COMP5202

Login System Test Plan

# Introduction

This test plan describes the testing process for Login System Program. It is designed to ensure the appropriate information is displayed and test for inappropriate inputs from the user.

# Test Items

Login System v0.1 (Alpha release)

The application consists of one file LoginSystem.sln which is required for the application to execute.

# Test Environment

The application is designed to run in Visual Studio.

No other specific systems required for testing.

# Scenario

Scenario 1 – Test what will happen if a user inputs the correct username and password and that the information is successfully received and stored correctly.

Scenario 2 – Test what happens if the user enters an invalid username and password.

# Testing Method

The testing for this project will involve two main types of testing:

* Black box Testing – Check the inputs and outputs are appropriate
* White box Testing – Inspection of the code

Both testing methods will require user input from the tester and will then evaluate the resulting conditions.

# Pass / Fail Criteria

Scenario 1 - The user enters the correct username and password to be successfully logged in.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Inputs** | | | | **Outputs** |
|  | **Username** | **Password1** | **Password2** | **Successful Login** |
| Expected | LoganTawhiri | password | password | Success |
| Actual | LoganTawhiri | password | password | **Success** |

Scenario 2 – The user inputs an incorrect username and is unable to login until the right inputs are provided.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Inputs** | | | | **Outputs** |
|  | **Username** | **Password1** | **Password2** | **Successful Login** |
| Expected | LoganTa |  |  | Unsuccessful |
| Actual | LoganTa |  |  | **Unsuccessful** |

Scenario 3 – The user enters the correct username but the password does not meet the length requirements.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Inputs** | | | | **Outputs** |
|  | **Username** | **Password1** | **Password2** | **Successful Login** |
| Expected | LoganTaw | passwor | passwor | Unsuccessful |
| Actual | LoganTaw | passwor | passwor | **Unsuccessful** |

Scenario 4 – The user enters the correct username but inputs 8 ‘spaces’ for the password.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Inputs** | | | | **Outputs** |
|  | **Username** | **Password1** | **Password2** | **Successful Login** |
| Expected | LoganTaw |  |  | Fail |
| Actual | LoganTaw |  |  | **Successful** |

# Modification Notes

Scenario 1 – Check if the output matches what is required and / or provide information back to the user if an issue is encountered.

Scenario 2 – Check that the username used is valid and provide meaningful feedback to the user so they understand that their inputs need to be looked at and possibly changed.

Scenario 3 – Check that the password used is valid and if not tell them that it is invalid and will need to be changed.

Scenario 4 – Check that the password used is valid and if not tell them that it is invalid and will need to be changed.

Changes had to be made here as the user could input all ‘spaces’ within the username and password prompt. The .trim() syntax was added to the end of the code that asks the user for inputs to prevent this from happening.