TEAM ERROL

SOFTUNI BLOG

TEST PLAN

Document version	1.0
Version Date	03/14/2019
File name	SoftUni Blog – Test Plan
Document author	Elian Kurtev

Revision History:

Revision Date	Author / Reviser	Version	Changes Description

TABLE OF CONTENTS

1. Introduction

This test plan construe planned quality assurance goals, practices, tools and testing scenario for the SOFTUNI BLOG project. This test plan is going to be regularly bring up to date during the course of this project. The Quality Assurance team and the Project Manager are responsible for managing this document.

1.1 Purpose

The test plan will provide details on how testing will be performed for the SOFTUNI BLOG project and will provide a plan of action, the scope, approach and the resources too.

1.2 Project Overview

SOFTUNI BLOG is going to be mainly used by developers and tech geeks. The prime goal of this website is to keep the latest news around the IT world in one place. This website can be also accessed by everyone who wants to know about this field.

1.3 Scope

The scope of this document is to understand the testing methodology adapted for SOFTUNI BLOG.

2. Functionalities to be tested

ID	Feature to be tested	Description
1	Log in	A user of the website can access her/his profile through the log in portal.
2	Register /New account/	A guest of the website can make her/his own profile.
3	Publish an article	A user of the website can publish article/s which are published on the main page.
4	Edit an article	A user can edit an article which she/he created.
5	Delete an article	A user can delete an article which she/he created.
6	Read an article	A user or a guest can read an article on the main page of the website.
7	Change profile password	A user can change her/his password in the account information page.
8	Published articles	A user or a guest can see all articles published on the main page.
9	Log off	A user can log off of her/his profile.

These are the main features to be tested during the period.

3. Features not be tested

There will be no features not being tested.

4. Types of testing to be performed

The testing scenario includes the exact process that is going to be pursue during the testing

4.1 Unit testing

This type of testing will be performed by the developers. The purpose of the unit testing is to confirm that every small piece of the code is working properly as expected.

4.2 Integration testing

The integration testing will be carried out by the QA Specialists in order to verify that all the modules of the projects are working correctly between.

4.3 System testing

System testing is focused on the overall, end-to-end behavior of the system as a whole, both functional and non-functional. Below are described the system types that are going to be done during the testing of SOFTUNI BLOG.

4.3.1 Usability testing – It will mainly focus on the user's satisfaction to use the SOFTUNI BLOG and to check how usable the interface is.

4.3.2 Performance testing – This type of testing is focused on the load speed and the performance of the web page

4.3.3 Cross Browser testing – It is as process to test the web application across multiple browsers. Cross Browser testing ensures the compatibility of the website over different web browsers and the working preciseness.

4.3.4 Mobile web application testing – This testing will be performed to check if the website is responsible to different mobile devices.

4.3.5 Regression Testing

- Regression Testing involves testing done to make sure none of the changes made over the course of the development process have caused new bugs. It also makes sure no old bugs appear from the addition of new software modules over time.

4.4 Acceptance Testing

Acceptance testing, like system testing, typically focuses on the behavior and capabilities of a whole software product or system. The following acceptance testing will be executed by the Client.

4.4.1 User Acceptance Testing

– This type of testing of the system is typically focused on validating the fitness for use of the system by intended users in a real or simulated operational environment. The main objective is building confidence that the users can use the system to meet their needs, fulfill requirements, and perform business processes with minimum difficulty, cost, and risk.

4.5 Maintenance testing

4.5.1 Confirmation Maintenance Testing

- during this part of Maintenance Testing the modifications and errors are tested and retested until its execution becomes flawless. While retesting the original environment is maintained with the exact data inputs to make sure that no more errors occur and the validity of the modification or migration is confirmed without any doubts.

4.5.2 Regression Maintenance Testing

– once it has been confirmed that no more errors occur in the modification, it is now time to test that unintended defects have not spread elsewhere in the software.

During the test course will include and other techniques like Boundary Value Analysis and Equivalence Partitioning.

5. Test phase entry criteria

The entrance criteria for each phase of testing must be met before the next phase can commence. Pre-requisite to begin testing includes:

- The correct versions of the code have been deployed in the integration environment.
- Unit tests have been completed successfully.
- All the necessary documentation, design, release notes and requirements information should be available that will allow testers to operate the system and judge the correct behavior.
- All the standard software tools including the testing tools must have been successfully installed and functioning properly.
- Proper test data is available.
- QA resources should have sound knowledge of the functionality.

6. Test phase exit criteria

Requirement for successful exit include:

- All planned testing activities have been completed.
- All the test cases have been executed.
- All the defects are documented in the bug report document.
- All the defects have been fixed, retested and resolved.

7. Testing Tools & Browsers

Tool	Version	Description
Visual Studio	2017	All unit tests and integration tests will be executed in VS.
Postman	7.0.6	Additional integration tests should be executed in Postman.
		Performance testing

7.1 Tools

7.2 Browsers

Browser	OS Version
Microsoft Edge v. 44.17763.1.0	Windows 10 Home
Google Chrome v. 73.0.3683.75	Windows 10 Home
Safari v. 12.1.4	iOS 12.1.4
Google Chrome v. 73.0.3683.75	Android 9

8. Test scenarios/cases

All the test cases will be stored in Excel document. The main components included are:

- **ID** of the test case.
- **Priority** which shows how fast the problem should be solved.
- A short **title** of the test case.
- **Pre-condition** which is specifying the condition before the real steps execute.
- **Steps to reproduce** are the execution steps which need to be followed, to perform a certain test.
- **Expected result** after executing the steps of the test.
- **Pass/Fail** criteria after executing the steps of the test.

9. Issues tracking/bug report

All founded issues will be filled as actual result in the test cases document and will be reported in the bug report document too. The following elements will be provided for each issue:

- **ID** of the bug.
- **Priority** which shows how fast the problem should be solved.
- **Severity** showing what is the damage on the software.
- A sort **title** of the issue.
- **Pre-condition** which is specifying the condition before the real steps execute.
- **Steps to reproduce** are the execution steps which need to be followed, to perform a certain test.
- **Expected result** is the conclusion of executing the steps of the test.
- **Actual result** the real outcome of the executed steps.
- **Notes** can be written some free text.