

# Testing Document

**Authors:**

**May 5, 2019.**

**Group 6:**

- 1.Nishad Aherrao**
- 2.Alexander Windeler**
- 3.Bishesh Shrestha**
- 4.Khom Sitaula**
- 5.Shivangi**

# 1. Requirements Table

Identifier	Requirement	N/F	Source	C/S/B	Notes	Class/Method
1	Messages delivered in less than 1 second from being sent	N	Document	C		
2	Maximum of 50 users online at one time and an admin.	F	Document	S	Admin doesn't count in the 50 users.	Server class, determined by size of list_of_client vector
3	SuperChat runs on the linux operating system.	N	Document	B		
4	SuperChat will be implemented using -std=c++11	N	Document	S		
5	User can delete empty chat rooms	F	Document	C		Server class, delete_chatroom() method
7	User provides nickname to be used in chat rooms, can be changed every log in	F	Document	C		Client class, Create_client() method, name attribute
8	SuperChat must have two applications, The "Client" and the "Server".	N	Document	B		
9	The Client and the Server can run on the same or on different computers.	N	Document	B		
10	User can place others on personal ignore lists to stop seeing messages from them	F	Document	C		Client class, ignore_user() method
11	User can use multiple clients at once, as long as they have different nicknames	N	Document	C		
12	The user interface for the Client will use the ncurses.	N	Document	C		

13	The server will provide a user that join a chatroom all of the previous messages.	F	Document	S		Chatroom class, display_previous() method
14	Chat room will record all messages sent until deleted.	F	Document	S		Chatroom class, record_history() method
16	Ten chat rooms can exist at one time, including the lobby	N	Document	S		
16	Messages will be spell checked against a file provided by the instructor.	F	Document	C		Spell_check class, spell_check() method
17	The dictionary will have one word per line, and will not be in any order.	N	Document	C		
18	Messages sent will include the nickname of the person sending and a time stamp	F	Self Decided	S		Client class, Get_nickname() will get the nickname of the user and will add it to the front of the text.
19	'Tab completion' will be used on the client. Suggest the gnu readline library be used..	F	Document	S		Spell_check class, tab_complete() method
20	Persistent information stored across client invocations will be stored in a file named "~.SuperChat".	F	Document	S		Server class, Save_state()
21	Files can be transferred between users between their home directories.	F	Document	C		Message class, encode()
22	There is no maximum for users per chatroom as long as there are 50 or less users online at one time.	F	Self Decided	S		Server class, check_name() will validate the entry of the client
23	"Message obfuscation" will be supported. An integer, known to clients, will be used to obscure the meaning of the message for clients that don't know the number. (technique must be very simple!!)	F	Document	S		Message Class, encrypt(),decrypt().

24	There is a default chat room called the 'lobby'. It can not be deleted.	F	Document	S		Server class, create_chatroom() method
25	The client and server must function without failure for at least 30 minutes.	N	Document	B		
26	When SuperChat runs. It will bring user to a login page where user can enter nickname and will be logged in if there aren't 50 users online already. If there are 50 users online then it will pop an error message saying that SuperChat cannot accept any more users at the moment.	F	Self decided	S		Server class, check_name()
27	When SuperChat runs. It will bring user to a login page where user can enter nickname and will be logged in if there aren't 50 users online already. If there are 50 users online then it will pop an error message saying that SuperChat cannot accept any more users at the moment.	F	Self decided	C		Client class, create_client()
28	There will be a button the create a new chatroom	F	Self decided	C		Server, create_chatroom; Client, delete_chatroom;
29	There will be a button to delete and empty chatroom	F	Self Decided	C		Client, delete_chatroom;
30	There will be a leave button that removes a user from a chatroom and places them back in the lobby	F	Self Decided	C		Client, Leave()
31	There will be a logout button that disconnects the user from the server	F	Self Decided	C		Client, Logout()
32	Replies that are repeated 50 times (common replies) or more will be saved and used to help users 'auto reply'.	F	Document	S		spell_check class, auto_reply()

33	Users can select a secret number to encrypt their messages rather than a pre decided encryption to add more security.	F	Self-Decided	C		Client, highlight(),ncurses
34	Users can use buttons to navigate in between different options rather than using dull commands.	F	Self-Decided	C		Client, highlight(),ncurses
35	Decrypted messages stay on the screen for future reference	F	Self-Decided	C		Client, highlight(),ncurses

## 2. Test Case:

Step	Instruction	Expected Result	Actual Result
1	Prompt user to enter a nickname	Shows a window asking for user name "Nickname"	Shows a window asking for user name "Nickname"
2	Check if username already exists	If the username already exist shows "Error! Name taken"	Works for few users
3	Check if client is banned	If a client is banned then show "Error! You are banned"	Not working
4	Take user to lobby	Displays lobby screen with LOBBY title and messages if any	Displays lobby screen with LOBBY title and messages if any
5	User should be able to send messages	Displays send and received messages to all users	Displays send and received messages to all users
6	User's name should be printed followed by a time stamp	Nickname Hi[12:12...] Should be the format in which sent/received messages are displayed	Nickname Hi[12:12..] is the format in which sent/received messages are displayed
7	If a new user joins, all the messages should appear on the user's screen	All previous messages are loaded on the newly joined user	All previous messages are loaded on the newly joined user

8	All the functionalities should have a button for itself	Create chatroom, encrypt, decrypt, attachment, exit buttons to operate those commands	Create chatroom, encrypt, decrypt, attachment, exit buttons to operate those commands
9	User should be able to change nickname without logging out	Pressing nickname in the nickname box opens up a window and asks user to enter a new nickname	Pressing nickname opens up a window and asks user to enter a new nickname
10	Change in nickname should reflect on all windows	New Nickname is the nickname in the small box above send and send messages to all the clients with the new nickname	New Nickname is the nickname in the small box above send and send messages to all the clients with the new nickname
11	Create chatroom button should prompt the user for the name of the chatroom	Pressing create chatroom opens up a window to ask user for a chatroom and creates a chatroom	Pressing create chatroom opens up a window to ask user for a chatroom and creates a chatroom
12	It should check to see if that chatroom doesn't exceed the restriction of only 10 chatrooms available	Gives user an "Error! Too many lobbies"	Gives user an "Error! Too many lobbies"
13	Texting should work normally when created a new chatroom	When user send messages in new chatroom other user in that chatroom can see the messages	It's not showing the new messages to the other users
14	User should be in lobby when/if leaves another chatroom	User is in lobby after leaving a chatroom	Leave button is not functioning
15	All the messages from lobby should be loaded on the users screen once they leave a specific chatroom	Previous lobby messages are loaded on user's window	Leave exists user out of Superchat
16	User should be able to exit Superchat with exit button	Pressing exit exits out of Superchat	Pressing exit exits out of Superchat
17	Encrypt Message should encrypt the messages	Pressing the encrypt button pops up a window asking for an integer and encrypts all the messages there after	Pressing the encrypt button pops up a window asking for an integer and encrypts all the messages there after
18	Decrypt message should decrypt the	Pressing the decrypt button	Pressing the decrypt button

	messges	pops up a window asking for an integer to decrypt the message. If provided correct value, decrypts the messages	pops up a window asking for an integer to decrypt the message. If provided correct value, decrypts the messages after inputting the message
19	Spell check should provide suggestions if a word is spelled or half-typed	A small box above the send/receive constantly shows the correct/ed spelling	A small box above the send/receive constantly shows the correct/ed spelling
21	Attachment should allow user to share a file	Pressing attachment would open a window asking user to enter a filename and will send the file path	Pressing attachment would open a window asking user to enter a filename and will send the file path
22	Entire code is implented in c++	Code is written in c++	Code is written in c++
23	Messages are sent in less than 1 second	Time between one user sends a message and other receives is less than 1 second	Time between one user sends a message and other receives is less than 1 second
24	Program should run on linux operating system	It runs on Virtual Machine	It runs on Virtual Machine
25	Arrow keys navigate through the windows	Pressing arrow keys highlights different buttons	Pressing arrow keys highlights different buttons
26	User can use multiple clients at once as far as they have different nicknames	Allows user to login through multiple tabs	Allows user to login through multiple tabs
27	Client and server function without failing for 30 minutes	Client and server ran for 30+ minutes	Client and server ran for 30+ minutes
28	Client and server should be able to run on the same computer	Client and server runs on the same computer	Client and server runs on the same computer
29	Ncurses window should refresh and be consistent at reflecting changes	Everytime when changes are made, Ncurses refreshes and reflects them	Everytime when changes are made, Ncurses refreshes and reflects them
30	Send/ Receive button	Pressing send/receive button enable user to send messages	Pressing send/receive button enable user to send messages

31	User should be able to get out of the texting box	Pressing ! or ~ bring user on send/receive button and allows them to navigate through different buttons.	Pressing ! or ~ bring user on send/receive button and allows them to navigate through different buttons.
32	Tab completion	When user hits tab, it shows a list of suggestion of words	Not implemented
33	Delete chatroom should check to see if there isn't any user in the chatroom	If there is no user in the chatroom, it deletes the chatroom and says, "Chatroom deleted" else says "Error! Cannot delete the chatroom"	If there is no user in the chatroom, it deletes the chatroom and says, "Chatroom deleted" else says "Error! Cannot delete the chatroom"
34	Delete chatroom shouldn't delete lobby	If a user tries to delete lobby it pops up an error, "Error! Can't delete the lobby!"	If a user tries to delete lobby it pops up an error, "Error! Can't delete the lobby!"
35	HELP command	./help command provides user with instructions and helpful information to work with super chat	Works on terminal side, not connected with ncurses.
36	Chatroom lists	There would be a box on the right hand side of the screen which shows the list of available chatrooms	Not implemented

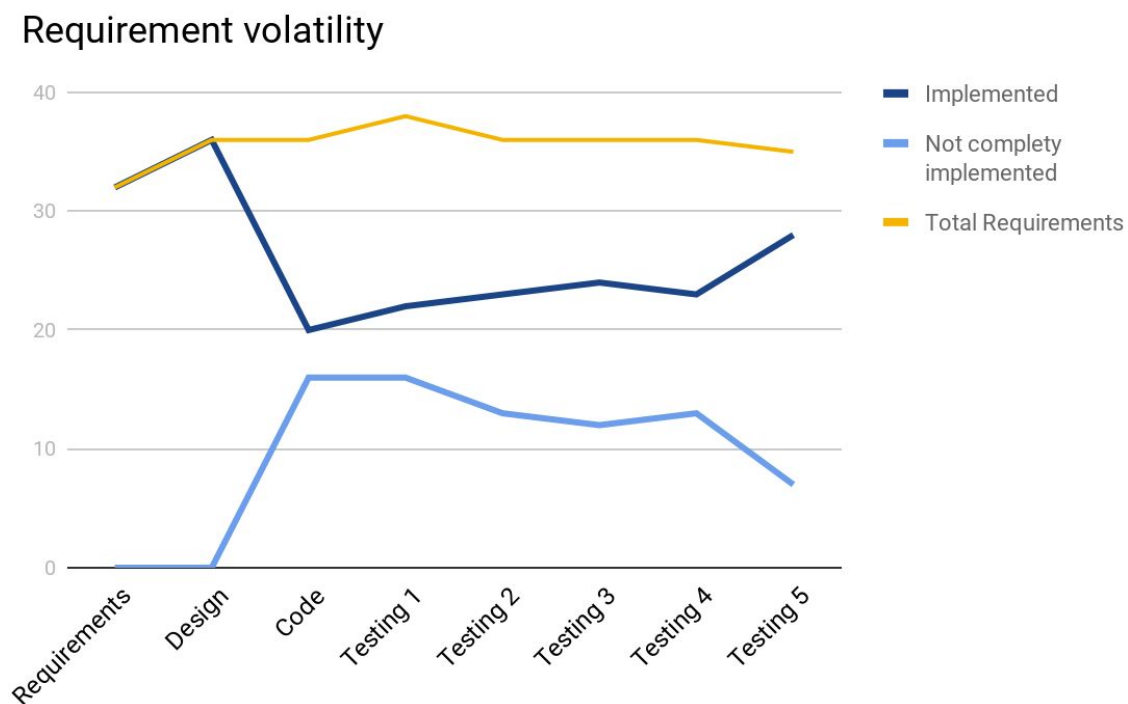
## 4.List of Defects:

1. Banning the client is not working.
2. While entering a nickname, it doesn't check to see if the nickname is banned.
3. Messaging doesn't work well in chatrooms other than lobby.
4. "Leave chatroom" functionality is not implemented
5. Tab completion is not implemented correctly.
6. List of chatrooms is not available
7. User can create chatrooms but no two user can be in the same chatroom other than the lobby.
8. Help command not connected with ncurses.



## 5. Summary:

Testing is an integral part of software development. Our group started testing few weeks back when we believed that we have the most basic functionalities down. We have implemented various stages of testing which has also affected our requirements chart as shown below:



As you can see from the chart, during testing we dropped some requirements as we figured that it would be nearly impossible for us to meet those while we also picked up on some small requirements. As we progressed with unit testing, we discovered few defects and fixed them and also focused on small blocks of functions and tried to make sure that those small parts of the code works well. Once we achieved success with small functions we started integrated it all together and went to the next stage of testing. The idea of making things work on the server end and client end individually and then integrate was really helpful. After doing small scale testing, getting in the week of presentation, we started working on system testing. We integrated

the front-end and back-end and thoroughly tested the system for defects. While doing so we encountered some issues because of nurses behavior which we fixed. We kept on testing our code while adding new features.

Our final test on the final code resulted in following results:

After running 36 test cases on our software for four to five times and trying to fix some basic and fixable defects, our code satisfies 28 requirements without any defects. There are 7 requirements which are not completely implemented and has some defects. Those 7 requirements can be broken down to 8 defects.

In conclusion, our group started with 32 requirements which then went up to 36 out of which 28 has been implemented without any defects. Our group believes that it is a great success considering various factors such as time constraints, lack of previous knowledge about libraries etc... and how we were still able to make a chatroom with an amazing Graphical User Interface with the functionalities that it implements. Even though multiple chatrooms feature is not implemented, our version of SuperChat is of high quality, with spell checks for user input and buttons for navigation.