

# Linux Chat Application

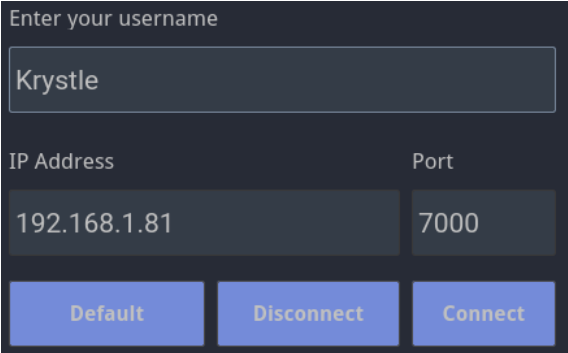
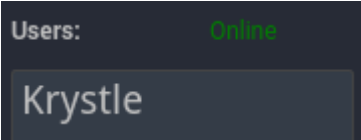
Oscar Kwan  
Krystle Bulalakaw

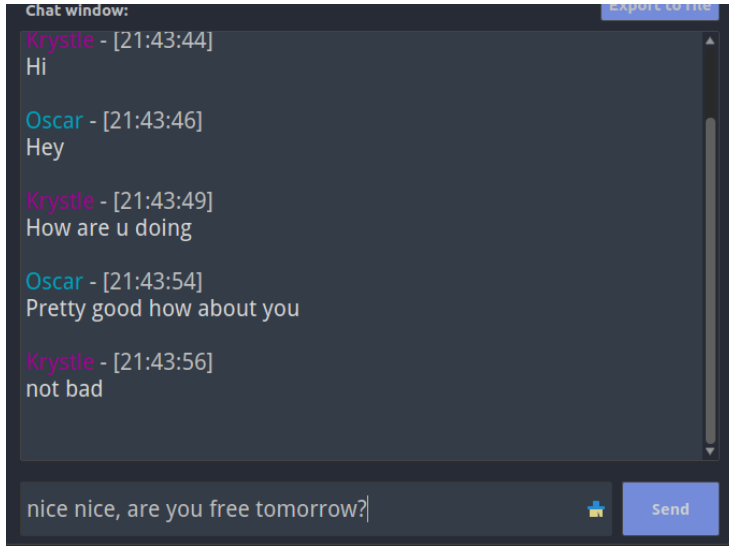
## Test Document

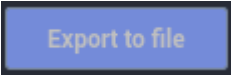
March 23, 2016

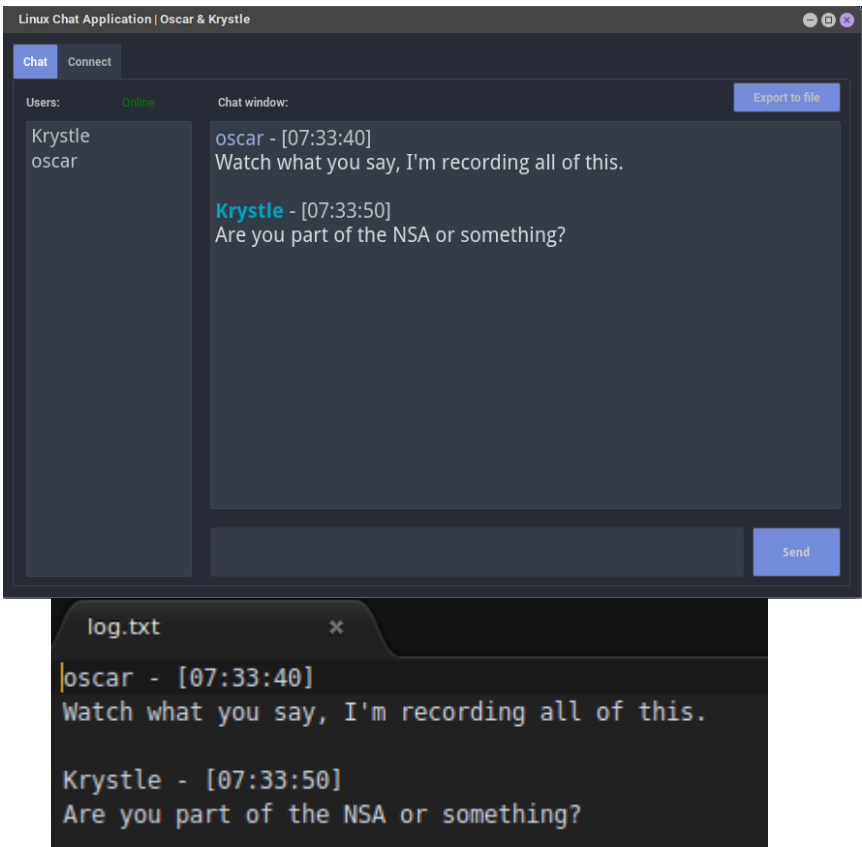
### Test Cases

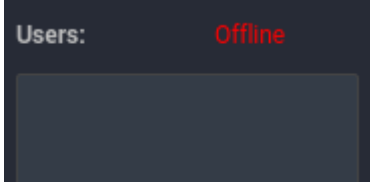
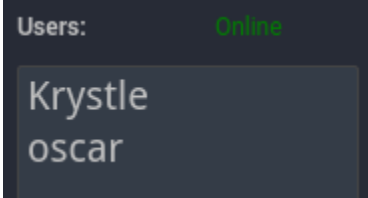
Test ID	Test Description	Test Steps	Expected Results	Pass/Fail
1	Server compiles and runs without and warnings or errors.	<ol style="list-style-type: none"><li>1. Navigate to the folder containing <i>server.cpp</i></li><li>2. Type <i>gcc -Wall -o server server.cpp</i></li><li>3. Type <i>./server</i></li><li>4. Observe results</li></ol> <pre>867 krystle:~/.../Server (qthreading)\$ ./server Listening for connections on port 7000...</pre>	No warnings or errors are produced. A print statement indicates the server is listening.	Pass

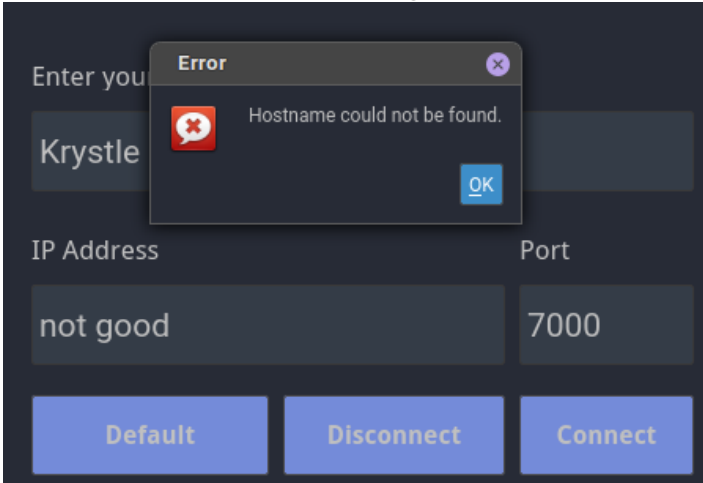
2	Client connects to server successfully.	<ol style="list-style-type: none"> <li>1. Launch server application</li> <li>2. Launch client application</li> <li>3. Navigate to the connect tab</li> <li>4. Enter in appropriate fields for username, ip address, and port.</li> </ol>  <ol style="list-style-type: none"> <li>5. Press connect button after these fields are entered</li> </ol>  <pre data-bbox="696 1102 1525 1353">867 krystle:~/.../Server (qthreading)\$ ./server Listening for connections on port 7000... Updated client list: fd[4] IP[192.168.1.81] Received from client 192.168.1.81: Krystlethisisusernameguud Got user name Updated username list: fd[4] name[Krystle] Sent to client 192.168.1.81: Krystlethisisusernameguud</pre>	<p>The server reports a connection of a client along with its name and IP address.</p> <p>The client indicates the user is online and on the list of connected clients.</p>	Pass
---	---	--	---	------

3	Client receives messages successfully.	<ol style="list-style-type: none"> <li>1. Launch server application</li> <li>2. Launch multiple server applications on different hosts</li> <li>3. Repeat Test #2 to connect all clients</li> <li>4. Go to Connect tab, enter text in chat box and press &lt;Enter&gt;</li> </ol> 	<p>Server indicates client messages sender and contents.</p> <p>Each client receives messages sent from another, along with name and timestamp.</p>	Pass
4.	Timestamps displayed correctly in window.	<ol style="list-style-type: none"> <li>1. Observe chat log from Test ID #3.</li> </ol>	Timestamp is displayed on the right of name	Pass

			whenever user sends or receives a message.	
5	Client logs chat session to file if pressed.	<ol style="list-style-type: none"> <li>1. Launch server application</li> <li>2. Launch client application</li> <li>3. Go to Connect tab</li> <li>4. Enter appropriate values for name, IP, and host and click Connect</li> <li>5. Go to Chat tab</li> <li>6. Send and receive text to other users (the chat is logged only if there is some history)</li> <li>7. Click Export to file  </li> <li>8. The log file can be found in <i>linux-chat/Source/build-linuxchat-Desktop_Qt_5_5_1_GCC_64bit-Profile/log.txt</i></li> </ol>	The contents of log.txt should match the chat history from the beginning up to the point where <b>Export to file</b> was clicked.	Pass

				
6	User can disconnect.	<ol style="list-style-type: none"> <li>1. Launch a server and client application and connect as normal.</li> <li>2. Enter some test text to ensure you are connected and receiving from the server.</li> <li>3. Navigate to Connect tab and click <b>Disconnect</b></li> <li>4. Observe terminal and Chat tab.</li> </ol>	The server should report that a disconnect message was received, and that it disconnected the	Pass

		<pre>Received from client 127.0.0.1: closesd Got disconnect message Updated client list: fd[5] IP[127.0.0.1] Updated username list: fd[5] name[oscar] Client 127.0.0.1 disconnected.</pre> 	<p>client and updated the client/username list.</p> <p>The client should appear as offline and not see any other online users.</p>	
7	User can reconnect after disconnecting.	<ol style="list-style-type: none"> <li>1. Repeat Test #6.</li> <li>2. Click the <b>Connect</b> button once more.</li> <li>3. Observe terminal and chat tab.</li> </ol> <pre>Updated client list: fd[4] IP[192.168.1.81] fd[5] IP[127.0.0.1] Received from client 192.168.1.81: Krystlethisisusernameguud Got user name Updated username list: fd[4] name[Krystle] fd[5] name[oscar] Sent to client 192.168.1.81: Krystlethisisusernameguud</pre> 	<p>The server should report that a client connected, and it updated the client/username list.</p> <p>The client should appear as online and see the list of connected users.</p>	Pass

8	Attempting to connect with Invalid parameters results in an error message box.	<ol style="list-style-type: none"><li>1. Launch server and client application</li><li>2. Attempt to connect with invalid parameters:<ol style="list-style-type: none"><li>a. Blank</li><li>b. Invalid host</li><li>c. Invalid IP</li><li>d. Server not running</li></ol></li></ol> 	The client should be indicated with an error message that it was not able to connect due to socket error, host resolution error, etc.	Pass
---	--	---	---	------