Fibonacci Calculator

Test Plan Document

Prepared by:

Prerana Shrestha

**Fibonacci Calculator Frontend and Backend Testing**

1. **Introduction**

This document details the high-level test approach for the Fibonacci Calculator. It outlines the testing flow, the test activities, and the specific responsibilities of the various organizations/entities involved in testing and/or certifying the feature functionality to be delivered with the upcoming release.

1. **Testing Approach**

The following lists the high level tasks that will be undertaken to ensure a planned and methodical approach for testing the delivered functionality:

1) Requirements clarification sessions will be conducted with the stakeholders, development and functional analysis teams.

2) Based on documented requirements, positive and negative test cases will be built with proper steps and testing pre-requisites.

3) Test case reviews will be undertaken to ensure development and functional analysis teams are on board with the test coverage. If any changes are recommended and agreed upon during the review meeting, the QA team will implement those changes to the test cases.

4) Test data will be created to support the testing effort if necessary.

5) After the development of the product, Test scripts will be created and stored (Example: in Quality Center).

6) During final QA phase, the scripts and manual test cases will be executed with results stored (Example: in Quality Center).

7) Defects will be opened (example: in Quality Center) and processed through the appropriate workflow.

8) Once all the test cases are cleared, regression scripts will be run to ensure new changes have not broken existing functionality.

1. **Testing Environment**

The following environmental requirements will need to be satisfied to support the testing outlined in this approach document.

*Development and QA environments must be available for testing. Once the code is certified in the development environment, it will be moved to the QA environment.*

1. **Individual Testing Approaches**

This section will describe the high-level test approach for each organizational/entity, including QA, that will be involved in testing the feature-functionality requested in the Business Ask(s).

The QA team will test the before mentioned scope that has been committed to. Any proposed changes made post-content commit will need to go through proper planning and risk analysis by both the DEV and QA teams prior to being accepted.

**QA Team Test Approach**

The following sub-sections outline the high-level test approach, the test phases, and the types of testing that will be performed by QA Team to support the feature-functionality in this Project.

**1.1.1 Test Phases**

The QA team will be responsible for testing during the Release testing. Testing will include the following test types:

Front End and Back End Functional Testing – xx/xx/2016 to xx/xx/2016

**1.1.2 Test Assumptions**

The following assumptions were made to support the above testing approach:

1) Development team has completed unit testing and delivered the code on the planned code drop date.

2) No new requirements are going to be added after requirements freeze date

**1.1.3 Test Timeline**

This section defines the overall timeline for the test phases identified by the QA Team to support its portion of the Release:

Example: Development/Unit Test/Performance Testing/Prod Soak Deployment/Database Deployment/ Go Live

These estimates were based on information available at the time this document was drafted and may not represent the final estimate of test effort needed.

1. **Testing Approach:**

This section will describe the different testing approaches that will be performed by the member of a QA team to test the feature and functionality requested in the Business Asks.

1. Functional Testing
2. Regression Testing
3. Any other testing needed according to the need of the business.
4. **Overall Testing Risks & Constraints**

This section identifies the testing risks that are being identified as a result of the testing outlined in this approach document.

1. **High Level Test Scenarios (Functional and End-to-End Testing)**

Test Scenarios:

Access correct URL

Access incorrect URL

Passing strings character instead of numbers

Passing numbers over 46, 47, 48, 79 and 90

Passing negative numbers

Passing special characters

1. **Sign-off**

The following individuals have agreed to the contents of this approach document on behalf of their organization.