

Test Plan

Directly Inspired by User Stories

Designed to ensure correct performance across all included aspects of the system and to maintain proper communication between the client and server. This approach is best described as a grey box style of testing since each tester has examined the project's code visually, but during actual execution, the user ideally does not have direct access to the source code. Because each test targets a specific functionality or use case, the testing resembles functional and integration level checks, progressing from individual behaviors to full system interaction, which aligns more closely with bottom-up integration testing rather than pure unit testing.

Client

Connect + Disconnect Tests

1. Attempt to use an invalid server address as the client.

Pass Condition:

System must provide an appropriate error message, e.g, "Connection failed. Invalid server address or port")

2. Attempt to connect the client to the server with a valid IP address when the server is unreachable/not running.

Pass Condition:

System must provide an appropriate error message, e.g, "Unable to connect to server. Please try again later."

3. With an active server that is actively listening for connections, the client attempts to connect to the server with a valid IP Address.

Pass Condition:

Client must be able to connect to server successfully.

4. With a current connection between client and server, click on the disconnect button as the client.

Pass Condition:

Server successfully closes the socket and disconnects the client while the app remains open.

Register Tests

5. With a current connection between the client and server, attempt to register for a new account. Purposefully entering an invalid email format paired with a password attempt to complete registration.

Pass Condition:

System must provide an appropriate error message, e.g "Invalid email format"

6. With a current connection between the client and server, attempt to register for a new account. Enter a valid email address and a password that is less than 8 characters and attempt to complete registration.

Pass Condition:

System must provide an appropriate error message, e.g "Password must contain at least 8 characters."

7. With a current connection between the client and server, attempt to register for a new account. Enter a valid email address, but a password that contains a combination of characters and numbers where there are fewer than 3 letters or 3 numbers. Attempt to complete registration.

Pass Condition:

System must provide an appropriate error message, e.g "Password must contain letters and numbers!"

8. With a current connection between the client and server, attempt to register for a new account. Enter a valid email address and password but a username that is known to already be registered within database.

Pass Condition:

System must provide an appropriate error message, e.g "jglasby1 already exists!"

9. With a current connection between the client and server, attempt to register for a new account. Enter a valid email address, a password that meets all requirements, and a username that is not currently stored in the database.

Pass Condition:

System must accept the registration request, store the new user in the USER database, and display an appropriate success message, e.g “Successful registration”.

Login Tests

10. With a current connection between the client and server, attempt to log in using an invalid username and or password such that the username does not exist in the database.

Pass Condition:

System must provide an appropriate error message, e.g “Invalid username and or Password. Please try again.”

11. With a current connection between the client and server, attempt to log in using a valid username but an incorrect password.

Pass Condition:

System must provide an appropriate error message, e.g “Invalid username and or Password. Please try again.”

12. With a current connection between the client and server, attempt three consecutive invalid login attempts.

Pass Condition:

System must lock the account, send recovery instructions to the associated email, and display an appropriate error message, e.g “This account has been locked. Check email for instructions to recover your account.”

13. With a current connection between the client and server, attempt to log in using a valid username and correct password.

Pass Condition:

System must successfully authenticate and transition the client into the main interface.

Logout Tests

14. With a current connection between the client and server, and with the user already logged in, select the option to log out.

Pass Condition:

System must de-authenticate the user, terminate the session, and redirect the client back to the login page.

Change Password Tests

15. With a current connection between the client and server, and with the user already logged in, attempt to change the password by entering a new password that meets all requirements.

Pass Condition:

System must accept the new password, update it in the database, and display an appropriate success message.

16. With a current connection between the client and server, and with the user already logged in, attempt to change the password by entering a new password that does not meet the required criteria.

Pass Condition:

System must provide an appropriate error message indicating the password does not meet requirements.

Password Recovery Tests

17. With a current connection between the client and server, attempt to recover a password by entering a username that is not registered in the system.

Pass Condition:

System must provide an appropriate error message, e.g “There are no accounts associated with the username entered. Please enter the correct username.”

18. With a current connection between the client and server, attempt to recover a password by entering a valid, registered username.

Pass Condition:

System must retrieve the user's email address, send the recovery email, and display an appropriate message, e.g “Password recovery email has been sent. Please follow recovery instructions.”

Shutdown Tests

19. With the client application running, select the option to shut down the program.

Pass Condition:

If the user is logged in, the system must log the user out, disconnect from the server, and fully exit the application successfully.

Server

Powerup Tests

1. With the server powered off, start the server program to initiate a cold power-up sequence.

Pass Condition:

Server must load configuration files, initialize the GUI, initialize the databases, open the server socket, begin listening for client connections, and update the GUI to display "Server Online".

Establish Client Connection Tests

2. With the server running and listening for connections, have a client attempt to connect using a valid IP address.

Pass Condition:

Server must accept the socket connection, assign a worker thread to the client session, store the session in memory, and update the GUI to display the connected client.

3. With the server running and listening for connections, simulate a client connection attempt during a server-side error such as max capacity reached or internal socket failure.

Pass Condition:

Server must refuse the connection, send a failure response to the client, and log the error in the GUI (e.g. "Connection refused due to server load or internal error").

Shutdown Tests

4. With the server active, select the "Shutdown Server" option from the admin GUI.

Pass Condition:

Server must stop accepting new connections, notify all connected clients, close active sockets, terminate worker threads, and close the GUI, fully shutting down the program.

Force Shutdown Tests

5. With the server running, simulate an abrupt termination of the server process (e.g. forced OS kill).

Pass Condition:

All sockets must be severed, connected clients must experience connection loss, and on the next server startup the system must return to a clean initialized state.

Query Tests

6. With the server active, click the “Query” button in the server GUI.

Pass Condition:

Server must retrieve total registered users, total logged-in users, locked-out users, and currently connected active sockets, and must display the updated results instantly in the server GUI.

Server-Side Logout Handling Tests

7. With a client connected and logged in, issue a LOGOUT request from the client.

Pass Condition:

Server must verify the session, update the logged-in list in the database, remove the user from the GUI display, update counters, and send the appropriate confirmation message back to the client (“Logout successful”).

Server-Side Registration Validation Tests

8. With a validated registration request sent from the client, process REGISTER(username, password, email) on the server.

Pass Condition:

Server must check for duplicate usernames, validate email format, validate password rules, and if all criteria are met, store the new user in the USER database and send a success response to the client. If invalid, server must return the appropriate error.

Server-Side Password Recovery Process Tests

9. With a RECOVER(username) request sent by the client, attempt password recovery on the server.

Pass Condition:

Server must check for username existence, retrieve the stored email if found, generate a recovery token or temporary password, send an email through SMTP, and return a success message to the client. If the username is not found, server must return the appropriate error.

Server Lockout Enforcement Tests

10. With multiple incorrect login attempts received from the client, allow the server to enforce the 3-attempt lockout rule.

Pass Condition:

On the third failed attempt, the server must mark the account as LOCKED in the database, send a lockout email to the user, update the GUI to reflect the locked-out status, and return an "Account locked" response to the client. Further attempts must be rejected.

Server-Side Password Change Validation Tests

11. With a valid CHANGE_PASSWORD(username, newPassword) request, attempt to update the password on the server.

Pass Condition:

For a valid new password that meets all requirements, the server must update the password in the database and return a success message ("Password updated").

12. With an invalid CHANGE_PASSWORD(username, newPassword) request where the password does not meet requirements, attempt to update the password on the server.

Pass Condition:

Server must reject the update request and return an appropriate error message indicating that the password does not meet requirements.