COMP5202

# Final Assessment 1st Program Testing plan

## Introduction

This program is designed based on the 1st program scenario for the final assessment of COMP5202. This program is designed to take a username and password from the user and check if those inputs are valid.

## Test Items

COMP5202\_FinalAssessment\_Program\_1.sln

## Test Environment

Visual Studio 2017 or later

## Scenario

Multiple users are going to follow the instructions given by the program.

## Testing Method

## Pass / Fail Criteria

Scenario 1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  | **Inputs** |  |  | **Outputs** |
|  |  |  | **Username** | **Password** | **Username again** | **Category** |
| Expected |  |  | Blake00 |  |  | Error Msg |
|  |  | Actual | Blake00 |  |  | Error Msg |
| Expected |  |  | LoganTaw | PasswordYes | PasswordYes | Pass |
|  |  | Actual | LoganTaw | PasswordYes | PasswordYes | Pass |
| Expected |  |  | SmileyFace | Sunny | Sunny | Fail |
|  |  | Actual | SmileyFace | Sunny | Sunny | Fail |

Scenario 2

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  | **Inputs** |  |  | **Outputs** |
|  |  |  | **Username** | **Password** | **Username again** | **Category** |
| Expected |  |  | \*8 spaces\* |  |  | Error Message |
|  |  | Actual | \*8 spaces\* |  |  | Error Message |
| Expected |  |  | June1945 | XIXIggez | XIXIggez | Pass |
|  |  | Actual | June1945 | XIXIggez | XIXIggez | Pass |
| Expected |  |  | Augustus99 | wordPass | wordPass | Pass |
|  |  | Actual | Augustus99 | wordPass | wordPass | Pass |

Scenario 3

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  | **Inputs** |  |  | **Outputs** |
|  |  |  | **Username** | **Password** | **Username again** | **Category** |
| Expected |  |  | UserName99 | nameUser48 | nameUser48 | Pass |
|  |  | Actual | UserName99 | nameUser48 | nameUser47 | Fail w/ Error Msg |
| Expected |  |  | BlahBlah | BloopBloop | BlopBolop | Fail w/ Error Msg |
|  |  | Actual | BlahBlah | BloopBloop | BlopBolop | Fail w/ Error Msg |
| Expected |  |  | Pineapple231 | ApplePen | ApplePen | Pass |
|  |  | Actual | Pineapple231 | ApplePen | ApplePen | Pass |

Scenario 4

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  | **Inputs** |  |  | **Outputs** |
|  |  |  | **Username** | **Password** | **Username again** | **Category** |
| Expected |  |  | Blake9999 | ThisIsAPass | NoItIsn’t | Fail w/ Error Msg |
|  |  | Actual | Blake9999 | ThisIsAPass | NoItIsn’t | Fail w/ Error Msg |
| Expected |  |  | ICan’tThinkOfAName | PasswordQuestionMark | PasswordQuestionMark | Pass |
|  |  | Actual | ICan’tThinkOfAName | PasswordQuestionMark | PasswordQuestionMark | Pass |
| Expected |  |  | User92 |  |  | Fail w/ Error Msg |
|  |  | Actual | User92 |  |  | Fail w/ Error Msg |

## Modification Notes

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

Scenario 2 – Check that the values 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 if the password is entered correctly, then check if the second password is correct. If not return an error message stating the password was incorrect and prompt the user to re-enter the password.

Scenario 4 – Check if the user has entered nothing but spaces, if so return an error message that warns the user that the username they have tried to enter is invalid and prompt them to press enter to try again.