

Software Testing Plan

21/5/2022

V 1.0

Prepared by: Mai Ashraf

# Revision History

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Owner Name** | **Date** | **Status** | **Approved by** | **Version** |
| Mai Ashraf | 21/5/2022 | Stable |  | 1.0 |
|  |  |  |  |  |
|  |  |  |  |  |

# Contents

[Revision History 1](#_Toc104556614)

[Contents 2](#_Toc104556615)

[1. Introduction 4](#_Toc104556616)

[1.1. Purpose of the document 4](#_Toc104556617)

[1.2. Project Overview 4](#_Toc104556618)

[1.3. Audience 4](#_Toc104556619)

[1.4. Roles and Responsibilities 4](#_Toc104556620)

[1.5. Quality Objectives 4](#_Toc104556621)

[2. Test Strategy 5](#_Toc104556622)

[2.1. Overview 5](#_Toc104556623)

[2.2. Test Objectives 5](#_Toc104556624)

[2.3. Test Principles 5](#_Toc104556625)

[2.4. Scope of Testing 5](#_Toc104556626)

[2.4.1. In Scope 5](#_Toc104556627)

[2.4.2. Out Scope 6](#_Toc104556628)

[2.5. Test Levels 6](#_Toc104556629)

[2.5.1. System Test 6](#_Toc104556630)

[2.5.2. User Acceptance Test 6](#_Toc104556631)

[2.6. Test Techniques 6](#_Toc104556632)

[2.6.1. Black-Box Technique 6](#_Toc104556633)

[2.6.2. Experience-Based Technique “Exploratory Test” 7](#_Toc104556634)

[2.7. Test Types 7](#_Toc104556635)

[2.7.1. Functional Test 7](#_Toc104556636)

[2.7.2. Change-related Test. 7](#_Toc104556637)

[2.8. Test Acceptance Criteria 8](#_Toc104556638)

[2.9. Test Deliverables 8](#_Toc104556639)

[3. Execution strategy 8](#_Toc104556640)

[3.1. Entry and Exit Criteria 8](#_Toc104556641)

[3.2. Validation and Defect Management 8](#_Toc104556642)

[3.3. Defect Tracking & Reporting 9](#_Toc104556643)

[4. Test Management process 10](#_Toc104556644)

[4.1. Test Management tool 10](#_Toc104556645)

[4.2. Test Design Process 10](#_Toc104556646)

[4.3. Test Execution Process 10](#_Toc104556647)

[5. Test Environment and Tools 11](#_Toc104556648)

[5.1. Test Tools 11](#_Toc104556649)

[5.2. Test Environment 11](#_Toc104556650)

[6. Terms 11](#_Toc104556651)

# Introduction

## Purpose of the document

The test plan describes the test approach and overall framework that will be used in testing. This document introduces:

* Test Strategy: rules the test will be based on, including the givens project (start /end, dates, assumptions) description of the process to set up a valid test.
* Execution Strategy: describes how the test will be performed and process to identify and report defects, and to fix and implement fixes.
* Test Management: process to handle the logistics of the test and all the events that come up during execution.

## Project Overview

The Learning Hub functions as an online free encyclopedia with a variety of material, allowing users to discover and receive knowledge in a variety of life domains while also participating in a community where they may contribute and share their expertise.

## Audience

* Project team members perform tasks specified in this document, provide input and recommendations on this document.
* Project Manager Plans for the testing activities in the overall project schedule, reviews the document, tracks the performance of the test according to the task herein specified, approves the document and is accountable for the results.
* The stakeholders’ representatives and participants may take part in the user acceptance test to ensure the business is aligned with the results of the test.

## Roles and Responsibilities

Detail description of the Roles and responsibilities of different team members like

* Test Manager: Maha Mohamed.
* Configuration Manager: Rana Mostafa.
* Developers: Radwa Kamal, Fatma Mekhemer.
* Testers: Mai Ashraf, Mostafa Gamal.

## Quality Objectives

* Ensure the Application Under Test conforms to functional and non-functional requirements
* Ensure the application under test meets the quality specifications defined by the client
* Bugs/issues are identified and fixed before go live
* All test cases designed are compared with the SRS and approved
* Send out our website with defects fewer than 10% of products with a defect
* Maintain the customer statisfaction

# Test Strategy

## Overview

Our strategy for the project is using waterfall model, after finishing development it’s planned to start testing phase to certify that the Application Under Test meets client expectations.

This software Engineering model is very simple to plan and manage. Where requirements are clearly defined and stated beforehand, can be easily tested using a waterfall model.

## Test Objectives

The objective of the testing is to ensure that Learning Hub Web Application is functioning as requested by the user and captured in the SRS document.

The test will execute and verify the test scripts, identify, fix, and retest all high and medium severity defects per the entrance criteria, prioritize lower severity defects for future fixing via CR.

The final product of the test is twofold:

* A production-Ready software.
* A set of stable test scripts that can be reused for functional and UAT test execution.

## Test Principles

* Testing will be focused on meeting the business objectives, cost efficiency, and quality.
* Testing processes will be well defined, yet flexible, with the ability to change as needed.
* Testing activities will build upon previous stages to avoid redundancy or duplication of effort.
* Testing will be a repeatable, quantifiable, and measurable activity.

## Scope of Testing

### In Scope

Features that are in the scope of testing

|  |  |
| --- | --- |
| FR1\_G, FR11\_U, FR15\_A | Navigate Website |
| FR2\_G | Register |
| FR3\_G, FR16\_A, FR7\_U | Choose Category |
| FR4\_G, FR12\_U, FR17\_A | View Article |
| FR5\_U, FR18\_A, | Login |
| FR6\_U, FR19\_A | Logout |
| FR7\_U | Choose Category |
| FR8\_U | Publish an article |
| FR9\_U | Upload Video |
| FR10\_U | Upload Recorded Voice |
| FR13\_A | Delete User Account |
| FR14\_A | Delete User Content |

### Out Scope

Features that are not in the scope of testing:

User Interfaces - Hardware Interfaces - Software Interfaces - Database logical - Communications Interfaces - Website Security and Performance.

|  |  |
| --- | --- |
| FR4\_G | View Video |
| FR4\_G | View Record |

## Test Levels

### System Test

* **PURPOSE**: this test focuses on validating the complete and fully integrated software product and evaluating the end-to-end system specifications.
* **TESTER**: performed by Testing Team.
* **METHOD**: Since the business users are the most indicated to provide input around business needs and how the system adapts to them, it may happen that the users do some validation not contained in the scripts.
* **TIMING**: After all integrations of system is complete.

### User Acceptance Test

* **PURPOSE**: this test focuses on validating the business logic. It allows the end users to complete one final review of the system prior to deployment.
* **TESTER**: The UAT will be performed by the end user
* **METHOD**: Since the business users are the most indicated to provide input around business needs and how the system adapts to them, it may happen that the users do some validation not contained in the scripts.
* **TIMING**: After all other levels of testing are done. UAT test will be executed by the user after providing the last release.

## Test Techniques

### Black-Box Technique

* **PURPOSE**: Focuses on input and output of software applications and it is entirely based on software requirements and specifications.
* **TESTER**: Testing Team.
* **METHOD**: Tests created only based on SRS and testers must include both happy and negative scenarios to test every single possible case.
* **TIMING**: Within every test level and type.

### Experience-Based Technique “Exploratory Test”

* **PURPOSE:** Make sure critical defects are removed before the next levels of testing can start.
* **SCOPE:** First level navigation, User, Guest, and Admin modules.
* **TESTERS**: Testing team.
* **METHOD**: This Exploratory Testing is carried out in the application without any test scripts and documentation
* **TIMING**: At the beginning of each cycle.

## Test Types

### Functional Test

* **PURPOSE**: Functional testing will be performed to check the functions of application. The functional testing is carried out by feeding the input and validates the output from the application
* **SCOPE**: Functional testing should be done on all levels of the system (system - UAT) most of the scope should be more focused on the functionality of the product.
* **TESTERS**: Testing team.
* **METHOD**: Tests will be performed according to the functional scripts recorded in the test case sheet in Testing Folder.
* **TIMIING**: After Exploratory testing is completed.

### Change-related Test.

#### **Confirmation Tests**

* **PURPOSE**: Retesting the software product to make sure the previously posted bugs are fixed or not in the system.
* **TESTERS**: Testing team
* **METHOD**: The tests performed again to ensure the bugs have been fixed and tests passed.
* **TIMIING**: After the development team fixing the bugs found and before the regression tests executed.

#### **Regression Tests**

* **PURPOSE**: To discover any defects introduced or uncovered as a result of the changes in the software being tested or in another related or unrelated software components.
* **TESTERS**: Testing team
* **METHOD**: The tests performed again to ensure the bugs have been fixed and re-tests passed according to selection and prioritization. New tests created when needed
* **TIMIING**: After Confirmation tests executed.

## Test Acceptance Criteria

* Approved functional Specification document, Use Case diagram must be available.
* Test cases should be approved before the Test execution
* Development completed are done (passed).

## Test Deliverables

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Name | Author | Reviewer |
| Test\_Deliver\_01 | Test Plan | Project Manager | Analyst |
| Test\_Deliver\_02 | Functional Test Cases | Test Team | Project Manager |
| Test\_Deliver\_03 | Bug Report | Test Team | Project Manager |
| Test\_Deliver\_04 | Test Reports | Test Team | Project Manager |

# Execution strategy

## Entry and Exit Criteria

* The entry criteria refer to the desirable conditions in order to start test execution; only the migration of the code and fixes need to be assessed at the end of each cycle.
* The exit criteria are the desirable conditions that need to be met in order proceed with the implementation.
* Entry and exit criteria are flexible benchmarks. If they are not met, the test team will assess the risk, identify mitigation actions and provide a recommendation. All this is input to the project manager for a final “go-no go” decision.

|  |  |  |  |
| --- | --- | --- | --- |
| Exit Criteria | Test Team | Technical Team | Notes |
| 100% Test Scripts executed |  |  |  |
| 95% pass rate of Test Scripts |  |  |  |
| No open Critical and High severity defects |  |  |  |
| 95% of medium severity defects have been closed |  |  |  |
| 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 test script |  |  |  |
| Test Closer Memo Completed |  |  |  |

## Validation and Defect Management

* It is expected that the testers execute all the scripts. However, it is recognized that the testers could also do additional testing if they identify a possible gap in the scripts. If a gap is identified, the scripts and traceability matrix will be updated and then a defect logged against the scripts.
* The defects will be tracked through Traceability matrix and Trello. The technical team will gather information daily from Trello.
* It is the responsibility of the tester to open the defects, link them to the corresponding script, assign an initial severity and status, retest and close the defect; it is the responsibility of the Defect Manager to review the severity of the defects and facilitate with the technical team the fix and its implementation, communicate with testers when the test can continue or should be halt, request the tester to retest, and modify status as the defect progresses through the cycle.

Categories of Defects found during the Testing:

|  |  |
| --- | --- |
| Severity | Impact |
| 1 (Critical) | * This bug is critical enough to crash the system, cause file corruption, or cause potential data loss * It causes an abnormal return to the operating system (crash or a system failure message appears). * It causes the application to hang and requires re-booting the system |
| 2 (Major) | * It causes a lack of vital program functionality with workaround. |
| 3 (Moderate) | * This Bug will degrade the quality of the System. However there is an intelligent workaround for achieving the desired functionality - for example through another screen. * This bug prevents other areas of the product from being tested. However other areas can be independently tested. |
| 4 (Minor) | * There is an insufficient or unclear error message, which has minimum impact on product use. |

## Graphical user interface Description automatically generatedDefect Tracking & Reporting

# Test Management process

## Test Management tool

* Tasks: As every task in the project the tasks will be assigned to a member and to a specific deadline on TRELLO management tool.
* Test Cases: The test cases are written in the test case template provided by the team and approved by the project manager by Excel Sheets.
* Defects: The defect management system is done by writing a bug report with specific details that will be provided by the team by Excel Sheets.

## Test Design Process

* The tester will understand each requirement and prepare corresponding test case to ensure all requirements are covered.
* During the preparation phase, tester will use the prototype, use case and functional specification to write step by step test cases.
* Each of the Test cases will undergo review a peer and the review defects are captured and shared to the Test team
* Each Test case will be mapped to Requirements as part of Traceability matrix.

## Test Execution Process

* Once all Test cases are approved, tester will start an exploratory test of the application to ensure the application is stable for testing.
* Each Tester is assigned Test cases directly.
* If any showstopper during exploratory testing will be escalated to the respective development team member to fix.
* Each tester performs step by step execution and updates the executions status. The tester enters Pass or Fail Status for each test case directly.
* If any failures, defect will be raised as per severity guidelines detailing steps to simulate along with screenshots if appropriate.
* If there are any defects that are not part of steps but could be outside the test steps, such defects need to be captured in Traceability matrix and map it against the test case level or at the specific step that issue was encountered after confirming with Test team.
* This process is repeated until all test cases are executed fully with Pass/Fail status.
* During the subsequent cycle, any defects fixed applied will be tested and results will be updated.

# Test Environment and Tools

## Test Tools

* Requirements Tracking Tool.
* Bug Report Writing Tool.
* Test Case Writing Tool.

## Test Environment

The test environment should be look like the production environment for users but in this project the environment will be the development environment.

# Terms

|  |  |
| --- | --- |
| TERM/ACRONYM | DEFINITION |
| UAT | User Acceptance Test |
| AUT | Application Under Test |
| FR\_G | Functional requiremenets\_Guest |
| FR\_U | Functional requiremenets\_User |
| FR\_A | Functional requiremenets\_Admin |