T25 Network Library: Manual Testing Script

ChatServer Unit Tests:

Test Suite Set Up:

→ Step 1: Open a terminal and navigate to the the base directory for the 'chat-client' plugin (relative to the project base directory, this directory is './trunk/plugin/').

→ Step 2: Run the command './export.sh' to perform a fresh installation of the 'chat-client' server.

→ Step 3: Start up an instance of Jenkins on the local machine.

Test Suite Tear Down:

→ Step 1: Terminate the locally running instance of Jenkins.

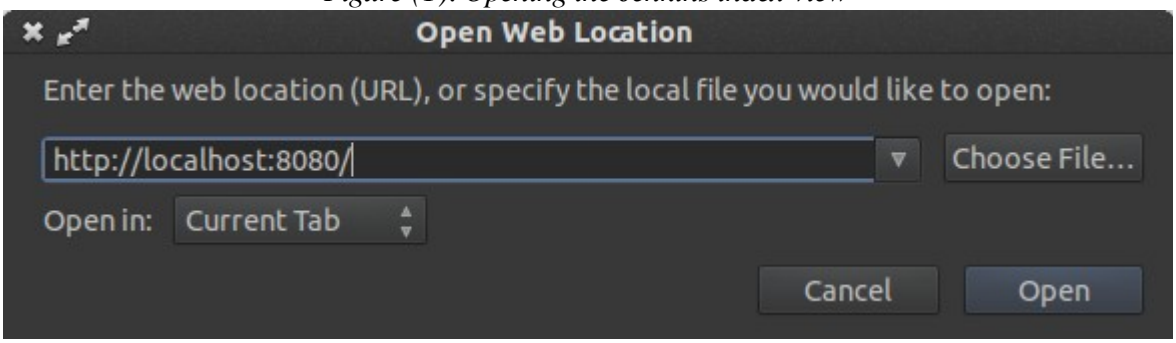
→ Step 2: Open a terminal and navigate to the the base directory for the 'chat-client' plugin (relative to the project base directory, this directory is './trunk/plugin/').

→ Step 3: Run the command 'mvn clean' to remove the installation of the 'chat-client' from the local host.

Unit Test Set Up:

→ Step 1: Open up the index page of the Jenkins server in a web browser (this index page will be at the address 'http://localhost:8080' for a default Jenkins configuration). Refer to figure (1) for reference.

Figure (1): Opening the Jenkins index view



Unit Test Tear Down:

→ Step 1: Close all opened Jenkins pages within the used web browser.

Unit Tests:

1. Test Basic Connectivity Functionality

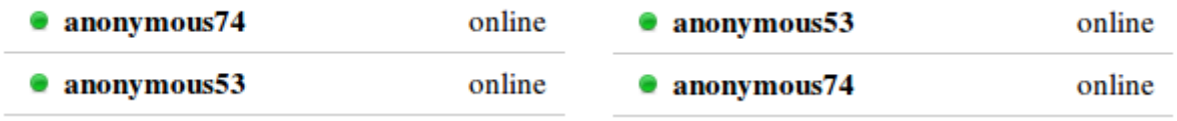
→ Step 1: Assert that the Jenkins index page contains a single tab at the bottom of the page. Refer to figure (2) for a visual representation of the tab (note that the contained text may differ).

→ Step 2: Ensure that the tab contains a green dot and text that reads 'online' within its borders.

Figure (2): Example connectivity tab

2. Test Multi-User Connectivity Functionality
   → Step 1: Run the entirety of unit test (1), performing all necessary assertions.
   → Step 2: Open up the index page of the Jenkins server in a different web browser from that used in step 1 (incognito mode for Firefox or Google Chrome can also be used as "different" web browsers). Refer to figure (1) for reference.
   → Step 3: From the page opened during step 1, assert that the page now contains two tabs at the bottom of the page: the top tab containing the same text from the tab observed earlier and the bottom containing different text.
   → Step 4: Ensure that both tabs contain a green dot and text that reads 'online' within their borders. See figure (2) for reference.
   → Step 5: From the page opened during step 2, assert that the page contains two tabs at the bottom of the page with the top-to-bottom ordering of the tabs being opposite to the ordering in the first page. See figure (3) for a visual representation.

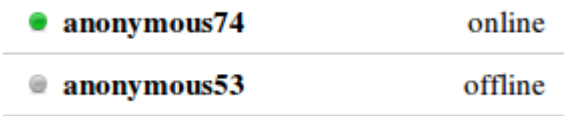
Figure (3): Example of switched ordering for user tabs



   → Step 6: Ensure that both tabs contain a green dot and text that reads 'online' within their borders. See figure (2) for reference.

3. Test Disconnect Functionality
   → Step 1: Run the entirety of unit test (2), performing all necessary assertions.
   → Step 2: Close the second window opened during the execution of the prerequisite test.
   → Step 3: Observe the tabs at the bottom of the screen in the remaining window. Ensure that there are still exactly two tabs at the bottom of the screen.
   → Step 4: Observe the bottom tab closely and assert that the text within its contained borders has changed to read 'offline' and that the green dot within the tab has turned to gray (see figure (4) for a visual aid).

Figure (4): Example of user tabs after a disconnect



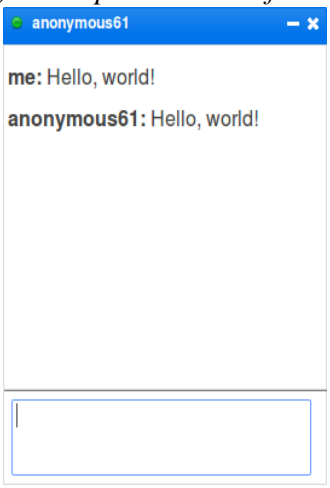
4. Test Continuity of User Identifier
   → Step 1: Upon opening the Jenkins index page, record the name contained within the single tab at the bottom of the screen (see figure (1) for a visual on the tab).
   → Step 2: Reload the Jenkins index page.
   → Step 3: Verify that the name in the single tab at the bottom of the page is identical to the value recorded in step 1.

5. Test Basic Message Sending Functionality
   → Step 1: Left click on the single tab at the bottom of the page.
   → Step 2: Verify that a chat box appears in the bottom-right corner of the web page similar to the chat box displayed in figure (5).
   → Step 3: Left click on the message space within the chat box located in the previous step. This message space is highlighted in gray in figure (5).

Figure (5): Example chat box before messaging

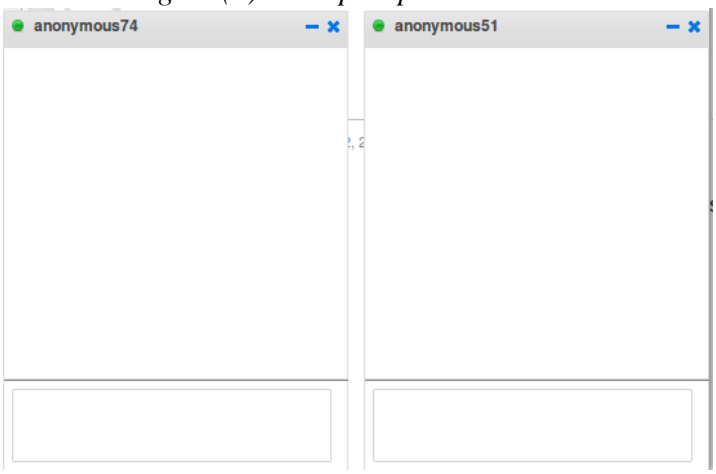
   → Step 4: Enter the text "Hello, world!" into the message space and press the [Enter] key.
   → Step 5: Verify that the previously entered text appears twice in the window: once with the prefix "me:" and another time with the prefix "name" (where "name" corresponds to the text contained in the top bar of the tab window). Refer to figure (6) for a visualization of the desired behavior.

Figure (6): Example chat box after messaging



6. Test Multi-User Message Sending Functionality
   → Step 1: Run the entirety of unit test (2), performing all necessary assertions.
   → Step 2: From the first view, left click on both of the displayed tabs at the bottom of the page in top-to-bottom order.
   → Step 3: From the first view, left click on both of the displayed tabs at the bottom of the page in bottom-to-top order.
   → Step 4: Verify that two chat boxes have appeared within each view (see figure (7) for a visual).

Figure (7): Multiple open chat boxes



→ Step 5: From the first view, left click on the gray area in the bottom of the rightmost chat box and type the message “View 1” followed by the [Enter] key.
→ Step 6: Assert that only the chat box into which text was entered was updated and that the updated visual for this chat box is similar to figure (6) except with the phrase “View 1”.
→ Step 7: From the second view, left click on the gray area in the bottom of the rightmost chat box and type the message “View 2” followed by the [Enter] key.
→ Step 8: Assert that the chat box into which this text was typed was updated with one entry of the form “me: View 2”.
→ Step 9: Within the first view, ensure that the leftmost chat box contains a single entry of the form “name: View 2” (where 'name' corresponds to the string labeling the chat box).
→ Step 10: Finally, assert that all unchecked chat boxed up to this point do not contain any entries.