**Test Plan**

Tester: Noel Jules Cabahug Date designed: Nov 26, 2024, 11:27 PM

Commit ID: 2255b3de0305bce88a595ef177d4a821bcbd9ef0 Date conducted:

Test Type & Test Type ID: White Box - 01

**Requirement Addressed**

Assess if the code logic and structure provide proper functionality for managing screen transitions, user authentication, and database interactions.

**Objective**

To ensure that the screen transitions, login and signup mechanisms, and database read/write logic, are implemented correctly and work as intended.

**Test cases**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Description** | **Input field** | **Input value** | **Expected Output** | **Actual Output** | **Pass/Fail** |
| Test the login functionality for both valid and invalid user credentials by simulating input field values for username and password. Ensure appropriate transitions and messages occur. | attempt\_login() | * Valid credentials * Invalid credentials. | * For valid credentials: Transitions to the dashboard screen and prints "Logged in successfully." * For invalid credentials: Remains on the login screen and prints "Invalid Credentials." |  |  |
| Ensure the signup mechanism handles all cases correctly by simulating input for new users, existing users, and mismatched passwords. Validate database updates and error messages. | attempt\_signup() | * New user credentials * Existing user credentials * Mismatched passwords | * For new users: Prints "Successfully registered" and transitions to the login screen. * For existing users: Prints "Account already exists." * For mismatched passwords: Prints "Passwords do not match." |  |  |
| Verify that the read\_user\_database method properly reads and parses the db.txt file into a dictionary without errors and outputs the correct data format. | read\_user\_database() | self | The method returns a dictionary representation of the database without throwing any errors, and all key-value pairs (username-password) are read correctly. |  |  |
| Validate that the update\_user\_database method correctly appends new user credentials to the db.txt file without affecting existing data or causing errors. | Update\_user\_database(username, password) | * New username * New password | New credentials are appended to the db.txt file, and no errors are raised. |  |  |
|  |  |  |  |  |  |