Use/Case

# Connect (Use Case)

## Successful Connection

1. Description: The user requests to connect to the Client GUI using an IP address
2. Pre Condition: The user has internet connection
3. Post Condition: The user has connected to the Client GUI
   1. User enter ip address ‘127.0.0.0’
   2. User connects to Client GUI

## Unsuccessful Connection

1. Description: The user requests to connect to the Client GUI using an IP address
2. Pre Condition: The User has internet connection
3. Post Condition: The User has not connected to the Client GUI
   1. User enter ip address ‘127.0.0.1’
   2. User does not connect to Client GUI

# Disconnect (Use Case)

## Successful Disconnect

1. Description: The user requests on the Client GUI to disconnect from the server
2. Pre Condition: The user is already connected to the Client GUI
3. Post Condition: The client has disconnected from the Client GUI
   1. User selects the [Disconnect] button on Client GUI
   2. User disconnects from Client GUI

# Log in(Use Case)

## Successful Log in

1. Description: The client requests to Log in to an account and enters a username and password. Client sees code sent to email.
2. Pre Condition: The client has successfully connected to the server and a thread has started for that socket connection
3. Post Condition: The client has successfully logged in to the user a [logged in state]
4. Story:
   1. Client joins server using server ip address
   2. Client clicks button [Log in]
   3. Client enters username: ‘justinbouse2’
   4. Client enters password: ‘password1’
   5. Clients reads output saying that the password does not meet minimum requirements

## Log in attempts with Lock out

1. Description: The client requests to Log in to an account and enters a username and password
2. Pre Condition: The client has successfully connected to the server and a thread has started for that socket connection
3. Post Condition: The user has has been locked out
4. Story:
   1. Client joins server using server ip address
   2. Client clicks button [Log in]
   3. Client enters username: ‘justinbouse2’
   4. Client enters password: ‘password’
   5. Clients reads output saying invalid login information try again
   6. Client enters username: ‘justinbouse2’
   7. Client enters password: ‘passworD’
   8. Clients reads output saying invalid login information try again
   9. Client enters username: ‘justinbouse2’
   10. Client enters password: ‘password2’
   11. Client reads output saying your username has been locked out use password recovery

# New User Registration(Use Case)

## Account creation success

1. Description: The user requests to register an account and enters a username, password and email address. Client sees output successful account creation
2. Pre Condition: The user has successfully connected to the CLIENT GUI and a thread has started for that socket connection
3. Post Condition: An account with the users information has successfully been created and the information stored
4. Story:
   1. User joins server using server ip address
   2. User clicks button [Create Account]
   3. User enters username: ‘justinbouse2’
   4. User enters password: ‘password1’
   5. User enters emails: justinbouse14@gmail.com
   6. User reads output on “Account creation successful”

## Unsuccessful Account creation

1. Description: The client requests to create an account and enters a username, password and email address. Client is not able to create an account
2. Pre Condition: The user has successfully connected to the server and a thread has started for that socket connection
3. Post Condition: An account with the information entered by the user has not been created
4. Story:
   1. User joins server using server ip address
   2. User clicks button [Create Account]
   3. User enters username: ‘justinbouse2’
   4. User enters password: ‘pass’
   5. User enters emails: justinbouse14@gmail.com
   6. User reads output ‘minimum requirements for account information not met’

## Account already exists with username entered

1. Description: The user requests to create an account and enters ausername, password and email address. Client sees output that account with username already exists
2. Pre Condition: The client has successfully connected to the server and a thread has started for that socket connection
3. Post Condition: An account with the user information has not been created
4. Story:
   1. Client joins server using server ip address
   2. Client clicks button [Create Account]
   3. Client enters username: ‘justinbouse2’
   4. Client enters password: ‘password1’
   5. Client enters emails: justinbouse14@gmail.com
   6. Clients reads output saying that the username already exists

# Server (Use Case)

1. Case 1
   1. Name: Query Registered Users
   2. Description: The server will receive a request to check the amount of users that are registered.
   3. Pre Condition: Server receives total registered users request.
   4. Post Condition: Server returns a message with number of registered users.
   5. Story:
      1. Admin logs on to the server.
      2. Admin sends the server a request to count the number of registered users and return the number.
      3. Server connects to the database and gets the number of registered users.
      4. Server sends the admin a message with the number.
2. Case 2
   1. Name: Query Logged-in Users (active)
   2. Description: The server will receive a request to check the amount of users that are currently logged in.
   3. Pre Condition: Server receives total logged-in users request.
   4. Post Condition: Server returns a message with the number of active logged-in users.
   5. Story:
      1. Admin logs on to the server.
      2. Admin sends the server a request to count the number of active logged-in users and return the number.
      3. Server connects to the database and gets the number of active logged-in users.
      4. Server sends the admin a message with the number.
3. Case 3
   1. Name: Query Users Logged In (inactive)
   2. Description: The server will receive a request to check the amount of users that are currently logged in.
   3. Pre Condition: Server receives total logged-in users request.
   4. Post Condition: Server returns a message with the number of inactive logged-in users.
   5. Story:
      1. Admin logs on to the server.
      2. Admin sends the server a request to count the number of inactive logged-in users and return the number.
      3. Server connects to the database and gets the number of inactive logged-in users.
      4. Server sends the admin a message with the number.
4. Case 4
   1. Name: Query Users Locked Out
   2. Description: The server will receive a request to check the amount of users that are have been locked out.
   3. Pre Condition: Server receives total locked-out users request.
   4. Post Condition: Server returns a message with the number of locked-out users.
   5. Story:
      1. Admin logs on to the server.
      2. Admin sends the server a request to count the number of locked-out users and return the number.
      3. Server connects to the database and gets the number of locked-out users.
      4. Server sends the admin a message with the number.
5. Case 5
   1. Name: Query Connected Users
   2. Description:
   3. Pre Condition:
   4. Post Condition:
   5. Story: