[Cat Photo Sharing Startup]

**Test Strategy**

**Revision History**

| Date | Version | Author | Description |
| --- | --- | --- | --- |
| 16.10.2023 | 1.0 | AB | Initial Draft |
| 18.10.2023 | 1.1 | AB | Revised and final Draft |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

[**1. Scope 3**](#_gjdgxs)

[**2. Test Approach 4**](#_30j0zll)

[**3. Test Environment 5**](#_1fob9te)

[**4. Testing Tools 6**](#_3znysh7)

[**5. Release Control 7**](#_2et92p0)

[**6. Risk Analysis 8**](#_tyjcwt)

[**7. Review and Approvals 9**](#_3dy6vkm)

# Scope

It defines parameters like

* Who will review the document?
* Who will approve this document?
* Testing activities carried out with timelines

**Document Review and Approval**

This document will be reviewed by the QA Lead and approved by the Project Manager.

**Testing Activities**

1. Functional testing of the Cat Photo Sharing application.
2. Performance testing for handling concurrent users and photo uploads.
3. Security testing for identifying vulnerabilities.
4. Compatibility testing on various devices and browsers.
5. User acceptance testing (UAT) with real users.
6. Usability testing for the user interface and overall user experience.

**Timelines**

The testing activities will be conducted in parallel with the development phases, and test cycles will align with project milestones. A detailed test schedule will be provided in a separate test plan document.

# Test Approach

It defines

* Process of testing
* Testing levels
* Roles and responsibilities of each team member
* Types of Testing ( Load testing, Security testing, Performance testing etc.)
* Testing approach & automation tool if applicable
* Adding new defects, re-testing, Defect triage, Regression Testing and test sign off

**Process of Testing**

* A systematic approach to testing based on the software development life cycle (SDLC) is followed.
* Conduct requirements review and analysis are performed.
* Develop and execute test cases.
* Track and report defects.
* Regression testing after each code change is performed.

**Testing Levels**

* Unit Testing: Performed by developers to test individual components and functions.
* Integration Testing: Verifying interactions between different modules.
* System Testing: Ensuring the entire system works as a whole.
* User Acceptance Testing (UAT): Conducted by real users to validate that the application meets their expectations.

**Roles and Responsibilities**

* QA Lead: Test planning, test case development, test execution, and defect tracking.
* Developers: Unit testing and resolving defects.
* Project Manager: Overall project management and ensuring testing aligns with project timelines.
* Business Analysts: Requirements validation.
* Real Users: UAT and usability testing.

**Types of Testing**

* Functional Testing: Ensure all features work as expected.
* Performance Testing: Assess the application's performance under load.
* Security Testing: Identify vulnerabilities and protect against data breaches.
* Compatibility Testing: Verify the application works on various devices and browsers.
* Usability Testing: Evaluate the user interface and overall user experience.

**Release Control**

* Manage releases with version control.
* Ensure all test execution aligns with the appropriate release.

# Test Environment

* Define number of requirement and setup required for each environment
* Define backup of test data and restore strategy

**Number of Environments**

* Development Environment
* Testing Environment
* Staging Environment
* Production Environment

**Test Data**

* Use real cat photos and synthetic data for testing.
* Ensure data privacy and compliance with relevant data protection regulations.

# Testing Tools

* Automation and Test management tools needed for test execution
* Figure out number of open-source as well as commercial tools required, and determine how many users are supported on it and plan accordingly

**Automation Tools:**

* Selenium for web automation.
* Appium for mobile automation.

**Test Management Tools:**

* JIRA for issue tracking.
* TestRail for test case management.

# Release Control

* Release management plan with appropriate version history that will make sure test execution for all modification in that release

**Release Management Plan:**

* Weekly sprints and releases.
* Every release will be assigned a version number.

**Version History:**

* Detailed release notes with changes, improvements, and new features.

# Risk Analysis

* List all risks that you can estimate
* Give a clear plan to mitigate the risks also a contingency plan

**Identified Risks**

* Inadequate performance under high user loads.
* Security vulnerabilities leading to data breaches.
* Compatibility issues on specific devices or browsers.
* Lack of user engagement and adoption.
* Data privacy violations.

**Risk Mitigation**

* Performance Testing: Conduct load testing to identify performance bottlenecks and optimize.
* Security Testing: Regularly test for vulnerabilities and apply security patches.
* Compatibility Testing: Test on a wide range of devices and browsers.
* User Engagement: Continuously gather user feedback and iterate based on it.
* Data Privacy: Ensure compliance with data privacy regulations and educate users about data usage.

# Review and Approvals

* All these activities are reviewed and sign off by the business team, project management, development team, etc.
* Summary of review changes should be traced at the beginning of the document along with approved date, name, and comment

This test strategy document will be reviewed and approved by the following stakeholders:

* QA Lead
* Project Manager
* Development Team
* Business Analysts