Ali Al-Sharefi
Dat18C
27-06-2019
Sparring med Wajid

Mandatory assignment – Chat box/Chat server

Arbejds Proces:

Til at starte blev der dannet en ide forhold til hvilke stumper af dele koden skulle indeholde, Først og fremmest skulle der skabes en forbindelse mellem server og client, her er der blevet taget udgangspunkt i Grabba eksemplet fra fronter. Da det var på plads, sørget vi for at der kom en kommunikation på tværs af server og klient. Derefter Sørget vi for at selve samtale blev udprintet i en logfil for sig selv. Dernæst sørget vi for at tilføje tråd pulje som skulle sikre sig at der kan tisluttes flere klienter.

Hvad programmet består af:

Mit program består af 3 forskellige klasser vi har ServerThread, Server samt en Client.
”ServerThread” tager sig af Threads, samt loggeren. Det betyder at hver gang en klient opretter forbindelse til serveren, vil serveren give besked at klienten nu har oprettet forbindelse til serveren.
Serveren kan tage imod flere bruger, det maksimale antal bruger ligger på 5, Dette betyder vi har 5 threads kørende. Derudover vil serveren også kreere en log, dette betyder at hver gang der har været en samtale i den pågældende server. Vil den skrive alt historik ud i en TXT fil.

”Server” Sørger for at der er mulighed for at, der kan tilsluttes flere klienter. Så den vil blandt andet kreere en Thread pool med maximum antal klienter der har mulighed for at tilgå det.
”Client” håndterer adgangen og hvilken port den pågældende client, vælger at koble sig på.
Som klient skal du vælge mellem 5 porte. Når du har valgt den pågældende port du vil koble dig på, SÅ vil du så blive bedt om at taste et brugernavn ind. Brugernavnet opfylder opgaven krav, her anvendes der et regex pattern. Når du så har valgt dit brugernavn, er du så connected til serveren. Har du planer om at forlade serveren skal du skrive ”STOP” med store bogstaver, så vil socketen lukke forbindelsen ned.

Ali Al-Sharefi
Dat18C
27-06-2019
Sparring med Wajid

Design pattern:

Forhold til design pattern har jeg ikke rigtig kunne fastgøre hvilket pattern der er blevet anvendt.

Fokusset har primært været at lave et program der kan køre, og opfylde opgavens krav

Jeg synes programmet hælder sig mere over i Observer pattern, Da ServerThread håndtere Serveradgangen, og Client håndterer clienterne. Men personligt vil jeg selv sige vi ikke anvender som sådan et design i det pågældende program.

Til gengæld anvendes der et Regex pattern til både at styre Usernames characters og antallet af ord der må skrives i samtalen..

Konklusion:

Programmet fungerer således at man kobler sig op på serveren og igennem serveren kan man observere samtalen fra de forskellige ”Clienter” der har oprettet forbindelse. Det betyder at Clienterne ikke kan se hinandens besked, men man kan se alle beskeder på serveren. Der mulighed for at tilkoble 5 klienter på en server.

From client to server. The user name is given by the user. Username is max 12 chars long, only letters, digits, ‘-‘ and ‘_’ allowed. – UDFØRT

From server to client. Client is accepted. – UDFØRT

From server to client. Client not accepted. Duplicate username, unknown command, bad command or any other errors. – UDFØRT

From client to server. From server to all clients. First part of message indicates from which user it is, the colon(:) indicates where the user message begins. Max 250 user characters. - UDFØRT

From client to server. Client sends this heartbeat alive every 1 minute. – IKKE UDFØRT

From client to server. Client is closing down and leaving the group. – UDFØRT

LIST <> From server to client. A list of all active user names is sent to all clients, each time the list at the server changes. – IKKE UDFØRT