**Use Case 1: Register Account**

Function: register (String username, String password)

Input partition: unique | existing | empty (for username)

valid | invalid | empty (for password)

**Test #1 testValidReg ()**

Purpose of Test:

To verify if a user will be able to register with a unique username and valid password.

Test Environment:

Mac OS using JUnit

Test Input:

Unique username and valid password (ex. “admin”, “Admin001\*”)

Expected Result:

Should display the confirmation message and redirect to the login page.

Likely Problems:

It might not redirect to the login page after a successful registration.

**Test #2 testExistingUser ()**

Purpose of Test:

To verify if a user cannot register with a username that already exists.

Test Environment:

Mac OS using JUnit

Test Input:

Existing username and any password (ex. “user00”, “User001\*”)

Expected Result:

Should display the error message to have the user to choose another username.

Likely Problems:

The same username with different capitalization might be considered unique.

**Test #3 testInvalidPw ()**

Purpose of Test:

To verify if a user cannot register with a password that does not meet the requirements.

Test Environment:

Mac OS using JUnit

Test Input:

Unique username and password that does not meet the requirements (ex. “admin”, “admin”)

Expected Result:

Should display the error message to remind the password requirements.

Likely Problems:

A password that does not meet the requirements might not be detected.

**Test #4 testEmptyReg ()**

Purpose of Test:

To verify the registration page when either field is empty and submit button is clicked.

Test Environment:

Mac OS using JUnit

Test Input:

Empty username and/or empty password (ex. “”, “User001\*” / “admin”, “”)

Expected Result:

Should display the error message to remind all fields should not be empty.

Likely Problems:

The empty string might be accepted as a valid input.

**Use Case 2: Login to Account**

Function: login (String username, String password)

Input partition: correct | incorrect | empty (for both username and password)

**Test #1 testValidLogin ()**

Purpose of Test:

To verify if a user will be able to log in with a correct username and password.

Test Environment:

Mac OS using JUnit

Test Input:

Correct username and password (ex. “admin”, “Admin001\*”)

Expected Result:

Should redirect to the homepage.

Likely Problems:

It might not redirect to the homepage after a successful login.

**Test #2 testInvalidUser ()**

Purpose of Test:

To verify if a user cannot log in with a username that is not registered.

Test Environment:

Mac OS using JUnit

Test Input:

Invalid username and any password (ex. “user99”, “User001\*”)

Expected Result:

Should display the error message to have the user to check the username. The message will also remind the user to register first with the link provided.

Likely Problems:

The same username with different capitalization might be considered invalid.

**Test #3 testWrongPw ()**

Purpose of Test:

To verify if a user cannot log in with an incorrect password.

Test Environment:

Mac OS using JUnit

Test Input:

Valid username and incorrect password (ex. “admin”, “user00”)

Expected Result:

Should display the error message to have the user to check the password.

Likely Problems:

An incorrect error message might be displayed to the user.

**Test #4 testEmptyLogin ()**

Purpose of Test:

To verify the login page when either field is empty and submit button is clicked.

Test Environment:

Mac OS using JUnit

Test Input:

Empty username and/or empty password (ex. “”, “User001\*” / “admin”, “”)

Expected Result:

Should display the error message to remind all fields should not be empty.

Likely Problems:

The empty string might be treated as a valid input.