**Assignment Breakdown**

**Part 1 chat room and general logic:**

The chat room will be quite simple to do because the code was already pretty much given in class. For the features that were not provided, I will create function for them. For the canvas portion that compares the two colorsAs for how my chat will display, I will create multiple divs. One containing the username prompt, one the main chat, and one for the private chat. I will then hide or show these divs depending on the context.

My rough sketch for my chat room design will be this:Diagram, text

Description automatically generated

. I will use the user Ids to keep track of the different users. The private messages will still be on the chat file however, the text will show up depending on which screen you are currently on (if a user enters a private message, and the one on the receiving end is in a private chat, he will receive the message in the private chat with a label saying that it came from the main chat.) To differentiate the private from the main chat, I will use a dark background so that it is different from the main chat’s white background. For the feel of my website, I’ve decided on a pink background, but not too neon so it stays professional. I think I will also add some light blue in there. For the 3 Final 5, I’ve already decided on keeping track of game, having all users display and the user having a avatar.

**Part 2 Game:**

My game will be pretty simple. Within my chat room file, I will have a function that will check for the different commands. If the !play username command has been made, a response will be sent. This response will use the once event listener so that only one person can join. When someone joins, a div will appear on the left side of the screen containing the game (might look better at the bottom, we’ll see). The game will work like a normal rock paper scissors game and the functions to handle that will be made in the game file. Once someone leaves, the div should disappear from the screen and the game will disconnect. I think I will also make a game object that keeps track of the amount of times a user as won or loss and the conditions to win or loose.

**Part 3 Private chat room:**

This will be a bit more complicated but to implement it, I think I will use the Id of the user, and by this id, invite them into the room. I will also reference the guide you gave us on Moodle. For the private rooms, I will create an object called private room, saving it’s number and the users that are inside. For this class, I will create methods that will be able to remove a user from the private room. As for how to send a request. I have decided to utilize a !invite command which invites a existing user to the room. For how the user accepts a request, I have decided that the user enters a command which would be !accept (type of request) (user that sent the invite). There is a lot of error checking to do on this one so I’ll have to be careful (room stops when only one person is in room, cannot invite same person two times etc.)

**Part 4 logging to file:**

This will be pretty easy, I will create methods on the server file that are able to create a file if one does not exist and write to it the events of the chat, game and private chat.