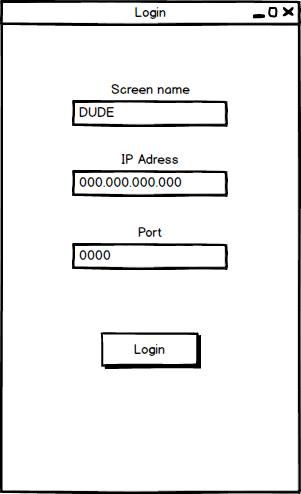
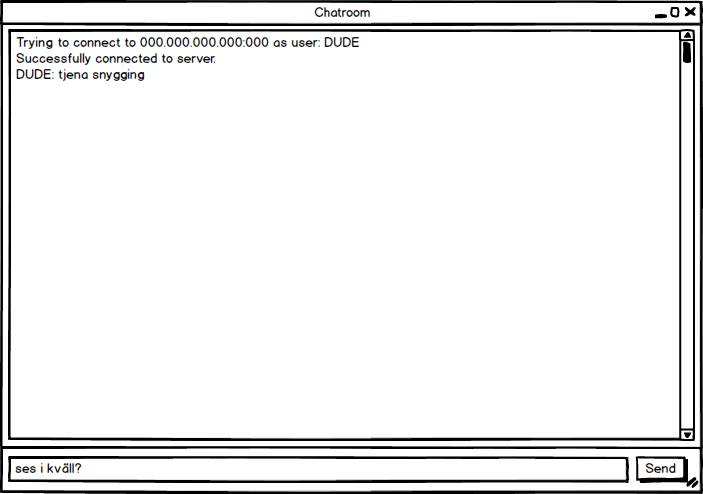
Projektplan – Chatklient

**Programbeskrivning**  
En enkel chatklient. Den ska ha en loginsida där du anger ditt screenname/nick, samt IP-adress och port till den server du vill ansluta till. I chatrummet ska man kunna kommunicera genom att skicka meddelanden genom denna server. Dessa meddelanden ska dyka upp i en chatruta som är synlig för alla som är anslutna till en viss server. Nedanstående konceptskiss illustrerar hur programmet kommer att se ut.



**Användarbeskrivning**Programmet är användbart för alla människor som vill ha en enkel chatservice för att kommunicera med varandra. De behöver ha vissa grundläggande kunskaper om hur en chatklient fungerar, och de måste kunna hitta rätt IP-adress och port för att ansluta till en viss server.

**Användarscenarier**Scenario 1 – Logga inDet första personen ser när hen öppnar programmet är startskärmen som ser ut ungefär som i ovanstående skiss. Personer skriver in sitt önskade screen name. För att kunna använda programmet måste användaren ha en IP-adress och port till den server hen vill ansluta till. Dessa uppgifter skrives in i respektive ruta. Därefter trycker användaren på login-knappen.

Scenaro 2 – Använd chatklienten (chatta)  
Efter användaren har loggat in dyker nästa ruta upp, och hen får ett meddelande som säger att hen lyckats ansluta till servern. Användaren kan nu skriva in meddelanden i en textruta längst ned på fönstret, och trycka på send för att skicka meddelandena till servern. Dessa meddelanden skrives ut i chatrutan, tillsammans med de andra meddelandena som servern tar emot från andra användare.

**Testplan**

För att kontrollera att programmet fungerar som det ska kommer vi utföra enhetstester med JUnit. Därefter kommer vi att testa programmet i praktiken, dels genom att själva använda oss av det, men även genom att låta andra människor testa programmet. Användarnas uppgifter kommer att bestå av att starta programmet, logga in (ansluta till en server) och skicka meddelanden. Deras synpunkter kommer sedan kunna användas för att förbättra programmet och fixa eventuella buggar.

**Programdesign**

Det är i dagsläget svårt att avgöra precis vilka klasser vi kommer att använda oss av, då vi inte har så mycket erfarenhet av nätverksprogrammering. Vartefter vi lär oss mer om hur detta fungerar, kommer vi få allt mer klarhet i hur vi ska utforma programmet. Det finns dock några klasser vi tror att vi kommer att använda oss av.

De två nyckelklasser är server och client. Serverklassen ansvarar för kommunikationen mellan klienterna, och behöver således en ”receive”-metod såväl som en ”send”-metod. Client-klassen ansvarar för kommunikationen mellan servern och de enskilda klienterna. Denna klass behöver således också send- och receivemetoder.

Utöver dessa nyckelklasser kommer vi behöva någon klass för själva chatfönstret (ChatWindow) och en klass för loginfönstret (LoginWindow). För att skriva ut meddelanden behöver vi en print-metod i ChatWindow-klassen. För att ansluta till servern behöver vi någon typ av ConnectToServer-metod i LoginWindow-klassen.

**Tekniska frågor**

För att kunna bygga vårt system måste vi skaffa information inom följande områden:

* Github.
  + Hur använder man sig av det?
    - Hur sätter man upp ett projekt som kan redigeras över github?
* Nätverksprogrammering.
  + Hur ansluter man en klient till en server
    - lokalt?
    - över internet?
* Objektorientering/design.
  + Hur skapar man system med bra objektorientering?
    - Klassschema.
    - Arv.
    - Interface.

**Arbetsplan**

Vi kommer till stor del att arbeta med projektet tillsammans. Detta för att kunna diskutera olika tillvägagånssätt och komma överens om programmets utformning. Projektet kommer att genomföras enligt följande arbetsplan:

1. Lära oss github på egen hand.
2. Läsa på om nätverksprogrammering på egen hand.
3. Strukturera upp programmet med hjälp av t.ex. klassdiagram tillsammans.
4. Dela upp projektet i olika ansvarsområden tillsammans.
5. Koda, tillsammans såväl som på egen hand med våra egna ansvarsområden.
6. Enhetstester – Få programmet att fungera.
7. Användartester.
8. Förbättring.
9. Färdigställ programmet.

Under hela projektets gång kommer vi att dokumentera vårt arbete och sammanfatta detta i en slutrapport.