COMP90015 – Distributed Computing

GROUP ASSIGNMENT, SEMESTER 2 2017

Group member:

Shuai Wang	ID: 830166	Email: shuaiw6@ student.unimelb.edu.au
Peng Cheng	ID: 880420	Email:P.cheng5@student.unimelb.edu.au
Taoji Feng	ID: 880400	Emial:Taojif@student.unimelb.edu.au
Luyang Yang	ID:880199	Email:Luyang@student.unimelb.edu.au

1. Introduction

The scope of this project is to develop a shared whiteboard system with a chat window. The system should be a client-server architecture system. One server should be able to allow different users to drawing on the same white board at the same time, and the chatting function is also achieved by this server. After editing on this picture, the user is able to do some other operations on this white board, such as save, save as and more.

2. System architecture

The system should be a client-server architecture system. This system contains two components, one is client side and another is server side. The client is designed with GUI, and users are able to do any actions on the client side directly. The server side is used to handle the message received from client, and also give reply message to client.

3. Implementation details

Client details

The client is designed with a GUI, and this GUI contains three main parts. The first part is the function part. It is located at the top of this system. In this part, there is three drop down lists. The first one is called File. Second is called Connect, and the third one is called peerlist. The File one contains new, open, save, save as and exit function which is used for users do different operation on the existing white board. To be more specific, the new button is used for users to open a new clean drawing board when they want to make a new drawing board to draw a new picture. The open button is used to allow users to open a local existing drawing board file to do more modification. The save button is used to save the opening drawing board to local device and the user could select the storage path of this file. The file type is special, and it only can be opened by our software. The save as button is designed for users to save a picture format file for the opening drawing board, and this file could be opened by any image viewing tools. Such as windows image viewer and more. The exit button is used to exit the system. The Connect dropdown list is used for connection establish and close. There are two buttons under Connect, one is join, and another is leave. The join is used to connect to the running drawing board with a specific username port number and IP address. The leave button is used to log out from the running board. Users are able to join the running drawing board any time with the manager’s permission. He could also leave

this drawing broad any time. The peerlist is used to show who is in this whiteboard now. When the user click this button, there will be a new dialog shows the current users on this white board. This list would be updated when there is any change to the current user list. This is also used to help manager to kick users (Kick would be explained more specifically in the chatting server details part). Under the drop down list, there is a column of buttons which is used for drawing. These drawing buttons is used to help the user to draw some common shapes. Such as circle, line, rectangle and more, there is also an eraser button used for cleaning. Except these drawing buttons, there is also a button used to add text on the drawing board. The user could add any comment to this drawing board, and other users will receive a message shows he left this board. The second part is the drawing area which is under the tools part. The draw area is used for user to draw different shapes and add comment. The last part is the chatting part. It is used to sending message to chatting server and also show message from others. This chatting channel is broadcast. Every user who is allowed to join this draw board is able to view all messages.

Chat server details

The chat server is used to deal with the chatting function. This server has a simple GUI which is used to help users to manage the server. To be more specific, the GUI contains two main areas. One area is used to show some basic information about this chatting channel and provide basic operation buttons for this server. The basic information is the port number of this chatting channel, and the basic operations are start and stop. Another one is used to show some current situation information about this chatting channel. For example, the users are able to view the user list of this chatting channel, and also some notice about new users' connection and leave.

The chat server is multi-thread server. The architecture of this multi-thread server is a thread per user. When a new user connects to the server, the server would build a new thread for this user. The new thread is used to deal with every request from this user. For example, when the user click send button to send a message to others, the server would use the already built thread to deal with this new socket and transfer this socket to other clients. Other clients would receive this socket and use their own thread to analyze it.

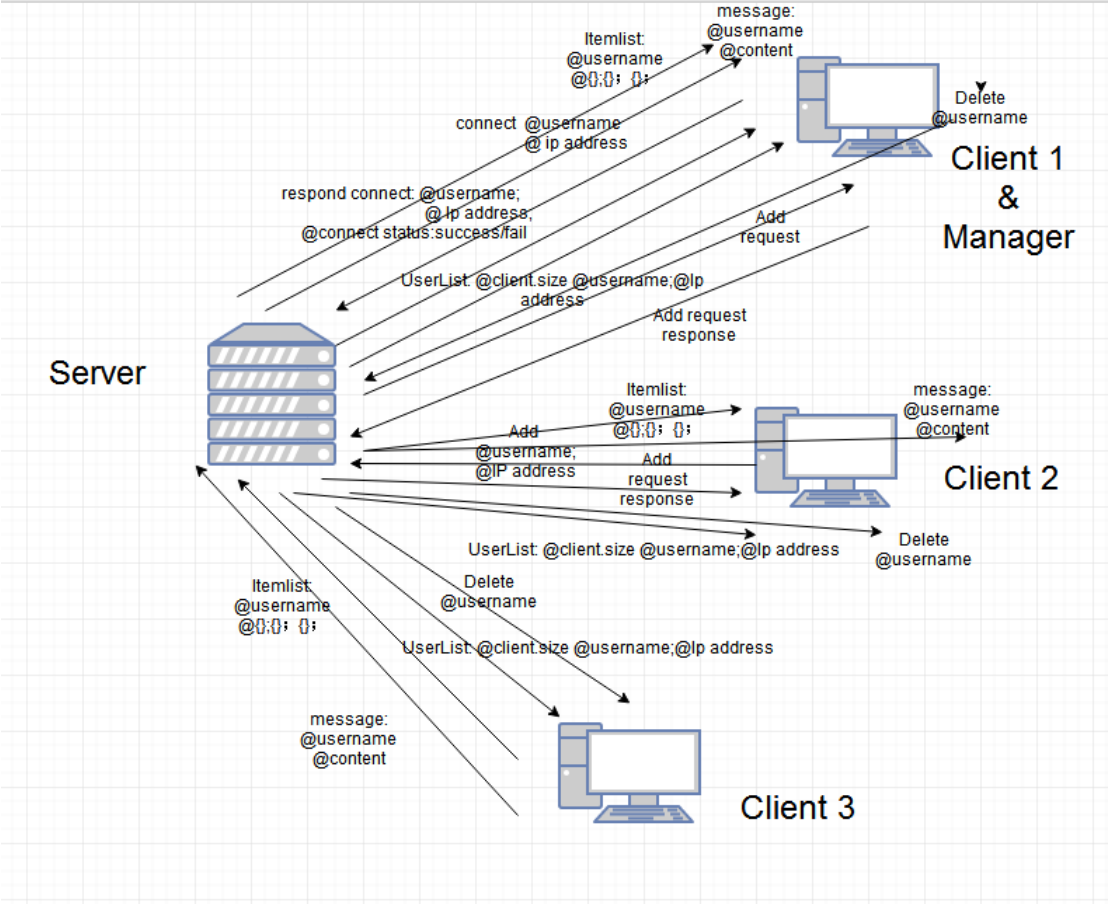
Except the basic communication function requirements, the chat server is also able to deal with authority problem. To be more specific, the authority is about the manager permission. In this system, only the manager is able to do the open, new, save, save as and exit function. When the manager use exit function, the server would receive a message shows the manager of this white board is left and the server would also be shut down. All other users are not able to use any function anymore. The manager also has right to allow other people to join this chat board or not. When the client asks for joining, the server would send a message to manager. If manager's reply message is accept, the new user would be added into this white board. Otherwise, the new user would be declined. Lastly, the manager is also able to kick users from this white board. Manager could input the username of existing users, and click kick button to kick this user from this white board. The kicked user also will receive a message shows he has been kicked.

Drawing server details

Drawing server is used to synchronize different drawing operation to different users' drawing board. To be more specific, each time when the user updates their drawing board, other users' drawing board would be update automatically at the same time. Therefore, every user's drawing board would be able to keep synchronized. For example, if one user draws a new rectangle on the opening drawing board, this rectangle would be synchronized to other users' drawing board automatically, and the properties of this rectangle such as location, size, color and more would also keep same. The synchronization process is achieved by update the drawing item list. Every time when the user updates their drawing board, there would be a socket sent from client side to server side. The socket contains the properties of this new updating, such as location color and more. These properties would be built as an item list included in the socket. The server would read this socket and sent this socket to other clients. Lastly, other clients would read this socket and repaint their drawing board based on the socket's information.

4. Communication protocols

The connection way between server and client is achieved by socket, the message format including Json and hash map, and the protocol we chose is the TCP, because it is a reliable protocol. For each client has multiple threads some are used for chatting and some are used to do the drawing actions. The interaction diagram between server and client is as shows below: the left side is server and exchanging data with multiple clients. Client 1 as the first client who is also a manager that has the privilege to approve the new client add in and kick out the exist user.



5. Class UML diagrams

There are two UML files in our project file. You can see them when you imported our two project files in Eclipse.

6. Conclusion

The project implements a share whiteboard system that allows multiple users use various drawing tools such as line, circle, rectangle and oval, to draw the picture on a shared interactive canvas. Since it's a share whiteboard which means all the users can see the same image of the whiteboard and have the privilege of doing all the drawing operations. Therefore, the process of clients drawing is synchronous. The system use TCP/IP protocol and socket technology for communication, and a multiple-thread server chat server as part of this system is also implemented, which meets the demand of user chatting online while using whiteboard.

7. Appendix

The project contribution of each member:

	Shuai Wang	Luyang Yang	Peng Cheng	Taoji Feng
Contribution	25%	25%	25%	25%
Work	1. Working on the drawing function 2. Working combination of different people's program 3. Working on final software testing	1. Working on the file class function 2. Working on final software testing 3. Working on documentation and requirements double check	1. Working on chatting function 2. Working on combination of different people's program 3. Working on final software testing	1. Working on manager authority settings. 2. Working on documentation 3. Working on combination of different people's program