
* Ensure pagination for search results works as expected.
* Handle empty search queries and invalid terms gracefully.

**Feedback Form:**

* Verify all fields are displayed correctly (name, email, star rating, feedback).
* Test required fields validation (name, email, star rating).
* Check for appropriate error messages on validation failures.

|  |  |  |
| --- | --- | --- |
| **Tester’s Name** | **Department/ Area** | **Role** |
| Junaid Amjad | QA | QA Engineer |
|  |  | Test Lead |
|  |  | Test Analyst |

## Non-Functional Testing

**Performance:**

* Measure loading times for initial GIFs, subsequent loads, and search results.
* Identify and optimize bottlenecks for a smooth user experience.

**Responsiveness:**

* Test on different devices and screen sizes (desktop, mobile, tablets).
* Ensure layout adapts and content is readable across various browsers.

**Usability:**

* Validate that the Giphy UI is easy to use and the validation messages are easy to understand

**Accessibility:**

Verify WCAG compliance for visual impairments, motor disabilities, and cognitive limitations.

**Security:**

Test for potential vulnerabilities like XSS, CSRF, and SQL injection.

|  |  |  |
| --- | --- | --- |
| **Tester’s Name** | **Department/ Area** | **Role** |
| Junaid Amjad | QA | QA Engineer |
|  |  |  |

## User Acceptance Testing

* Ensure the app meets user requirements for loading and searching GIFs.
* Validate user experience with the feedback form, including ease of use and error handling.
* Obtain feedback from stakeholders regarding overall usability.

|  |  |  |
| --- | --- | --- |
| **Tester’s Name** | **Department/ Area** | **Role** |
| Adams | Business | Business Owner/ Product Owner |
| Junaid Amjad | QA | QA Engineer |
|  |  |  |

## Regression Testing

* Ensure new changes do not introduce regressions in existing features.
* Re-run tests for loading trending GIFs, searching GIFs, and the feedback form after any modifications.

Participants:

|  |  |  |
| --- | --- | --- |
| **Tester’s Name** | **Department/ Area** | **Role** |
| Junaid Amjad | QA | QA Engineer |
|  |  | Test Lead |
|  |  | Test Analyst |

# Execution Strategy

## Entry Criteria

|  |  |  |  |
| --- | --- | --- | --- |
| **Entry Criteria** | **Test Team** | **Technical Team** | **Notes** |
| *Test environment(s) is available* | C:\Users\arxp\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\7F9Z3IW4\MC900441310[1].png |  |  |
| *Test data is available* | C:\Users\arxp\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\7F9Z3IW4\MC900441310[1].png |  |  |
| *Code has been merged successfully* |  |  |  |
| *Development has completed unit testing* |  |  |  |
| *Test scripts are completed, reviewed and approved by the Project Team* |  |  |  |

## Exit criteria

|  |  |  |  |
| --- | --- | --- | --- |
| **Exit Criteria** | **Test Team** | **Technical Team** | **Notes** |
| *100% Test Scripts executed* | C:\Users\arxp\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\7F9Z3IW4\MC900441310[1].png |  |  |
| *90% pass rate of Test Scripts* | C:\Users\arxp\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\7F9Z3IW4\MC900441310[1].png |  |  |
| *No open Critical and High severity defects* |  |  |  |
| *All remaining defects are either cancelled or documented as Change Requests for a future release* |  |  |  |
| *All expected and actual results are captured and documented with the test script* |  |  |  |
| *All test metrics collected based on reports from daily and Weekly Status reports* |  |  |  |
| *All defects logged in Defect Tracker/Spreadsheet* |  |  |  |
| *Test environment cleanup completed and a new back up of the environment* |  |  |  |

## Validation and Defect Management

* Validate test cases against expected results.
* Document defects in a defect tracker or spreadsheet.
* Track defects through resolution, retesting, and closure.

Defects found during the Testing should be categorized as below:

|  |  |
| --- | --- |
| **Severity** | **Impact** |
| *1 (Critical)* | * *Functionality is blocked and no testing can proceed* * *Application/program/feature is unusable in the current state* |
| *2 (High)* | * *Functionality is not usable and there is no workaround but testing can proceed* |
| *3 (Medium)* | * *Functionality issues but there is workaround for achieving the desired functionality* |
| *4 (Low)* | * *Unclear error message or cosmetic error which has minimum impact on product use.* |

# Environment Requirements

## Test Environments

* Stable internet connection for API calls.
* Test environment matching production environment setup.

# Dependencies

* Availability of test items such as APIs, test data, and test environments.
* Availability of testing resources including testers and technical support.

# Tools

* Jira for defect management
* Cypress for test automation
* TestRail for test management
* JMeter for performance testing