Test Plan – Ensek Candidate Test webpages

|  |  |
| --- | --- |
| Document Information | |
| Project Name | Ensek Candidate Test webpage |
| Project Number | N/A |
| Project Account Code | N/A |
| Author | Dilip Vaghela |
| Issue Date | 17/11/2024 |

|  |  |  |  |
| --- | --- | --- | --- |
| Document History | | | |
| Version | Date | Summary of change | Author |
| 1.0 | 17/11/2024 | Release version | Dilip Vaghela |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Approvals | | | |
| Name | Title | Signature | Escalation |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| DOCUMENT CIRCULATION | | | |
| Version | Date | Person/Group | Author |
|  |  |  |  |

CONTENTS

[1. HOW TO USE THIS DOCUMENT 4](#_Toc182791800)

[2. INTRODUCTION 4](#_Toc182791801)

[3. OBJECTIVES 4](#_Toc182791802)

[4. SCOPE 5](#_Toc182791803)

[4.1 Test items 5](#_Toc182791804)

[4.2 Functional testing 5](#_Toc182791805)

[4.2.1 Integration – Feature testing 5](#_Toc182791806)

[4.2.2 Integration – Exploratory testing 5](#_Toc182791807)

[4.2.3 System – Regression testing 5](#_Toc182791808)

[4.2.4 Integration – Accessibility testing 5](#_Toc182791809)

[4.2.5 System – Selenium tests 5](#_Toc182791810)

[4.3 Non-functional testing 5](#_Toc182791811)

[4.3.1 System - Lighthouse audit 5](#_Toc182791812)

[5. Out of scope 6](#_Toc182791813)

[5.1 Test items 6](#_Toc182791814)

[5.1 Functional testing 6](#_Toc182791815)

[5.1.1 Unit – Unit tests 6](#_Toc182791816)

[5.1.2 Acceptance – UAT testing 6](#_Toc182791817)

[5.2 Non-functional testing 6](#_Toc182791818)

[5.2.1 Integration - Analytics testing 6](#_Toc182791819)

[5.2.2 System - Load testing 6](#_Toc182791820)

[5.2.3 System - Penetration testing 6](#_Toc182791821)

[5.1 Approach / Deliverables 7](#_Toc182791822)

[5.2 Functional testing 7](#_Toc182791823)

[5.2.1 Integration – Feature testing 7](#_Toc182791824)

[5.2.2 Integration – Accessibility testing 7](#_Toc182791825)

[5.2.3 System – Selenium tests 7](#_Toc182791826)

[5.3 Non-functional testing 7](#_Toc182791827)

[5.3.1 System - Lighthouse audit 7](#_Toc182791828)

[5.4 Schedule 7](#_Toc182791829)

[5.5 Defects & Metrics 8](#_Toc182791830)

[5.5.1 Defects 8](#_Toc182791831)

[5.5.2 Test Metrics 8](#_Toc182791832)

[5.6 Resources & Responsibilities 10](#_Toc182791833)

[5.6.1 1Roles and Responsibilities 10](#_Toc182791834)

[6. RISKS & CONTINGENCIES 11](#_Toc182791835)

[7. SIGN OFF / APPROVALS 12](#_Toc182791836)

[8. APPENDIX 13](#_Toc182791837)

# HOW TO USE THIS DOCUMENT

This document is an example of a Software Test Plan. It uses example sections from various plans and is not a single live document. The intention is to show what type of information should be incorporated into a Test Plan, and is not representative of a single project.

To ensure a well-tested product, testers need a thorough understanding of the context of the project, the project goals, the system functionality, the system risks and vulnerabilities, the test equipment, and the procedures and tools to be used. The Software Test Plan aims to capture some of this information.

It is important to capture as much information about the testing requirements as early as possible within the Software Development Life Cycle (SDLC). Early involvement of testers means less of a learning curve to master the system, more time to determine how to test, time to prepare testing and the test environment, less cost to make changes and adjustments. The test team should actively participate in developing the overall project work plan, contributing to work product breakdown, scheduling, and providing the testing estimates

# INTRODUCTION

* Website has been created to showcase the following
  + Able to learn about the company
  + Able to contact the company
  + Able to buy and sell energy
  + Able to register to buy and sell energy
  + Able to sign in as existing user

# OBJECTIVES

* *To test the functionality exercised by one part of the website does not cause conflicts or issues elsewhere in the site.*
* *To ensure that the products functionality performs as of the User requirements*

# SCOPE

## Test items

This section of the plan will outline those ideas which are in scope for this testing project.

## Functional testing

### Integration – Feature testing

Does the new feature behave as it should for our users?

New feature is tested against business defined acceptance criteria

### Integration – Exploratory testing

Testing the site without any test cases or acceptance criteria. Learning, exploring and trying ad-hoc behaviours. Using their knowledge, intuition, and experience to the test the site.

### System – Regression testing

Testing the site to see if existing changes have not regressed

### Integration – Accessibility testing

Testing the site to see how usable a site is for people with disabilities or other unique challenges.

### System – Selenium tests

Automated test scripts to ensure the system is working as indented. The steps are executed by a framework with different browsers and viewpoints.

## Non-functional testing

### System - Lighthouse audit

Audit conducted to review / score the site based on performance, accessibility, best practice and SEO.

# Out of scope

## Test items

This section of the plan will outline those ideas which are out of scope for this testing project.

## Functional testing

### Unit – Unit tests

Automated test scripts created to test elements / components at code level

### Acceptance – UAT testing

The site is tested is tested by the end users or client in real world situations.

## Non-functional testing

### Integration - Analytics testing

Evaluates the correct levels of tagging / metrics are being sent to the analytics platform

### System - Load testing

Evaluates how a site behaves under normal and peak load conditions

### System - Penetration testing

Evaluates security and vulnerabilities of the sites.

## Approach / Deliverables

This section should also contain the tools to be used:

## Functional testing

### Integration – Feature testing

* Testcases – Excel document
* Defect tracker – Excel document

### Integration – Accessibility testing

* [WAVE](https://wave.webaim.org/) add on – Chrome extension
  + PDF report will be generated

### System – Selenium tests

* Following configuration will be used:
  + Test runner: Nunit
  + Framework: Selenium
  + Programming language: C#
  + IDE: Jetbrains Rider
* Framework and test scripts – Git repository

## Non-functional testing

### System - Lighthouse audit

* [Lighthouse](https://developer.chrome.com/docs/lighthouse/overview) add on – Chrome dev tools
  + PDF report will be generated

## Schedule

This section outlines the major dates associated with this plan – it is not a project plan but gives readers the testing milestones.

|  |  |  |  |
| --- | --- | --- | --- |
| ***Milestone Task - Phased Plan*** | ***Effort*** | ***Start Date*** | ***End Date*** |
| *Feature testing – Test cases creation* | *30m* | *17/11/24* | *17/11/24* |
| *Feature testing – Test cases execution* | *30m* | *17/11/24* | *17/11/24* |
| *Exploratory testing* | *1h* | *17/11/24* | *17/11/24* |
| *Accessibility testing* | *30m* | *17/11/24* | *17/11/24* |
| *Selenium tests – Test script creation* | *1h* | *17/11/24* | *17/11/24* |
| *Selenium tests – Test script execution* | *30m* | *17/11/24* | *17/11/24* |
| *Lighthouse audit* | *30m* | *17/11/24* | *17/11/24* |

## Defects & Metrics

### Defects

This section highlights how defects found during testing are to be classified and managed

*Any* ***defects*** *identified during testing will be logged in within a excel document with thr correct priority mentioned below.*

|  |  |
| --- | --- |
| ***Priority*** | ***Description*** |
| *0 – Showstopper (Fix immediately)* | *Showstopper error. Halts release, halts testing* |
| *1 – Urgent (Fix this Release)* | *Fatal error/crash, GPF, data corruption, infinite loops, stopping test* |
| *2 – High (Fix for next release)* | *Fatal errors in less critical areas, wrong answers to calculations, performance, resource problems, functionality not responding in specified manner, etc* |
| *3 - Medium (To be fixed)* | *Non-critical errors, difficult to use, screen/print corruption etc* |
| *4 – Low (Being considered)* | *Ideas to improve* |
| *5 – As Agreed* |  |
| *6 – As and When* |  |

*With challenging test schedules, change control must be exercised on all defects. However, it is expected that some defects found will not be fixed. Any defects that are found will be evaluated as to the risks associated to the release, before they are fixed.*

### Test Metrics

Testing metrics are an import aid to knowing progress on a test project. The overall number of defects, the rate at which defects are being found, and their severity – are all useful information in assessing the quality of the product under test.

For Example:

*Project Test metrics will be produced on a weekly basis during testing.*

*Total Defects*

*Outstanding Defects*

*Defect Severity Analysis*

*Defect Discovery Rate – (see below for example)*

**

These should be published with the project highlight reports*.*

## Resources & Responsibilities

### 1Roles and Responsibilities

This section details the expected workload to be undertaken by the test team and project members required to support the test function. Any additions or modifications will be discussed with the Project Manager.

*For Example:*

|  |  |  |  |
| --- | --- | --- | --- |
| ***Staff Name:*** | ***Role:*** | ***Deliverable or Work Package:*** | ***Responsibility:*** |
| *Teddy Tester* | *Lead System Tester* | *Test Plan*  *Test Schedule*  *Test Estimates*  *Test Scripts* | *Management of work products and the overall test lead and management of the test activities in addition to the production and management of integration test deliverables. It is also the responsibility of the integration test lead to facilitate communication and cooperation across project teams, and to liaise with project management* |

# RISKS & CONTINGENCIES

This section details any risks and contingences for the testing (as against the overall project)

For Example:

|  |  |  |
| --- | --- | --- |
| *Risk Or Issue* | *Risk Level* | *Likely Effects And Mitigation* |
| *Insufficient time for testing to be complete* | *High* | *Test coverage may not meet exit criteria. Focus testing on critical and high risk areas as defined by Cross Dependencies matrix and risk analysis.* |

# SIGN OFF / APPROVALS

All test plans are subject to sign-off – people within the programme should know how their products are going to be tested, and agree with the planned action. If not they should raise their issues and have the plan changed, if not their objections must be recorded.

For Example:

|  |  |  |  |
| --- | --- | --- | --- |
| ***Name:*** | ***Appointment:*** | ***Approval Signature:*** | ***Signatory Remarks:*** |
|  | *Project Executive* |  |  |
|  | *Product Manager* |  |  |
|  | *Release Manager* |  |  |
|  | *Project Manager* |  |  |
|  | *Test Manager* |  |  |
|  | *Business Analyst* |  |  |

# APPENDIX

Anything not quite fitting anywhere else can be added into this section. For example a list of escalation fixes and descriptions which are being delivered into a new version of the software but which were not part of the original project.