Quality Assurance Test Plan

WGU Student ID | **011933973**

A. Overview

1. **Summary of the Proposed Software Design Plan from Task 1**

The proposed software design plan addresses a critical issue within Endothon Finance's web application, where the app incorrectly retrieves financial data from the first five fiscal years of a business's operation rather than the most recent five years. This flaw significantly impacts the accuracy of loan profiles, leading to potential errors in loan decisions. The solution involves revising the app's logic to correctly identify and request the relevant fiscal years based on the business's age, ensuring accurate data collection for both older and newer businesses.

2. **Overall Objective of the Functional Requirements to be Tested**

The primary objective of the functional requirements in the quality assurance process is to ensure that the web application correctly identifies and requests the appropriate fiscal years based on the business's establishment date. This will involve testing the logic that determines whether the app requests the last five completed fiscal years or a combination of historical data and forecasted future years for newer businesses.

a. **Quality Metrics for Functional Requirements**

- **Accuracy:** This metric is crucial to verify that the app accurately selects the correct fiscal years, ensuring the reliability of the loan profiles generated.
- **Response Time:** It is essential that the app retrieves the financial data quickly, with a response time of under 2 seconds, to maintain a seamless user experience.
- **Data Completeness:** This metric ensures that the app collects all required financial data for each loan application, covering all relevant fiscal years.

3. **Overall Objective of the Non-Functional Requirements to be Tested**

The objective of the non-functional requirements is to validate that the web application meets performance standards and provides a user-friendly experience. This includes ensuring that the app is responsive across different network conditions and that it is intuitive and easy to navigate for users.

a. **Quality Metrics for Non-Functional Requirements**

- **Performance:** The app should demonstrate optimal load times under various conditions, ensuring that it remains functional and efficient for users, even in low bandwidth scenarios.
- **Usability:** This metric will focus on the ease of navigation and overall user satisfaction, ensuring that the user interface is straightforward and that tasks like financial data entry are easy to complete.

B. Project Scope

4. **In-Scope Functional Requirements**
    - **Correct Identification of Fiscal Years:** The QA process will test the app's ability to correctly identify and retrieve the most recent five fiscal years for businesses that have been in operation for over five years.
    - **Accurate Retrieval for Newer Businesses:** The QA process will also ensure that the app accurately retrieves historical data and generates future forecasts for businesses established within the last five years.
5. **In-Scope Non-Functional Requirements**
    - **Performance:** The QA process will assess the app's load times and responsiveness to ensure it meets performance standards across different network conditions.
    - **User Interface:** The QA process will evaluate the intuitiveness and ease of navigation within the user interface, particularly during the financial data entry process.
6. **Out-of-Scope Functionalities**

a. **Integration of Additional Financial Forecasting Tools**

- **Alignment with Business Requirements:** While integrating advanced financial forecasting tools could potentially enhance the web app's functionality, it does not address the immediate issue of correcting the logic for data retrieval based on the business's age.

- **Reason for Being Out of Scope:** This functionality is labeled as out of scope because the current focus is on resolving the existing data retrieval logic error rather than expanding the app's capabilities with new forecasting features.

b. **Development of a Mobile Application Version**

- **Alignment with Business Requirements:** Creating a mobile version of the web app could improve accessibility for users on different devices, but it does not directly address the core issue of incorrect fiscal year data retrieval.
- **Reason for Being Out of Scope:** This functionality is considered out of scope as the primary goal is to ensure the current web application functions correctly. Developing a mobile app would be an entirely new project requiring additional resources and time.

[C. Test Cases on next page]

C. Test Cases

1. Testing Overview

Test Case Table				
Test Type	**Description of Test**	**Objective**	**Test Owner**	**Environment**
Unit Test	This test will validate the logic that identifies the correct fiscal years for data retrieval based on the business creation date and the current year. The test will input various business creation dates and the current year to ensure the system accurately selects the five most recent fiscal years or a combination of available years and forecasts, as required. **Sample Input:** Business Creation Date - Jan 1, 2000 Current Year - 2024 **Expected Output:** The system should identify and return the years 2019, 2020, 2021, 2022, and 2023 as the relevant fiscal years for data retrieval.	Ensure the accurate collection of fiscal years based on the business age.	Software Developer	Development Environment
Integration	This test will verify the entire data flow from identifying the correct fiscal years to retrieving the corresponding financial data. It will	Ensure accurate data collection based on the	QA Engineer	Testing Environment

	ensure that once the fiscal years are identified, the system correctly fetches and compiles the required financial data for those years, validating that all components of the process work together seamlessly. **Sample Input:** Business Creation Date - Jan 1, 2000 Financial Data for Years - 2019-2023 **Expected Output:** The system should correctly collect and display financial data for the years 2019, 2020, 2021,2022, and 2023.	identified fiscal years.		
Performance	This test will evaluate the web application's load time under various network conditions, including high, medium, and low speeds. The goal is to ensure the application loads within acceptable timeframes across all conditions, meeting the target of 3 seconds for high speeds, 4 seconds for medium speeds, and 6 seconds for low speeds. **Sample Input:**	Ensure fast performance for a better user experience.	QA Engineer	User Testing Environment

	Simulated Network Speeds - High (100Mbps), Medium (10Mbps), Low (1Mbps) **Expected Output:** Load times should be under 2 seconds at high speed, under 4 seconds at medium speed, and under 6 seconds at low speed.			
Usability	This test will assess the usability of the web application by simulating a user navigating through the financial data entry form. It will measure the time taken to complete the form, ensuring that users can input all necessary data within 2 minutes, thereby validating the interface's intuitiveness and ease of use. **Sample Input:** A user attempting to complete the financial data entry form with provided financial data for the identified fiscal years. **Expected Output:** The user should be able to complete the financial	Ensure the interface is user-friendly	UX Designer	User Testing Environment

	data entry within 2 minutes with no errors or confusion			

2. Sequence of Testing

The testing process will begin with unit testing, focusing on validating the foundational logic that identifies the correct fiscal years based on the business creation date and the current year. This step is essential as it establishes the accuracy of the logic that underpins all subsequent operations. After confirming the logic through unit tests, the process will move on to integration testing, where the entire workflow will be assessed. This involves ensuring that the system accurately identifies the fiscal years and retrieves the corresponding financial data, confirming that all components work together seamlessly.

Following successful integration testing, performance testing will be conducted to evaluate the web application's load times under various network conditions, ensuring that it meets user expectations for speed and responsiveness. Finally, usability testing will take place, focusing on the user experience to ensure that the application is intuitive and user-friendly. This step will verify that users can efficiently navigate the application and complete necessary tasks, thus enhancing the overall user experience. This sequence ensures that the core functionality is thoroughly validated before moving on to performance and user experience evaluations, providing a comprehensive approach to quality assurance.

D. Sources

GeeksforGeeks. (n.d.). *Software testing techniques*. https://www.geeksforgeeks.org/software-testing-techniques/

Indeed Editorial Team. (n.d.). *Testing methodologies: Overview and types*. Indeed. https://www.indeed.com/career-advice/career-development/testing-methodologies