*Test Plan -*

*By Arsh Hasan*

*ChangeLog*

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Change Date** | **By** | **Description** |
| version number | Date of Change | Name of person who made changes | Description of the changes made |
| 1 | 29/07/2024 | Arsh Hasan | Initial version of the test plan document. Included test strategies, scope, quality objectives, roles, and responsibilities. |

1 Introduction 2

1.1 Scope 2

1.1.1 In Scope 2

1.1.2 Out of Scope 2

1.2 Quality Objective 2

1.3 Roles and Responsibilities 2

2 Test Methodology 3

2.1 Overview 3

2.2 Test Levels 3

2.3 Bug Triage 3

2.4 Suspension Criteria and Resumption Requirements 3

2.5 Test Completeness 4

3 Test Deliverables 4

4 Resource & Environment Needs 4

4.1 Testing Tools 4

4.2 Test Environment 4

5 Terms/Acronyms 5

# Introduction

This test plan document outlines the test strategies, processes, workflow, and methodologies used for the project "Number to Words Converter". The primary objective is to ensure the application meets both functional and non-functional requirements and is of high quality.

## Scope

### In Scope

* Testing the functionality of converting numbers to words.
* Validating both positive and negative input cases.
* Testing the React front-end for user interactions.
* Ensuring API responses are correct and handled properly in the front-end.
* Unit tests for service and controller layers in the ASP.NET Core back-end.
* Integration tests to ensure communication between React and ASP.NET Core API.

### Out of Scope

* Performance and load testing.
* Security testing.
* Testing with external APIs or databases that are not part of the project.
* Multi-language support testing.

## Quality Objective

The objective of the testing is to ensure:

* + - * The application conforms to the functional and non-functional requirements as outlined in the test pdf document.
      * The application meets the quality specifications defined by the client.

## Roles and Responsibilities

* + - * Developer and Test Manager – Arsh Hasan
      * Writing Test Cases and identifying potential issues and fixing them.

# Test Methodology

## Overview

Since the task is a small web application, and will not require extensive testing apart from unit testing, system testing and integration testing, I aim to employ Aglie test methodology to ensure the application developed meets the requirements.

## Test Levels

* **Unit Testing**: Testing individual components (React components, services, and controllers).
* **Integration Testing**: Ensuring that the React front-end communicates correctly with the ASP.NET Core API.
* **System Testing**: End-to-end testing of the entire application, including user interactions and API responses.

## Bug Triage

The goal of the triage is to

* Define the type of resolution for each bug.
* Prioritize bugs and determine a schedule for all "To Be Fixed" bugs.

## Suspension Criteria and Resumption Requirements

Suspension criteria define the criteria to be used to suspend all or part of the testing procedure while Resumption criteria determine when testing can resume after it has been suspended

## Test Completeness

Testing will be considered complete when:

* 100% of test cases have been executed.
* All manual and automated test cases are executed.

# Resource & Environment Needs

## Testing Tools

The following tools are required to test the task:

Automation Tools: Jest, React Testing Library, xUnit, Moq

## Test Environment

Following **software’s** are required in addition to client-specific software.

1. Windows 8 and above

# Terms/Acronyms

Make a mention of any terms or acronyms used in the project

| TERM/ACRONYM | DEFINITION |
| --- | --- |
| API | Application Program Interface |
| AUT | Application Under Test |