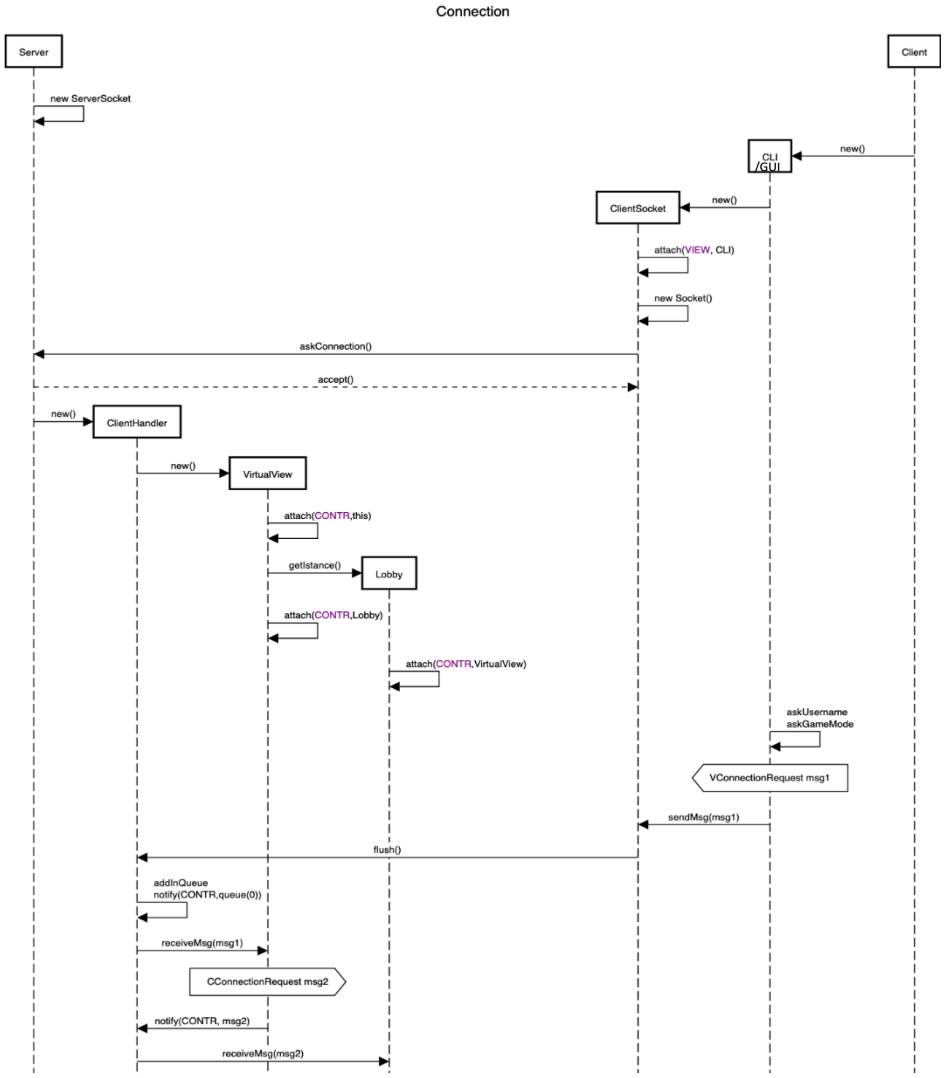
CONNECTION AND ACCESS

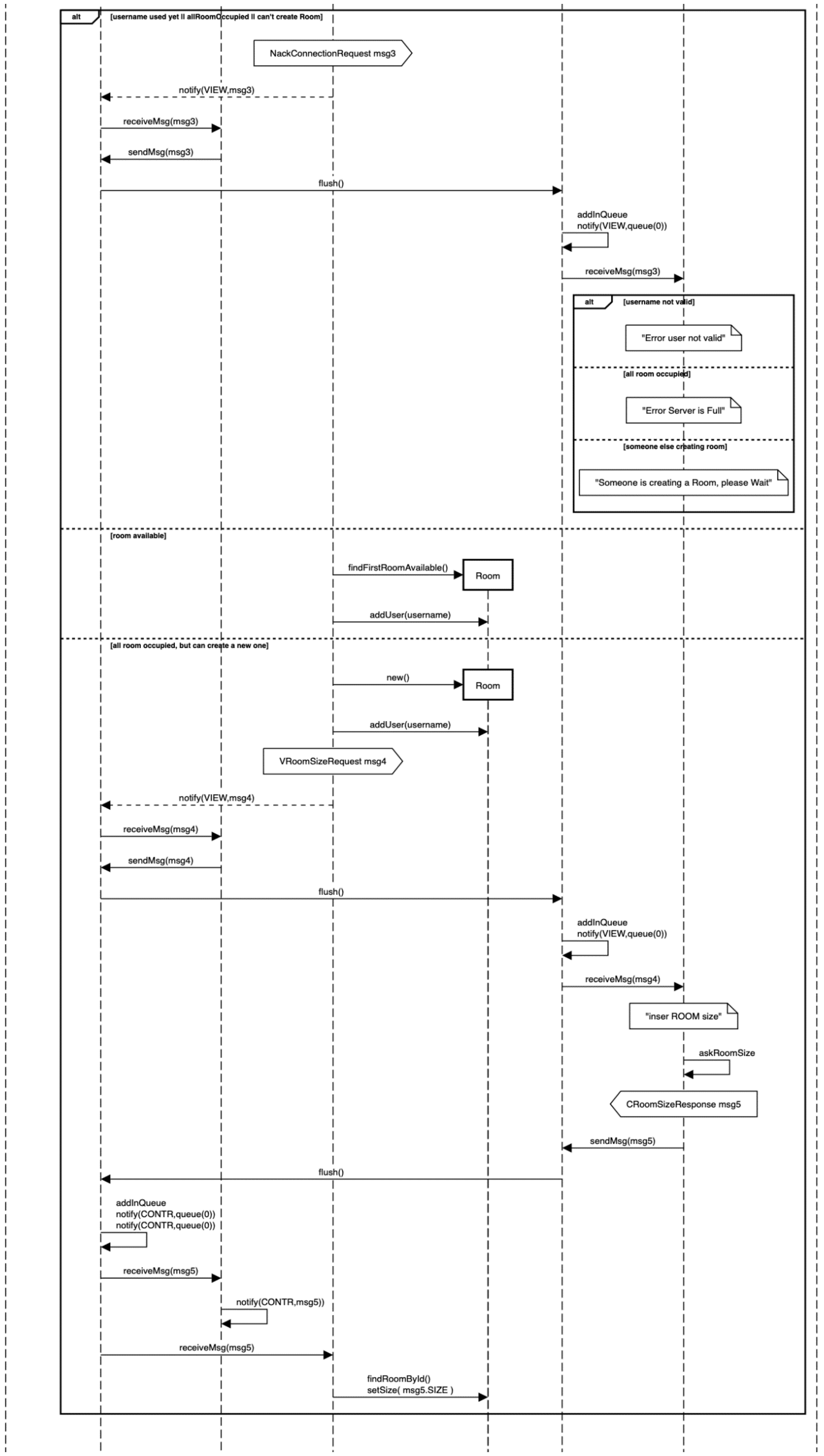
The server instantiates the socket and starts listening. The client chooses with which interface VIEW he wants to play (CLI or GUI), then is instantiated the ClientSocket, and using a specific port and a specific IP the Client asks connection to the Server; if the Server accepts the connection, it creates a ClientHandler object, which will represent the client in the Server part. It manages the communication with a specific client, so it is instatiated foreach Client that asks a connection. Thanks to this class, the server is able to manage multiple Clients simultaneously.

ClientSocket and ClientHandler are the only two that have an access to the Streams in the network. They have a queue of messages inside: if the message is a ping/pong it is considered only to know from each part that the connection is still Alive, otherwise is taken one message at time.

Then will be created the VirtualView object that will simulate the view in the server side. This will be used togheter as a CONTROLLER and a VIEW, so itself decides which messages need to be sent to the client and which to the server.

After all of that is done, the client sends to the server the login information and the server checks them. If login is successful, so the client has chosen an unused username or there is at least one room that isn't full and in that moment any other client is trying to create a Room, the client is inserted in the last available Room, or is created a new one of the SIZE decided by the player. Now the game is created and can start.




alt
[username used yet || allRoomOccupied || can't create Room]
NackConnectionRequest msg3
notify(VIEW,msg3)
receiveMsg(msg3)
sendMsg(msg3)
flush()
addInQueue
notify(VIEW,queue(0))
receiveMsg(msg3)
alt
[username not valid]
"Error user not valid"
[all room occupied]
"Error Server is Full"
[someone else creating room]
"Someone is creating a Room, please Wait"
[room available]
findFirstRoomAvailable()
Room
addUser(username)
[all room occupied, but can create a new one]
new()
Room
addUser(username)
VRoomSizeRequest msg4
notify(VIEW,msg4)
receiveMsg(msg4)
sendMsg(msg4)
flush()
addInQueue
notify(VIEW,queue(0))
receiveMsg(msg4)
"inser ROOM size"
askRoomSize
CRoomSizeResponse msg5
sendMsg(msg5)
flush()
addInQueue
notify(CONTR,queue(0))
notify(CONTR,queue(0))
receiveMsg(msg5)
notify(CONTR,msg5))
receiveMsg(msg5)
findRoomById()
setSize(msg5.SIZE)