**Quality Assurance Test Plan**

|  |  |
| --- | --- |
| **WGU Student ID** |  |

# A. Overview

## 1. Software design plan summary

The current software for the web application has an issue with identifying the age of a business and providing appropriate financial data based on the age. It currently provides financial information for the first five years of a business. However, the desired outcome is that the financial information for the latest five years of a business that is older than five years; if a business is younger than five years the software will provide the financial information for the business for the most recent years and then projected financial data for future years until an accumulated five years of financial data will be available to the user. The goal of this document is to identify the issue that is causing this problem and provide a viable solution to produce a correction to the web application that will provide the desired results to the user.

## 2. Functional requirements objective

One of the primary functional requirements to fix the user experiences issue is to determine if the business is older than five years or not. If this critical step is not operating properly then the proper further levels of data manipulation cannot occur to provide the user the appropriate information that should be displayed. To provide the user with appropriate data and for other functions to be able to manipulate the data appropriately this critical step must be made sure it is functioning correctly. The metric for this requirement, although quite simple, is to determine the fact that a business is older or younger than five years is correctly determined. Ideally, this should be a simple matter coding by comparing the creation date of the business (after verifying that the data for a business is accurate) against the current date (prefferably obtained from an outside source independent from the software like the time date stamp of the operating system) to determine the age of the business being profiled. A positive test would accurately determine the age of a business, otherwise any result would represent a failure.

If a business is older than five years the loan profile the business should display the most recent five years of the business’s financial information. This is dependent on the correct identification of the previous item, accurately determining the correct age of a business. If the correct age of the business is not determined than the further stage will be deeply impacted.

A determination whether the information displayed is accurate to the information stored within the database. A possible concern is that when a request is made for financial information it is accurately being displayed, however somehow withing the code it is being represented as financial information for years that it is not tied to. For example, the data is displaying the information for a business established in 2000 saying that the information is from 2000, 2001, 2002, 2003, 2004, 2005. It must be verified if that information is for those specific years or if it is actually data for the previous five years that has been mislabeled for older years.

### 2a. Functional requirements objective metrics

The metrics for the properly functioning web application are actually quite simple and can summarize succinctly with the following:

1. Identify correctly if a business is older or younger than five years. This is a yes or no scenario that should be easily identifiable. Furhermore, this is a crucial step due to the fact that it impacts the progression of the web application when interpreting and displaying appropriate financial data.
2. If a business is older than five years the most recent five years of financial data is displayed. The algorithm should be able to receive an age of a business and if it is older than five years it should retrieve and then display the previous five years of the business’s financial information.
3. Verifying that what is displayed to the user is accurate to what is displayed to the user. Particularly if the information is correctly labeled by the software. For example, if a record shows that the information for the business is from 2025, the information contained in the database for 20025 is what is being displayed.

## 3. Non-functional requirements objective

Financial data is involved throughout the entirety of this projects utilization. Therefore, security and encryption of data is paramount. This provides a layer of protection primarily for the benefit of the customer (whose financial information is being reviewed), but also for the company that is required under certain laws to verify confidentiality of certain information. A test should be made to determine that the financial information that is being broadcast across is correctly and strongly encrypted.

“Time is money”. This is particular potent for a business that makes financial decisions. The business is at a functional standstill because accurate loan profiles are not being provided. Therefore, financial decisions concerning loan legitimacy and authorization cannot be determined.

### 3a. Non-functional requirements objective metrics

Encrypt, encyrpt, encrypt. Data can be snatched up and used for a variety of nefarious deeds. This is particularly important for a business that primarily deals with financial information. A test should be conducted to ensure that this financial information is being encrypted properly based on local regulations. Correct encryption of the information is a must considering the importance of information being transmitted.

Business operations are being considerably affected currently…a loan applications financial data is not accurately being presented to a user of the business. This is causing delays in the approval process. Thereore, a very specific DESIRED timetable to implement a solution be outlined. This document says “desired”, but well programming. The current problem causing the bug is not identified, but it should be top priority in order to reduce affects on the business. A clear timetable should be identified so team members have a metric to realize their deliverables’ and integration of the financial solution can occur within the desired time frame.

# B. Scope

## 1. In-scope functional requirements

The first in-scope functional requirement to test is if a correct determination of whether a business is older or younger than five years.

The second in-scope functional requirement to test is displaying the previous five years of financial data for a business older than five years.

## 2. In-scope non-functional requirements

The first in-scope non-functional requirement to be tested is the correct encryption of data.

The second in-scope non-functional requirement to test is integration of the solution created to fix the bug in order to determine if other problems arise due to the solution.

## 3. Out-of-scope functionalities

A business less than five years old will not be tested.

The ability to forecast financial information for a business will not be tested.

### 3a. Out-of-scope functionalities explanation

The primary endeavour of this project is to ensure that financial data for a business is being correctly displayed for a business based on a set of predetermined criteria. However, the ticket makes no mention of issues occurring with a business that is younger than five years, therefore that scenario does not fall in scope with the current issue being addressed.

Furthermore, a business less than five years old requires the ability to project financial data for future years until a total of five years has been accumulated. This does fall under the ability to correctly display financial data for a business, however the current focus and problem is with a business that is older than five years. A business that is older than five years does not require the use of projecting financial data, therefore the operation of this particular feature falls way beyond the scope of the current issue being addressed.

# C. Test Strategy

## 1. Testing overview

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case Table** | | | | |
| **Test Type** | **Description of Test** | **Objective** | **Test Owner** | **Environment** |
| Unit | Provide a business with a variety of creation dates. Code should be able to correctly identify business that are older than five years. | Identify businesses that are older than five years. | Programmers | Self contained environment away from original code with an IDE with debugging software for the programming language being utilized. |
| Unit | When provided with a business that is correctly identified as older than five years the code will then correctly display the previous five years of financial information for the business. | Display the previous five years of financial data for a business older than five years. | Programmers | Self contained environment away from original code with an IDE with debugging software for the programming language being utilized. |
| Security Testing | The test take packets of financial data that are potentially being sent over a network by the application and verifying that the information has been correctly encrypted according to the security standards deployed for the application. It is expected that a secure packet will be encrypted according the encryption method utilized by the application | Test the encryption of financial data. | Security Specialist | Closed network between two computers that can simulate the transfer of data packets from one computer to another like the internet. |
| Integration Testing | The solution is implemented in a closed environment of the web application to determine if new problems are being caused by the proposed solution throughout the rest of the application. A correct implementation of the proposed solution would fix the issue brought forth by the ticket as well as cause no other issue to the existing application. | Implement the proposed solution and determine if other issues are caused by the fix implemented. | Programmers. | Self contained environment away from original code with an IDE with debugging software for the programming language being utilized. |

## 2. Sequence of testing

The first order of business to test is whether a business can be correctly identified as older than five years if its creation date is older than the current date by five years. If this is not correct from the very beginning it will cascade and cause multiple issues for the next stages of testing, which rely heavily on the ability to do this properly.

Once a business older than five years is correctly identified the testing phase can move on to testing the the ability to correctly display the desired financial information for the business. In this case that would be the five most recent years of financial information for the business in question. This test will verify if the previous five years of financial information is being displayed.

Once a viable solution has been created to fix the issue brought forth by the original ticket the solution now needs to be implemented in the application, or at least a replica of the original application contained on a self contained system. The reason for this is due to the fact that it is currently unknown if this fix will mesh well with the system or if it will cause further issues. This is the next step of testing.

Finally once the fix has been implemented in the test environment and it has been verified that it has caused no other programming issues with the application testing will occur to verify that the very important financial data is correctly being encrypted by the program after implementation of the fix.