Test Summary for Lab 2
(Shruti Ratna Charugulla)

Contents

Purpose

This report provides a summary of the testing conducted to verify the user login demo application provided in Lab #2.

Application Overview

The application is a command-line based authentication server written in Python. It allows a user to do three things listed below.

- Create an account with a valid password.
- Log in with a valid user name and password.
- Change the password of an existing account.

The application uses a simple database file to keep track of user statuses. The file is located in a temporary folder that is unique to an operating system.

Types of tests executed

Manual testing

The login application was tested manually. A brief description along with the output of each manual test is provided below.

1. Attempted to log in with a non-existent user account resulted in an *Access Denied* message.
2. Attempted to log in with an invalid password also resulted in an *Access Denied* message.
3. Created a user account with valid password login that resulted in *SUCCESS* message.
4. Tested both of the password requirements. When a password that was not 7 to 12 characters long was used, the application threw the error *Creating user failed: Password must be 7-12 characters long.* Also, when a password that did not contain a combination of both lower case and upper case letters was not used, the application threw the error *Password must be a combination of lowercase and uppercase letters and numbers.*
5. Changed password with invalid credentials resulted in an *Access Denied* error message.

Automation testing - Workflow tests

Two test cases listed below were executed through the script by including them in a code block.

User can create an account and log in

As part of this test, a valid user was created and then logged in as the same user successfully.

User cannot log in with bad password

As part of this test, a valid user was created but attempted to logon with an incorrect password only to result in an error as expected.

Automation testing - Higher-level tests

As part of this, the *given-when-then* format in the behavior-driven development (BDD) approach was used to create a test via high-level keywords defined in the script.

User can change password

As part of this test, given a user who has a valid account, its password was changed. And then, the user was logged in successfully with the new password. Moreover, the same user could not login with the old password as expected.

Automation testing - Data-driven tests

As part of this suite of tests, a template was created to test whether creating a user with invalid passwords fail with appropriate errors. Both invalid length and invalid content were used in a series of passwords and tested via a data-driven approach successfully.

Testing scope

In Scope

Only the scenarios presented in the previous section were within the scope of the testing effort.

Out of Scope

- Performance testing was out of the scope of this testing effort.
- Only a limited number of users (at most two) were created in the database.

Metrics

Test cases

Test Cases Planned	Test Cases Executed	TCs Passed	TCs Failed
5	5	5	0

Defects

None

Test Environment & Tools

Below is a table that includes a brief overview of the software and hardware setups used to execute the test cases described in this report.

Automation testing tool	Robot framework
Computer hardware	GenuineIntel x86_64 1800 MHz, 8GB RAM
Automation test scripting	Python
OS	Ubuntu 14.04.5 LTS (GNU/Linux 3.13.0-52-generic x86_64)

Exit Criteria

- All planned test cases are to be passed.
- No critical defects found.

Conclusion/Sign off

All exit criteria were met and satisfied. This application can 'Go Live'.