Project Design

============================

Our group decided to split up the classes into two distinct groups: a Server and a Client.

Server:

1. Chat server: The purpose of this class is to create the server portion of the program. The class will also be responsible for keeping a list of the users currently signed into the server, the list of chat room that are currently in on the server, and it will keep track of the messages that are sent between users. We will achieve all of these using vectors to keep track and delete the various lists as they are added to and deleted. The server will also be responsible for creating and removing chat rooms, starting and ending the server, and broadcasting messages.
2. Chat room: The purpose of this class is to implement the specific functionality of the chat rooms. Each chat room object will have a string for the name, a vector for the list of ID’s of the users currently connected to the specific chat room, and the message list within the specific chat room. The class will be responsible for having users join and leave the chatroom, searching for users in the chatroom, and for keeping track of the messages sent.
3. Chat user: The purpose of this class is to be able to hold the information of each individual user so that the server will be able to use it rather than just the client. Each object will have a string for the name and an integer for the ID. The class will have the functionality of joining and leaving a server and joining and leaving the chat room. The rest of the functionality such as sending messages will be done on the client side as to keep the two classes separate.
4. Chat session: The purpose of this class is to handle the actual exchange of messages between two clients on the server. This will be done using various ncurses and asio functions like *read\_line()*  and such. This will be the only instance where the actual exchange of messages will occur to avoid any redundant functionality.

Client:

1. User: The purpose of this class is to be the parent class of both types of users (super user and participant.) This will be a virtual class that will have most of the functionality of the other two classes that will be passed down via inheritance. Both types of users will have a string for their name, integer for their ID number, and a vector of strings for their chat history. Each of the classes will also have the functionality of muting other users, changing and making rooms, sending messages and returning the messages that were sent by that user, and to broadcast a message to the server.
2. Super user: The purpose of this class is to set up the functionality and privileges of the super user class. The only variable that is unique to this class is the *sup\_priv* Boolean value that will be set to true. It will also have one additional function to delete a chat room which is a privilege unique to super users.
3. Participant: The purpose of this class is to set up the functionality and privileges of the participant class. he only variable that is unique to this class is the *sup\_priv* Boolean value that will be set to false. It will have no additional functionality.