

FAGRAPPOR I TTM4810 EKSPERTER I TEAM
MOBILKOMMUNIKASJON FOR BEDRET LIVSKVALITET 2004

Gruppe 4: **Peter Andreas Breunig, Stian Dalene Gjedrem, Ståle Lygren
Bjarte Brautaset og Asgeir Torstensen**

Tittel: **Easy Nurse Call**

Beskrivelse:

I dag brukes mange forskjellige systemer for pasientvarsling på sykehus, også internt på sykehusene brukes ulike systemer. Vårt system vil skape noen endringer i organisasjonen på et sykehus. Vi har tatt utgangspunkt i det nye systemet som skal leveres ved St. Olavs Hospital i Trondheim, der hver enkelt pasient vil få en terminal. I tillegg ser vi for oss at hver enkelt sykepleier er utstyrt med en mobil enhet som kan sende og motta meldinger. Ved å vite hva pasienten ønsker, kan sykepleiere forberede seg før de kommer til pasienten. De vil da lettere kunne prioritere hvilke pasienter som trenger hjelp. Det vil i tillegg bli enkelt å få en oversikt over ledige ressurser på en sengepost og det vil bli lettere å få tak en bestemt sykepleier hvis ønskelig.

Innleveringsfrist: **April 21, 2004**

Levert: **April 21, 2004**

Veileder: **Lill Kristiansen**

Forord

Utgangspunktet for denne rapporten er det tverrfaglige prosjektfaget Eksperter i Team (EiT) ved Norges Teknisk-Naturvitenskapelige Universitet (NTNU). Faget er obligatorisk for 4. årskurs på sivilingeniørstudiet, og er på 7.5 studiepoeng. Prosjektet ble gjennomført våren 2004, i landsbyen Mobilkommunikasjon for bedret livskvalitet. Medlemmene av gruppen studerer ved forskjellige linjer ved NTNU, og har dermed forskjellig faglig bakgrunn. Gruppen har representanter fra linjene: Datateknikk, Energi og Miljø, Industriell økonomi og teknologiledelse, og Kommunikasjonsteknologi.

Vi har kartlagt eksisterende systemer for pasientvarsling gjennom vår forstudie og på bakgrunn av dette laget et system som forhåpentligvis kan være med på å forbedre hverdagen til både pasienter og sykepleiere ved et sykehus.

Gjennom de ca 14 ukene som faget har pågått har vi fått hjelp av flere personer som vi gjerne vil takke:

Arild Faxvaag, Førsteamanuensis ved Institutt for nevromedisin, for oppgavebeskrivelse samt mange bra tips og innspill.

Avdelingene A4, B4 og Kreftavdelingen ved St. Olavs Hospital, for hjelp til kartlegging av eksisterende pasientvarslingssystemer, samt tips og innspill.

Camilla Sivertsen, tidligere sykepleier og nå prosjektleder ved Cardiac Medical, for en interessant og motiverende presentasjon av pasientterminalen som Cardiac skal levere ved St. Olavs Hospital.

Til slutt vil vi takke landsbyens fasilitatorteam bestående av professor Lill Kristiansen og studentassistentene Ingrid Buli og Asle T. Elen for deres veiledning og innspill. Kanskje spesielt med tanke på gruppeprosessen.

Sammendrag

I forbindelse med utbyggingen av St. Olavs Hospital i Trondheim skal det installeres ny infrastruktur for IKT løsninger ved det nye sykehuset. Det nye kommunikasjonsanlegget vil ha støtte for blant annet trådløs kommunikasjon på hele sykehuset. I forbindelse med dette ønsket Helse Midt-Norge ved Arild Faxvaag at studenter ved EiT landsbyen Mobil-kommunikasjon for bedret livskvalitet ved NTNU, skulle se på muligheter for å utnytte det nye kommunikasjonsanlegget til å bedre hverdagen for sykepleiere og pasienter ved sykehuset.

Prosjektet startet med en forstudie på hvilke systemer som finnes for pasientvarsling ved sykehuset i dag, samt å finne andre studier som er blitt utført innenfor området pasientvarslingssystemer. Gjennom undersøkelser ved forskjellige avdelinger ved sykehuset ble det klart at flere ansatte ikke så noe nytte i et nytt pasientvarslingssystem. De avdelingene som hadde gamle systemer ville heller oppgradere til de nyere systemene som eksisterer ved sykehuset. Men det var også ansatte som mente at ett nytt slikt system ville være med på å forbedre hverdagen til pasienten og dem selv. På bakgrunn av dette og en mer omfattende studie utført i USA på liknende systemer ble det bestemt at vi skulle forsøke å utvikle et nytt og bedre pasientvarslingssystem basert på trådløs teknologi.

Neste del av prosjektet gikk så ut på å se hvilke krav brukerne ville stille til et slikt system og hvilke funksjoner ett slikt system burde inneholde. På bakgrunn av dette ble det laget systemkrav som skulle brukes som utgangspunkt til å implementere systemet.

I siste del av prosjektet ble det implementert en enkel prototyp som bygger på de systemkravene vi kom frem til tidligere i prosjektet. Prototypen mangler en god del funksjonalitet, men den gir et innblikk i hvordan vi ser for oss at et nytt pasientvarslingssystem kan se ut. Den er også ett godt utgangspunkt for å fortsette videre med utviklingen av systemet for eksempel av andre studenter ved NTNU.

Innholdsfortegnelse

Forord.....	3
Sammendrag.....	4
Figurliste.....	7
1 Innledning.....	8
2 Forstudier	9
2.1 Eksisterende systemer	9
2.1.1 Avdeling B4 ved St. Olavs Hospital	9
2.1.2 Sengepost 1 ved Kreftavdelingen St. Olavs Hospital	10
2.2 Annen studie.....	10
2.3 Konklusjon fra befaring og studier.....	11
3 Systemet	12
3.1 Use-Cases	12
3.1.1 Logge på systemet.....	13
3.1.2 Sykepleier utfører oppdrag.....	14
3.1.3 Kommunikasjon mellom sykepleiere.....	15
3.1.4 Oppdatere Status	16
3.1.5 Sykepleier logger av.....	16
3.2 Brukerkrav.....	17
3.2.1 Overordnet systembeskrivelse.....	17
3.2.2 Systemets funksjoner.....	18
3.2.3 Data i systemet	23
3.3 Systemkrav	24
3.3.1 Rammekrav	24
3.3.2 Systemets funksjonelle egenskaper.....	26
4 Implementering	27
4.1 Prototyp	27
4.1.1 Server	27
4.1.2 Klient.....	29
4.2 Videre utvikling.....	32
4.2.1 Terminal for administrasjon	32
4.2.2 Grupper.....	32
4.2.3 ”Smart” server som velger sykepleier.....	32
4.2.4 Geografisk lokalisering av sykepleiere	33
4.2.5 Kryptering	33
5 Diskusjon og konklusjon.....	34
5.1 Sikkerhet og konfidensialitet.....	34
5.2 Skepsis mot nye systemer	35
6 Referanser.....	36
Vedlegg A - Brukerveiledning for prototypen	37
A.1 Server.....	37
A.1.1 Bruker Status	37
A.1.2 Sende beskjed	38
A.1.3 Simulere pasient tilkalling.....	38
A.2 Klient.....	39
A.2.1 Logge inn.....	39
A.2.2 Hovedvindu	39
A.2.3 Endre egen status.....	40

A.2.4 Sende melding til andre brukere.....	41
A.2.5 Logge av	42
A.1.7 Avslutte programmet	42
A.2.8 Svare på pasientforespørsler.....	42
Vedlegg B – UML Klassediagram prototyp	43
Topp-nivå	43
Pakke: Server.....	44
Pakke: Client	45
Pakke: ClientGUI	46
Vedlegg C – XML-meldinger og konfigurasjonsfiler.....	47
C.1 XML-filer	47
C.1.1 Konfigurasjonsfil for server.....	47
C.1.2 Brukerdatabase	47
C.2 XML-meldinger.....	48
C.2.1 Logon.....	48
C.2.2 Logon respons.....	49
C.2.3 Status	49
C.2.4 Tekstbeskjed	50
C.2.5 Pasient tilkalling	50
C.2.6 Svar på pasienttilkalling	50
Vedlegg D - CD med kildekode.....	51

Figurliste

Figur 3-1 Oversikt over use-cases	13
Figur 3-2 Use-case: Logge på systemet	13
Figur 3-3 Use-case: Sykepleier utfører oppdrag	14
Figur 3-4 Use-Case: Kommunikasjon mellom sykepleiere	15
Figur 3-5 Use-Case: Oppdatere Status	16
Figur 3-6 Use-Case: Sykepleier logger av	16
Figur 3-7 Systembeskrivelse	17
Figur 3-8 MSC Logge på	19
Figur 3-9 MSC Kommunikasjon.....	20
Figur 3-10 MSC Utfør oppdrag.....	21
Figur 3-11 MSC Logg av	22
Figur 3-12 ER-modell over data i systemet	23
Figur 4-1 Server prototypens brukergrensesnitt	28
Figur 4-2 Send beskjed dialogen	29
Figur 4-3 Skjermbilde fra prototypen.....	30
Figur 4-4 Skjermbilde som viser kommunikasjon med annen bruker	31
Figur A-1: Skjermbilde av oppstart av serveren	37
Figur A-2: Skjermbilde av Send Beskjed dialog	38
Figur A-3: Skjermbilde som viser påloggingsvindu	39
Figur A-4: Skjermbilde som viser hovedvinduet	40
Figur A-5: Utsnitt av skjermbilde som viser status-menyen	41
Figur A-6: Skjermbilde av send melding til andre brukere	41
Figur A-7: Skjermbilde som viser samtale med annen bruker	41
Figur A-8: Utsnitt av skjermbilde som viser Program-menyen	42
Figur A-9: Skjermbilde som viser forespørsel fra pasient	42
Figur C-1 XML-fil - Konfigurasjonsfil for server	47
Figur C-2 XML-fil - Brukerdatabase	47
Figur C-3 Mal for XML-beskjed.....	48
Figur C-4 XML-beskjed – Logon	48
Figur C-5 XML-beskjed - logon_fail	49
Figur C-6 XML-beskjed - logon_ok	49
Figur C-7 XML-beskjed – status.....	49
Figur C-8 XML-beskjed – tekstbeskjed	50
Figur C-9 XML-beskjed - Pasient tilkalling	50
Figur C-10 XML-beskjed - Svar på pasienttilkalling.....	50

1 Innledning

Vi er en av seks grupper i landsbyen Mobilkommunikasjon for bedret livskvalitet. Dette er en av flere landsbyer i EiT våren 2004. Med utgangspunkt i tittelen på landsbyen skulle vi finne en oppgave vi skulle løse som gruppe i løpet av vårsemesteret.

Etter å ha blitt introdusert for en rekke forskjellige problemstillinger av faglærer og Arild Faxvaag, falt valget på en oppgave om pasientvarsling.

Hvordan dagens systemer for pasientvarsling fungerer er beskrevet i kapittelet som omhandler forstudiene, men kort sagt fungerer det slik: pasienten trykker på en knapp som fører til et signal på vaktrommet og tenner et lys utenfor pasientens rom. Sykepleierne må så oppsøke pasienten for å finne ut hva som er problemet. Vi ville forbedre dette systemet, slik at pleierne kan få mer nøyaktige meldinger om hva pasienten ønsker, og kan dermed gjøre eventuelle forberedelser på forhånd. I tillegg vil en viktig funksjonalitet i vårt system være lokaliseringssinformasjon (posisjonsinformasjon). På den måten kan sykepleierne bruke systemet til å finne andre pleiere.

Vi har ikke jobbet med hele problemstillingen for systemet. En annen gruppe som har jobbet med grensesnittet mot pasientene, og sending av meldinger fra pasientterminalen til serveren. Vi har sett på kommunikasjonen fra server til sykepleierne, og mellom sykepleierne. Gruppen i mellom seg har blitt enige om et format på meldingene som skal sendes, så det skal være en grei oppgave å sy de to løsningene sammen til et komplett system.

Vårt mål for arbeidet var å få laget en veldig enkel prototyp som kan vise hvordan vi tenker oss at systemet skal fungere. Informasjon om prototypen som er implementert finnes i kapittel 4 i denne rapporten.

2 Forstudier

2.1 Eksisterende systemer

Som en del av forstudiene foretok vi en befaring ved St. Olavs Hospital, for å se hvordan dagens system for pasientvarsling fungerer og hvordan sengepostene er organisert.

Vi besøkte to forskjellige avdelinger, avdeling B4 og sengepost 1 på Kreftavdelingen.

Avdeling B4 bruker et gammelt system som er vanlig på de fleste av sengepostene på St. Olav Hospital. Sengeposten på Kreftavdelingen har et nyere system, kalt Stentofon 8500.

2.1.1 Avdeling B4 ved St. Olavs Hospital

Her snakket vi med assisterende avdelingssykepleier Tone Nakken, og hun fortalte oss kort om hvordan sengeposten er organisert:

- Dagvakt:
 - 18 pasienter
 - 8 sykepleiere
 - 1 avdelingssykepleier
 - 1 assisterende avd. sykepleier
 - 1 sengepost sekretær (administrativt)
 - Sykepleierne ble tildelt pasienter ved begynnelse av hver vakt.
 - Avd. sykepleier og ass. avd. sykepleier har ikke pasientansvar, men bruker mesteparten av tiden på administrative oppgaver.
- Nattevakt:
 - Samme antall pasienter
 - 2 sykepleiere
 - Ikke tildelt pasienter

Denne avdelingen benytter som nevnt et gammelt og svært enkelt system for pasientvarsling. Pasienten trykker på en knapp ved sengen ved behov. Dette blir da varslet med lyd på vaktrommet, i tillegg til at en lampe lyser rødt på utsiden av det aktuelle rommet. Når sykepleieren ankommer rommet, kvitterer han/hun sin ankomst. Lampen på gangen utenfor rommet lyser da grønt. Når sykepleieren er ferdig, kvitteres det også for dette, og lampen

slukker. I utgangspunktet er sykepleierne tildelt sine pasienter som de skal ta dersom de er ledige, men i praksis er det nokså vilkårlig hvem som svarer på en tilkalling.

2.1.2 Sengepost 1 ved Kreftavdelingen St. Olavs Hospital

Her snakket vi med Tine Fykse Strøm. Denne sengeposten er organisert litt annerledes:

- 1 avd. sykepleier + 1 ass. avd. sykepleier
- 12 sykepleiere
- 28 sengeplasser

På denne sengeposten er sykepleierne delt inn i 4 faggrupper, med en sykepleier på hver gruppe som gruppeleder. Kreftavdelingen benytter seg altså av et litt nyere system kalt Stentofon 8500. Dette systemet er en videreutvikling av det som blir brukt på B4. I tillegg til lampen som lyser utenfor rommet, har de også en tavle i taket og et panel ved vaktrommet. Tavlen i gangen viser røde romnummer for ubesvarte tilkallinger, og grønne romnummer for besvarte tilkallinger. Panelet ved vaktrommet viser det samme som tavlen i gangen, og i tillegg viser den hvilken seng som kaller. Som for det eldre systemet må sykepleierer kvittere ved ankomst, og når han/hun er ferdig. Dette systemet kan også brukes for å tilkalle gruppeleder til vaktrommet. På panelet ved vaktrommet kan man taste inn enkle koder, for eksempel p1, som blir vist på tavla i gangen. Dette betyr da at leder for gruppe 1 skal møte på vaktrommet.

2.2 Annen studie

[1] refererer til en artikkel om en undersøkelse gjennomført ved to sykehus i USA, ett i midtvesten (MH), og ett på østkysten (EH). Utnyttelsen av sykepleierressursene (omfatter alle på avdelingen, ikke bare de med sykepleierutdannelse), og funksjonen av tilkallingssystemet ved en avdeling på hvert av de to sykehusene ble studert.

Begge avdelingene benyttet seg av et tilkallingssystem hvor sykepleierne gikk rundt med en personsøker. Når en pasient ved MH trykket på tilkallingsknappen, ble det varslet på vaktrommet, i tillegg til at kallet ble sendt til den sykepleieren som var ansvarlig for pasienten. Ved EH ble det benyttet et annet system. Tilkalling fra en pasient gikk til vaktrommet, der det satt en sekretær. Denne sekretæren tilkalte så den nødvendige

kompetansen via personsøkerne. I dette systemet var det også mulighet for kommunikasjon mellom pasient og sekretær, og videre mellom sekretær og sykepleier. Denne kommunikasjonen gjorde at pasienten kunne spesifisere hva den ønsket av hjelp, og denne informasjonen ble så brakt videre til den aktuelle sykepleieren av sekretæren.

I [1] står det mer utfyllende om resultatene fra undersøkelsen, men kort oppsummert viser undersøkelsen at mye tid ble spart ved EH fordi sykepleierne kunne forberede seg før de gikk inn til pasienten. Konklusjonen i [1] ble da at systemet ved EH førte til bedre forhold både for pasienter og for sykepleiere. En av fordelene for pasientene ved EH var at de fikk mye hurtigere svar på tilkallingen. Selv om de ikke tok kortere tid før pleieren var på plass, fikk pasientene svar med en gang, og følte derfor at de fikk raskere pleie. Sykepleierne kunne bruke systemet til å finne utstyr og personell, og sparte på denne måten en del tid.

2.3 Konklusjon fra befaring og studier

I tillegg til praktisk informasjon om organisering og funksjonalitet til pasientvarslingssystemene, spurte vi noen sykepleiere på de to avdelingene hva de synes om dagens system, og hva de tenkte om våre forslag til utbedringer. På avdeling B4 hvor de bruker det gamle systemet, ønsket de seg en fornyelse, men ikke slik som vi hadde tenkt. De ønsket å få Stentofon 8500, systemet som blir brukt på Kreftavdelingen. Det de derimot ikke ønsket var en mer spesifisert melding om hva pasienten ønsket. De var redde for at dette kunne føre til en prioritering av forskjellige tilkallinger, og at tilkallinger som burde vært besvart ikke ble fulgt opp fordi sykepleieren kunne gi dem for lav prioritet. De var også negative til tanken om at hver sykepleier skulle få sin egen PDA, eller tilsvarende, for å kommunisere med hverandre. På Kreftavdelingen var de godt fornøyde med systemet sitt, og stilte seg skeptiske til våre ideer. Det de derimot kunne se av nytte for systemet vårt, var muligheten for sykepleieren som fungerte som sekretær, og kunne dermed tilkalle andre sykepleiere ved telefon eller lignende. Dette var våre erfaringer fra en veldig liten undersøkelse, og i og med at andre har gjort mye mer omfattende undersøkelser med mer oppløftende resultater, bestemte vi oss for å fortsette med prosjektet.

3 Systemet

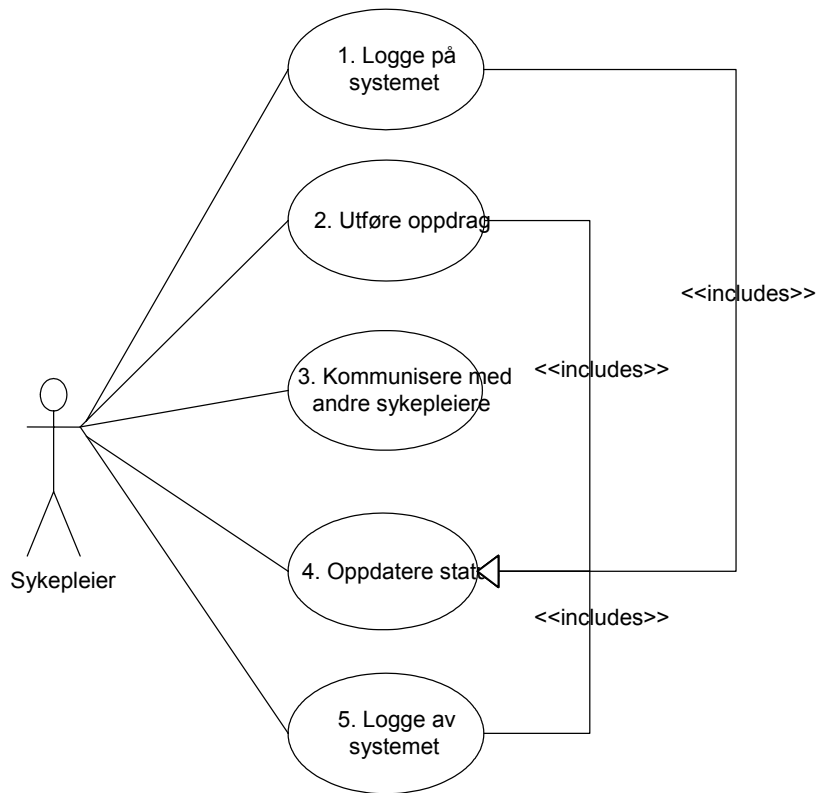
Selv om vår lille spørreunde på St. Olavs Hospital ikke ga grunn til å tro at systemet vårt ville bli godt mottatt, fortsatte vi med utviklingen.

Dette kapitlet er delt i 3 deler; *use-cases*, *brukerkrav* og *systemkrav*. *Use-casene* beskriver funksjoner brukeren av systemet ønsker å få utført. *Brukerkravene* er mer formelt. *Systemkravene* er av mer teknisk art og beskriver krav til selve systemet og maskinvaren som systemet skal kjøres på.

3.1 Use-Cases

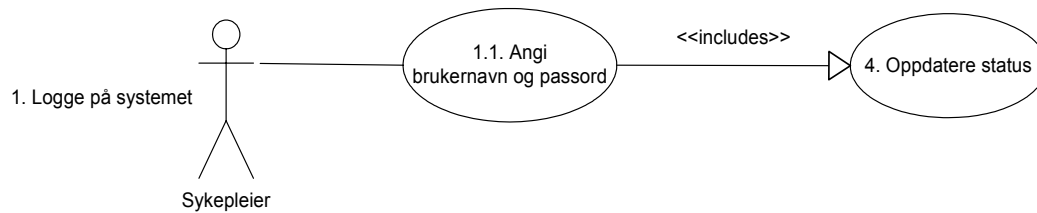
Et use-case er en kombinasjon av tekst og figur som skal beskrive en bestemt funksjon ved systemet. Use-case skal være enkelt, slik at det lett kan bli forstått av personer uten spesiell datakompetanse.

I våre use-cases er det en type aktør, sykepleieren. Denne rollen omfatter alle som jobber på en sengepost, ikke bare personell med sykepleierutdanning. En oversikt over de forskjellige use-casene er gitt i figur 3.1.



Figur 3-1 Oversikt over use-cases

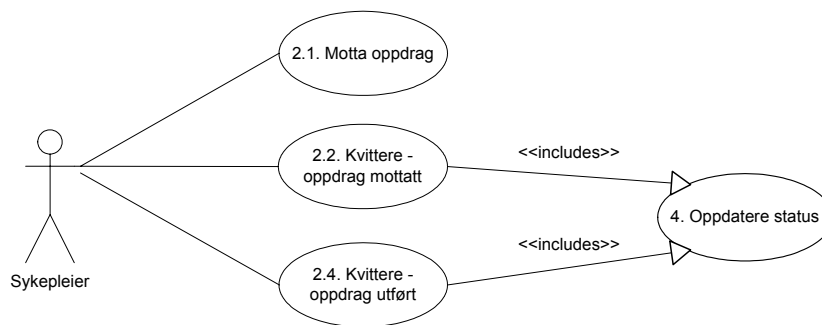
3.1.1 Logge på systemet



Figur 3-2 Use-case: Logge på systemet

Use Case	Sykepleier logger seg på
Aktør	Sykepleier
Trigger	Sykepleier velger funksjonen ”logg på” i brukerpanelet
Pre-betingelser	Sykepleier har fått tildelt en personlig, håndholdt terminal
Post-betingelser	<i>Sykepleier er enten:</i> 1. Pålogget systemet, eller 2. Fått en feilmelding som beskriver hvorfor ikke
Normal hendelsesflyt	1. Sykepleier får opp et skjermbilde med forespørsel om brukernavn og passord 2. Systemet kontrollerer at brukernavn og passord er korrekt 3. Sykepleier får beskjed om at påloggingen var vellykket 4. Systemet sender ut en oppdatert status til andre aktører i systemet om at denne sykepleieren nå er pålogget
Variasjoner	<i>Når det ikke oppnås kontakt med server:</i> 1. Sykepleier får beskjed om at systemet ikke er tilgjengelig <i>Ved feil brukernavn/passord:</i> 3. Sykepleier får beskjed om at brukernavn/passord ikke var vellykket og at pålogging var mislykket
Relatert informasjon	

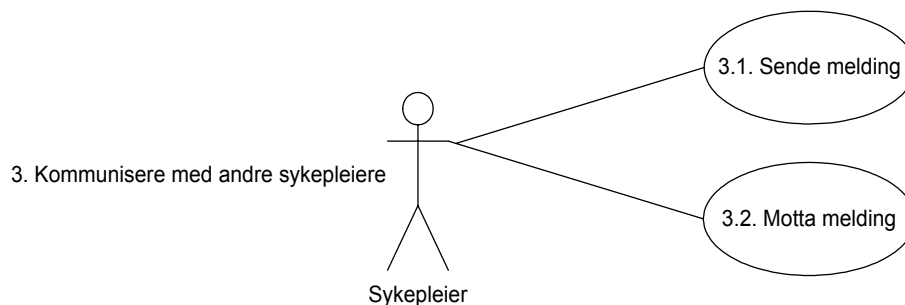
3.1.2 Sykepleier utfører oppdrag



Figur 3-3 Use-case: Sykepleier utfører oppdrag

Use Case	Sykepleier utfører oppdrag
Aktør	Sykepleier
Trigger	Pasient velger funksjonen ”tilkalle hjelp” på pasient terminal
Pre-betingelser	Sykepleier er ledig
Post-betingelser	Sykepleier har utført oppdraget.
Normal hendelsesflyt	1. Sykepleier mottar beskjed om at pasient trenger hjelp 2. Sykepleier kvitterer på at oppdraget er mottatt 3. Sykepleier kvitterer til slutt at oppdraget er utført
Variasjoner	<i>Hvis det tar for lang tid før sykepleier kvitterer på at oppdraget er mottatt:</i> 2b) Oppdraget går til en annen sykepleier <i>Når sykepleier ikke har mulighet til å ta oppdraget:</i> 2c) Sykepleier gir beskjed tilbake om at han/hun ikke kan ta oppdraget
Relatert informasjon	Status oppdateres fortløpende i hendelsesforløpet

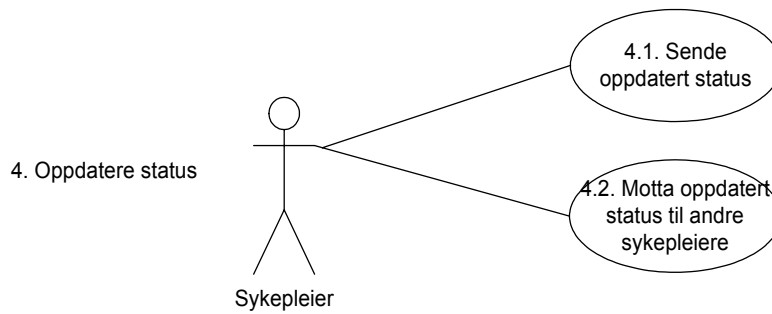
3.1.3 Kommunikasjon mellom sykepleiere



Figur 3-4 Use-Case: Kommunikasjon mellom sykepleiere

Use Case	Sykepleier kommuniserer med andre sykepleiere
Aktør	Sykepleier
Trigger	Sykepleier velger funksjonen ”kommunisere med andre”
Pre-betingelser	Sykepleier må være pålogget
Post-betingelser	
Normal hendelsesflyt	1. Sykepleier sender en melding 2. Sykepleier mottar en melding
Variasjoner	
Relatert informasjon	

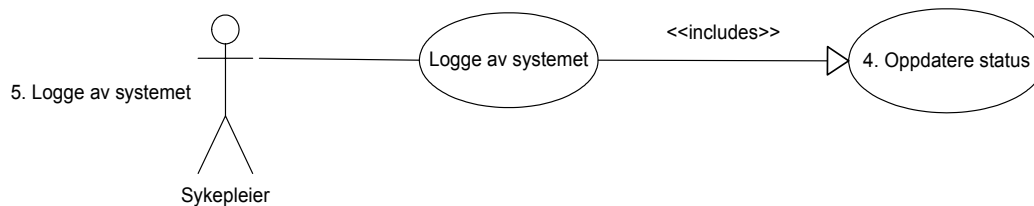
3.1.4 Oppdatere Status



Figur 3-5 Use-Case: Oppdatere Status

Use Case	Sykepleier oppdaterer status
Aktør	Sykepleier
Trigger	Endring av egen gruppemedlemmers status.
Pre-betingelser	Sykepleier må være pålogget
Post-betingelser	
Normal hendelsesflyt	1. Sykepleier sender status 2. Sykepleier mottar status, og oppdaterer.
Variasjoner	
Relatert informasjon	

3.1.5 Sykepleier logger av



Figur 3-6 Use-Case: Sykepleier logger av

Use Case	Sykepleier logger av systemet
Aktør	Sykepleier
Trigger	Sykepleier velger funksjonen "logge av"
Pre-betingelser	Sykepleier er logget på systemet
Post-betingelser	Status oppdatert – sykepleier ikke lenger logget inn
Normal hendelsesflyt	1. Sykepleier logger av
Variasjoner	
Relatert informasjon	

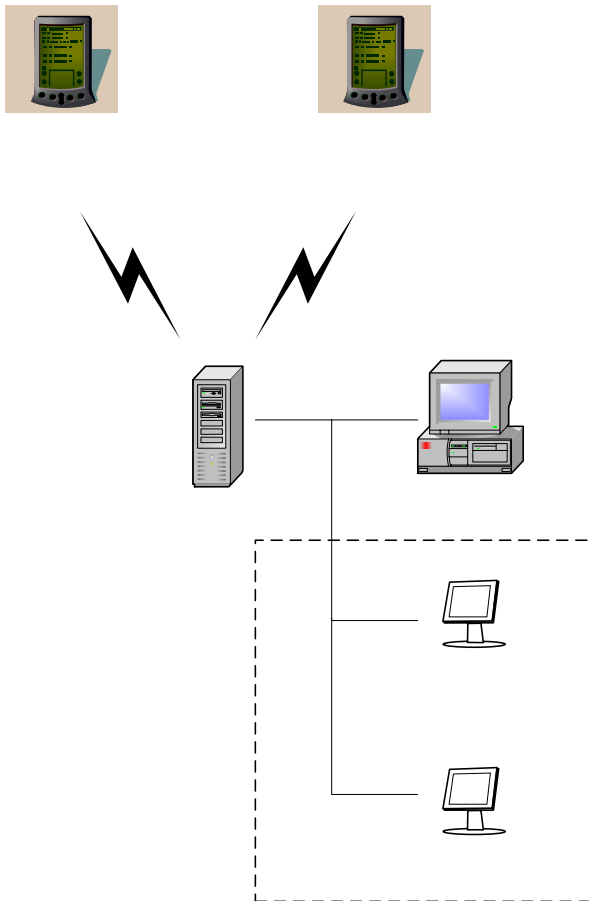
3.2 Brukerkrav

Denne delen skal først og fremst gi svar på hva slags krav brukerne har til systemet, og er første skritt på veien mot en mer fullstendig beskrivelse av systemet. Her vil man finne en overordnet beskrivelse av systemet, funksjoner levert av systemet, samt hvilke data som skal behandles.

3.2.1 Overordnet systembeskrivelse

Systemet består av følgende enheter (vist i figur 3-7):

- Trådløse, personlige terminaler (PDAer)
- Sentral server
- Fast terminal for administrasjon



Figur 3-7 Systembeskrivelse

I tillegg kommuniserer server med pasientterminaler, men disse er ikke omfattet av vårt prosjekt. Vi trenger bare å vite formatet på meldingene som sendes fra pasientterminalen. Hver sykepleier skal ha en håndholdt terminal. På denne terminalen skal de kunne motta og sende meldinger, både til og fra pasienter og andre sykepleiere. Serveren vil fungere som en ruter, og sørge for at en melding blir sendt til riktig sykepleier.

For administrasjon av systemet, har man en fast terminal. Denne kan brukes til å sette opp grupper av sykepleiere, fordele pasienter mellom sykepleierne m.m.

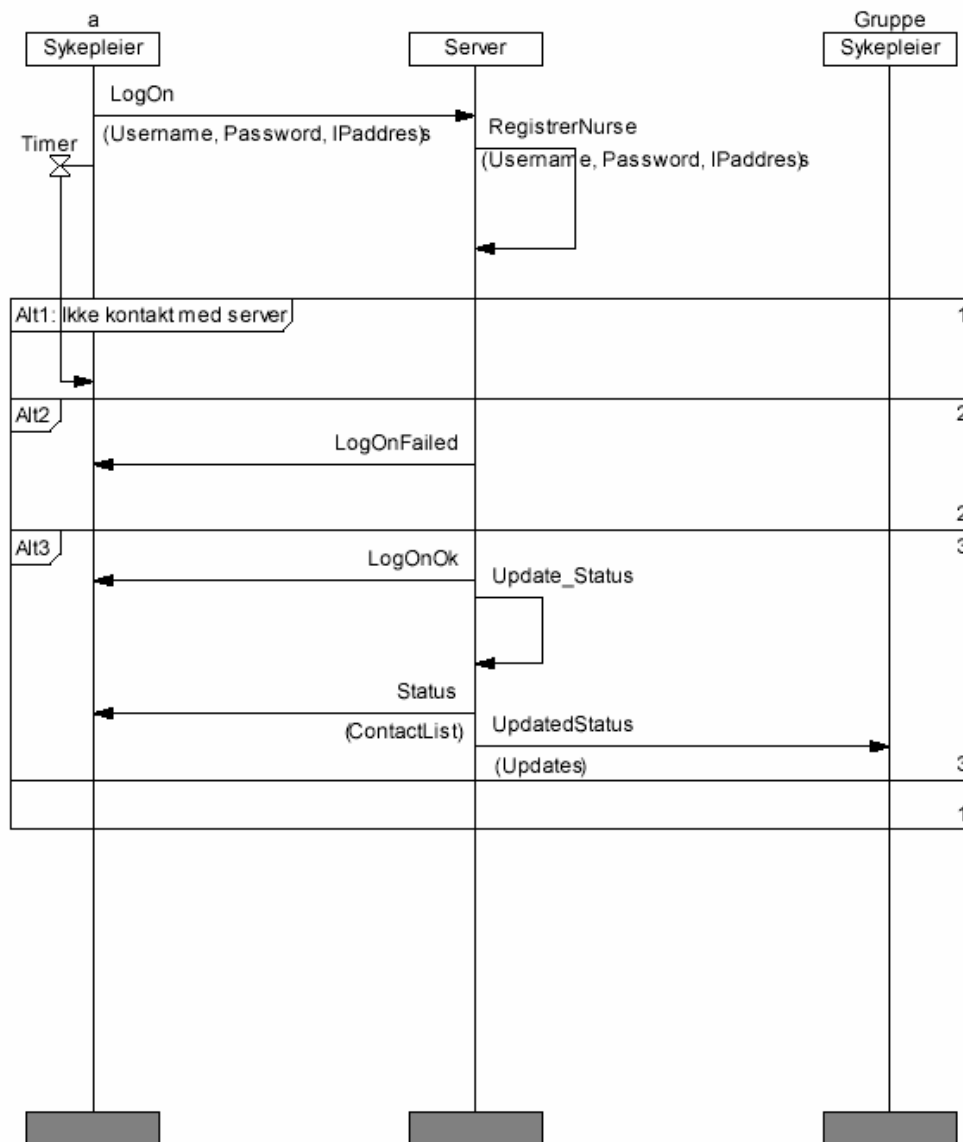
3.2.2 Systemets funksjoner

De forskjellige funksjonene i systemet er samlet i 4 hovedfunksjoner:

- Logge på systemet
- Kommunikasjon mellom brukere
- Utføre oppdrag
- Logge av systemet

Disse hovedfunksjonene er beskrevet ved hjelp av Message Sequence Charts, MSCs. De forskjellige aktørene i disse MSCs er *Sykepleier*, *Server* og *Pasient*. Siden vi har konsentrert oss om sykepleierdelen av systemet omhandler disse diagrammene funksjoner som utføres av sykepleiere. Aktøren *pasient* er kun tatt med i tilfellet hvor dette var nødvendig for å vise funksjonen ”utføre oppdrag”.

MSC Logge på



Figur 3-8 MSC Logge på

Beskrivelse av diagram Logge På:

I det en sykepleier sender en forespørsel om pålogging til serveren, er det tre alternativer for hendelsesforløpet.

Alternativ 1:

Det er ikke kontakt med serveren. Av en eller annen grunn er det ikke forbindelse mellom terminal og server, og påloggingsforsøket mislykkes derfor.

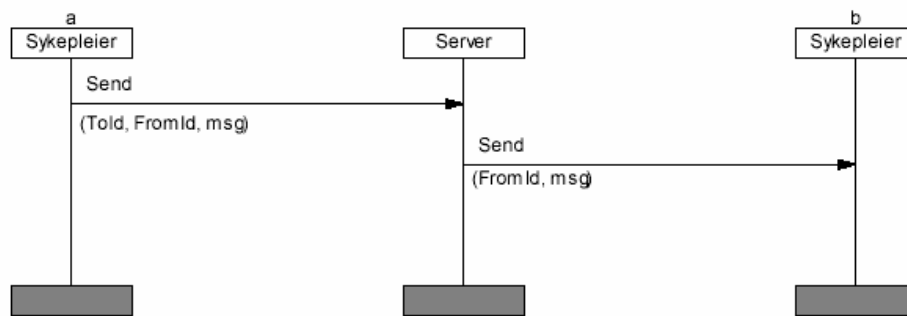
Alternativ 2:

Påloggingen feilet. Dette er fordi brukernavn eller passord er feil.

Alternativ 3:

Pålogging er vellykket, og oppdatert status for andre sykepleiere i samme gruppe som sykepleier a, blir sendt til denne sykepleieren. I tillegg blir alle de andre i samme gruppe oppdatert om den nye statusen til sykepleier a.

MSC Kommunikasjon

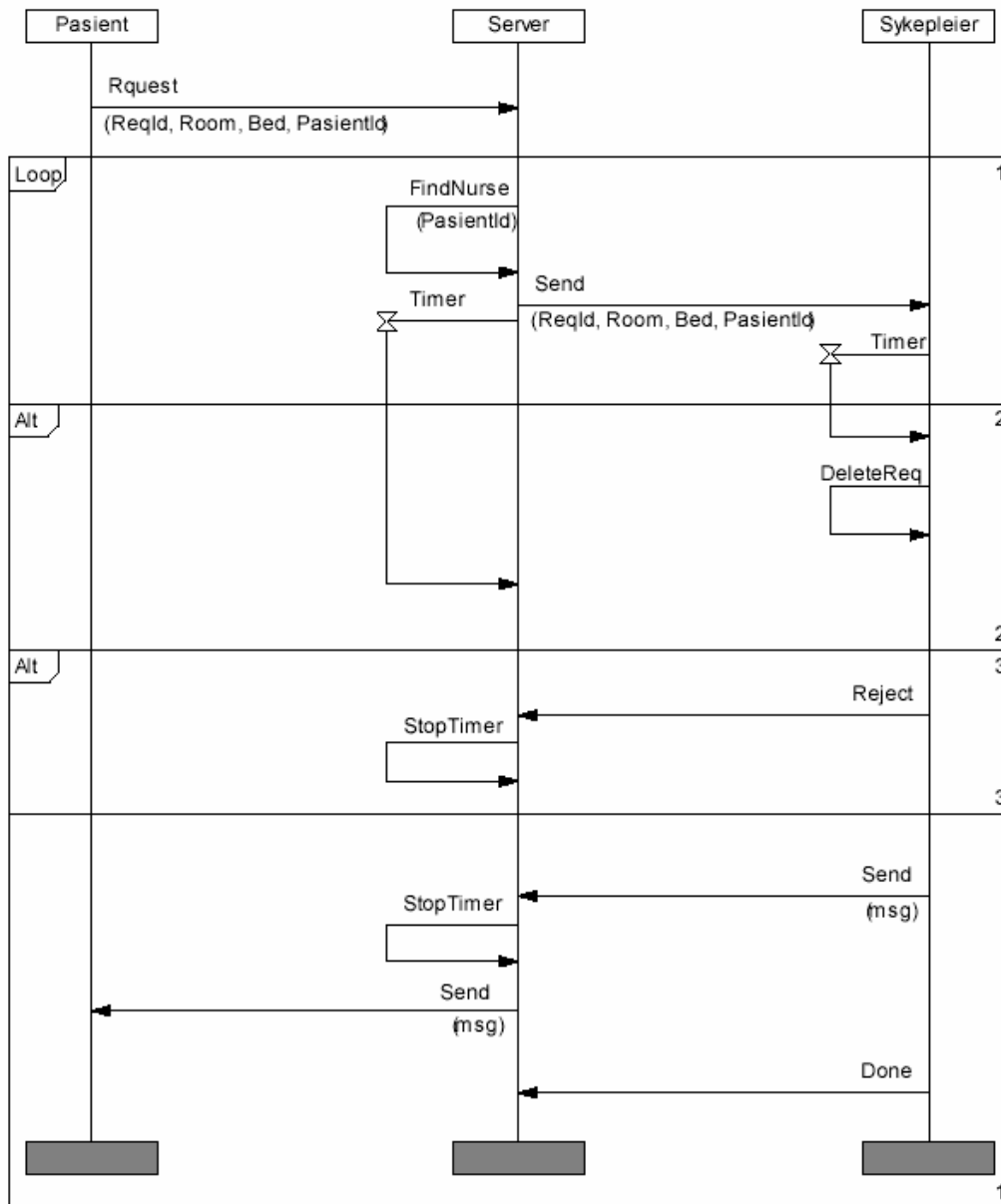


Figur 3-9 MSC Kommunikasjon

Beskrivelse av diagram Kommunikasjon:

Diagrammet er veldig enkelt. En melding blir sendt fra sykepleier a til sykepleier b. Serveren ligger i mellom og fungerer som en ruter. Ut fra FromId finner serveren IP-adressen til sykepleier b, og sender denne videre. Diagrammet forutsetter at forbindelsen mellom serveren og terminalene er i orden.

MSC Utfør_Oppdrag



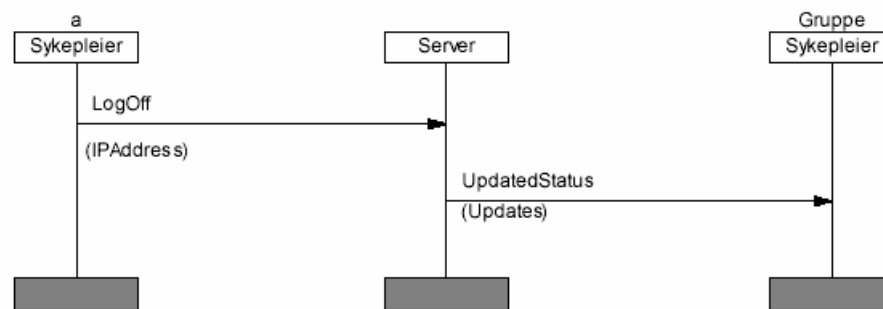
Figur 3-10 MSC Utfør oppdrag

Beskrivelse av diagram Utfør Oppdrag:

Pasienten sender først en forespørsel til serveren. Serveren bruker så dataene den har lagret til å finne ut hvilken sykepleier som skal ha meldingen, og sender meldingen videre til denne sykepleieren. Det er da to muligheter som vil føre til at prosessen med å finne en sykepleier gjentar seg. Det ene er at sykepleieren ikke svarer på forespørselen. Etter en gitt tid uten svar,

finner da servern neste sykepleier som den kan sende beskjeden til. Det er også en mulighet for at sykepleieren av en eller annen grunn, som ikke er registrert hos serveren, er opptatt med andre gjøremål, og derfor må avvise forespørselen. Dette gir også det resultatet at serveren igjen må finne en annen sykepleier å sende meldingen til. Dersom sykepleieren aksepterer forespørselen, sender han/hun en melding tilbake til pasienten. Dette kan være en egendefinert melding, eller forhåndsdefinerte meldinger som f.eks. "Er på vei", "Kommer om fem min" e.l. Når sykepleieren har gjort seg ferdig med en forespørsel, sendes det en beskjed til serveren om at jobben er gjort. I tillegg til det som blir vist i dette diagrammet, må også statusoppdateringer bli sendt til andre sykepleiere i samme gruppe som den som håndterer en forespørsel.

MSC LoggAv



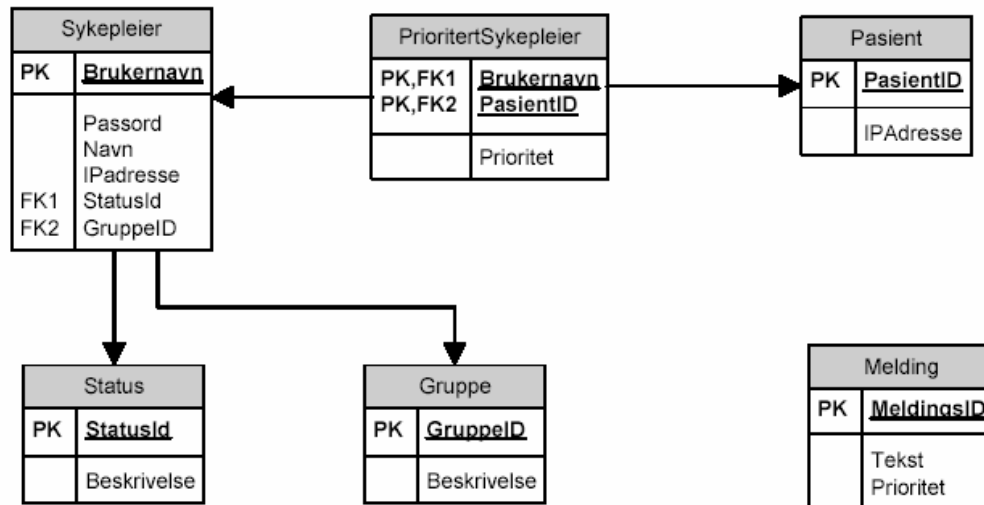
Figur 3-11 MSC Logg av

Beskrivelse av LoggAv:

Dette diagrammet er også enkelt. Sykepleier a sender en LogOff melding til serveren, som oppdaterer denne sykepleierens status i databasen, og sender ut oppdatert status til de andre sykepleierne i samme gruppe.

3.2.3 Data i systemet

ER-modellen vist i figur 3-12 illustrerer dataene i systemet:



Figur 3-12 ER-modell over data i systemet

Sykepleier

Inneholder liste over alle sykepleiere som bruker systemet, samt deres passord og fulle navn. I tillegg er også IP-adressen til sykepleierens håndholdte datamaskin lagret.

Pasient

Liste over pasienter. Inneholder i tillegg IP-adresse som kopler en pasient til en pasientterminal.

PrioritertSykepleier

Ofte er noen sykepleiere tildelt "egne" pasienter. Denne tabellen tar seg av denne koplingen. Når en pasient kommer med en forespørsel vil systemet først undersøke om det finnes noen prioritert sykepleier for denne pasienten.

Status

Liste over forskjellige statusmeldinger. Tanken er at nye statusmeldinger kan legges til i denne tabellen og deretter distribueres ut til alle brukere av systemet.

Gruppe

Hver sykepleier sorterer som regel under en fast avdeling eller sengepost. Denne inndelingen registreres i denne tabellen, slik at hver klient kun viser en liste over andre brukere som tilhører samme gruppe.

Melding

Pasienter kan komme med ulike forespørsler, og disse vil ha forskjellig prioritet. Meldingen ”er tørst” har for eksempel lavere prioritet enn meldingen ”må på do”. Denne tabellen inneholder derfor en liste over de ulike meldingene, og deres tilhørende prioritet.

3.3 Systemkrav

Denne delen skal gi en mer teknisk beskrivelse av kravene til systemet. Dokumentet tar for seg funksjoner, skjermbilder og beskriver sammenhengen mellom klassene som er skrevet. Dokumentet tar også for seg hvilke forbedringer og endringer sykepleiere kommer til å få ved å bruke programmet. Disse kravene skal være detaljert nok til at det kan brukes som utgangspunkt for implementeringen.

3.3.1 Rammekrav

Systemtilgjengelighet

Systemet er eneste bindeledd mellom pasient og sykepleier. Skulle det oppstå en kritisk situasjon hos en pasient samtidig som systemet er utilgjengelig, kan dette få katastrofale følger. Tilgjengeligheten og påliteligheten til systemet er derfor en svært kritisk faktor og nedetiden må være tilnærmet lik null.

Krav til systemtilgjengelighet:

- Nedetid mindre enn 0,00002 % (mindre enn 5 min / år)
- Drift 24 timer i døgnet - 7 dager i uken

Sikring mot tap eller ødeleggelse av data

- Det skal gjennomføres jevnlig sikkerhetskopier av systemet for å minimere tap av data ved en skade
- Systemet skal ha en såpass stor grad av redundans at systemfeil på deler av systemet ikke skal føre til driftsstans for hele systemet

Sikring mot tyveri eller misbruk av data

- Server skal være plassert i eget låsbart rom, sikret mot uvedkommende

Adgangskontroll, personverntiltak

- Alle brukere skal autentiseres med brukernavn og passord

Kapasitet

- Systemet skal kunne takle opp til 50 brukere på en avdeling
- Hver bruker skal kunne sende 1 melding hvert tiende sekund

Fremtidig utvidelse av systemet

- Systemet skal kunne samarbeide med andre avdelinger/sengeposter. Dette for å kunne flytte ressurser mellom avdelinger etter behov, spesielt nattestid og andre tidspunkter med lav bemanning.
- Systemet skal operere opp mot Elektronisk Pasient Journal (EPJ), slik at utførte oppdrag kan føres direkte inn i denne.

Tidsfaktorer

- Det skal maksimalt ta 5 sekunder fra en pasient gir signal til sykepleier mottar melding
- Svarer ikke sykepleier på meldingen innen 30 sekunder skal den videresendes til sykepleier med nest høyest prioritet

Brukervennlighet

- Systemet skal utvikles med tanke på at brukerne ikke innehar noen særlig it-kompetanse fra før. Dette stiller store krav til enkelthet og brukervennlighet i systemet.

Arbeidslokale og miljø

- Systemets komponenter skal må ikke være slik at de utsetter brukerne for ugunstige arbeidsstillinger

Nøyaktighet

- Systemet skal ha en svært høy grad av nøyaktighet. Brukerne skal kunne stole 100 % på at meldinger i systemet er riktige og autentiske.

3.3.2 Systemets funksjonelle egenskaper

Under følger krav til funksjonelle egenskaper for hver fysisk del av systemet.

Håndholdte terminaler

- Sykepleier skal kunne logge seg på med brukernavn og passord
- Sykepleier skal kunne sende og motta meldinger til/fra pasienter
- Sykepleier skal motta oppdatert status om de andre sykepleiere i sin gruppe
- Sykepleier skal kunne kommunisere med andre sykepleiere
- Sykepleier skal manuelt kunne endre sin egen status – for eksempel til ”på møte”
- Sykepleier skal kunne logge av systemet

Terminal for administrasjon

- Brukeradministrasjon - legge til, fjerne eller endre brukere
- Gruppeadministrasjon - opprette, fjerne, endre grupper og flytte personell mellom grupper
- Fordeling av ansvar for bestemte pasienter til sykepleiere. Herunder sette opp prioriterte lister over foretrukne sykepleiere

Server

- Fungere som meldingsruter - videresende meldinger mellom brukere av systemet
- Stå for arbeidsplanlegging – fordele oppgaver basert på informasjon om ledige ressurser, prioritering av sykepleiere og prioritering av oppgaver

4 Implementering

Vi har ikke hatt tid til å implementere systemet i sin helhet, men vi har laget en prototyp for å demonstrere hvordan vi tenker oss systemet. Denne prototypen er derfor veldig enkel, og langt fra alle kravene som er spesifisert i beskrivelsen av systemet er oppfylt.

Klassediagrammer (se vedlegg B) og kildekode til prototypen (se vedlagt CD) utfyller denne fagrapporten. Mesteparten av funksjonaliteten er implementert i prototypen, men vi har sett bort fra en del krav til sikkerhet og pålitelighet.

4.1 Prototyp

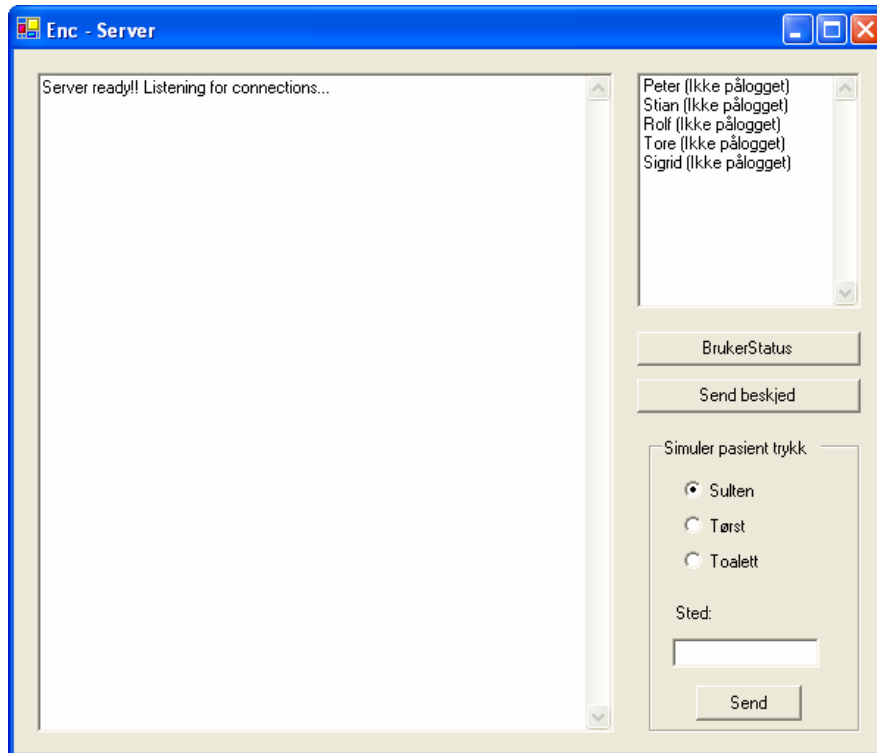
Prototypen er implementert i språket C# ved hjelp av utviklingsmiljøet Microsoft Visual Studio .NET. Grunnen til at vi valgte nettopp denne teknologien er den integrerte støtten for å lage programmer for små, håndholdte enheter. En innebygd emulator i Visual Studio gjør det mulig å kjøre programmer ment for PDA på en vanlig PC.

Kommunikasjon mellom server og klient skjer gjennom utveksling av xml-meldinger.

4.1.1 Server

Serverdelen av prototypen består i dag av et enkelt brukergrensesnitt med underliggende funksjonalitet som tillater pålogging av flere klienter, samtidig som den er laget for å kjøre på en datamaskin med nettilgang. Følgende funksjoner er implementert:

- Autentisering av klienter
- Routing av beskjeder mellom klienter
- Sende beskjed fra server til klient
- Simulere forespørsel fra pasient



Figur 4-1 Server prototypens brukergrensesnitt

Autentisering av klienter

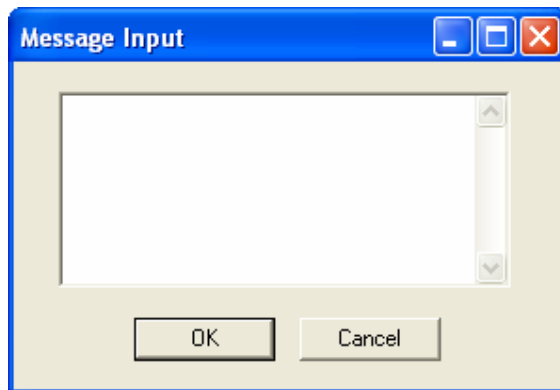
Alle klientene må autentisere seg mot en brukerdatabase for å få logge seg på. Dette gjøres ved at hver enkelt bruker presenterer brukernavn og passord til serveren. Når en klient har presentert riktig brukernavn og passord, vil hans status bli oppdatert, og brukeren regnes som pålogget i systemet. Selve brukerdatabaseen er i prototypen lagret som en enkel xml-fil. Mer informasjon om hvordan denne xml-filen er bygget opp finnes i vedlegg C. Statusen til alle brukerne i systemet vises i en liste øverst til høyre i brukergrensesnittet.

Routing av beskjeder

Når klientene er logget på, kan de sende melding til hverandre via serveren. Når serveren mottar en beskjed fra en av klientene, slår den opp i beskjeden og ser hvem beskjeden er til. Serveren sjekker så om vedkommende er logget på systemet, og sender så beskjeden videre til rett klient. Dersom personen ikke er logget på, forkastes bare beskjeden slik som prototypen er i dag. Formatet på beskjedene som sendes er nærmere beskrevet i vedlegget om xml (vedlegg C).

Sende beskjed til klient

I listen som viser statusen til brukerne i systemet kan man på serveren velge en person som har status som tilsier at han er logget på systemet og trykke på knappen Send Beskjed. Det vil så komme opp en dialog lik figuren nedenfor der man kan skrive inn beskjeden og trykk OK for å sende beskjeden og Cancel for å avbryte.



Figur 4-2 Send beskjed dialogen

Simulere forespørsel fra pasient

Denne funksjonen er laget for å simulere at en pasient trenger hjelp til noe. Man velger hva pasienten angivelig ønsker hjelp til ved å trykke på en av flere radioknapper, og så skrive lokasjon for pasienten inn en tekstboks. Både radioknappene og tekstboksen finnes i ”Simulere pasient trykk” panelet nede til høyre i brukergrensesnittet.

4.1.2 Klient

Klientdelen av prototypen er en applikasjon laget for å kjøre på en liten, håndholdt enhet (f.eks. en PDA) med trådløs nettilgang.

Klienten har implementert følgende funksjoner:

- Logge av / på systemet
- Vise liste over andre brukere og deres status
- Forandre egen status (LEDIG / OPPTATT / MØTE osv)
- Sende og motta meldinger til andre brukere
- Motta forespørsler fra pasienter og svare på disse

Figur 4-3 Skjermbilde fra prototypen viser et skjermbilde hvor brukeren ”ROLF” er logget på systemet med status ”LEDIG”. Resten av skjermbildet består av en liste over andre brukere og deres status:



Figur 4-3 Skjermbilde fra prototypen



Figur 4-4 Skjerm bilde som viser kommunikasjon med annen bruker

Her følger en kort beskrivelse av funksjonene som er implementert. For øvrig henvises det til vedlegg A for en mer detaljert beskrivelse.

Vise liste over andre brukere og deres status

Denne funksjonen gir brukeren en oversikt over annet personell som tilhører samme gruppe som brukeren. I tillegg vises også status for andre brukere.

Forandre egen status (LEDIG / OPPTATT / MØTE)

Bruker har mulighet til å forandre egen status. Det kan velges mellom ”Ledig”, ”Opptatt” eller ”Møte”. Den nye statusen sendes så ut til andre brukere av systemet. Ved en utvidelse av systemet er det lett å legge til flere statuser.

Sende og motta meldinger til andre brukere

I prototypen er det også implementert en funksjon for å kunne sende og motta personlige meldinger til andre brukere. Funksjonaliteten minner mye om Microsoft Messenger, et mye brukt program for slik kommunikasjon.

Motta forespørsler fra pasienter og svare på disse

Denne funksjonen er kun implementert i en enkel utgave. Ved forespørsler fra en pasient vises et vindu hos brukeren med informasjon om hvem som står bak forespørselen, hvor denne personen befinner seg, og hva forespørselen går ut på.

4.2 Videre utvikling

Prototypen vi har utviklet inneholder på langt nær alle funksjoner vi har tiltenkt det ferdige systemet. Her vil vi liste opp en del forslag til nye funksjoner vi mener systemet bør ha.

4.2.1 Terminal for administrasjon

Vi ser for oss at systemet administreres fra en stasjonær terminal. Det vil være naturlig med en terminal for hver avdeling. Fra denne terminalen kan nye brukere legges til og fjernes fra systemet. Her kan det også settes opp prioriteringslister hvis en sykepleier er tildelt en eller flere pasienter.

4.2.2 Grupper

Brukere av systemet bør organiseres i grupper, og det vil være naturlig å opprette en gruppe for hver avdeling. I tillegg kan det opprettes egne grupper for brukere med spesiell kompetanse, som for eksempel spesialsykepleiere.

4.2.3 "Smart" server som velger sykepleier

Det ferdige systemet bør utvides slik at det på en "intelligent" måte kan lete opp ledige ressurser når det kommer forespørsler fra pasienter.

Regler

Det vil være naturlig å implementere en slik server ved å bruke et sett med regler. En slik regel kan for eksempel være "finn nærmeste ledige sykepleier". Reglene bestemmer også hva som skal skje hvis en sykepleier ikke svarer på en forespørsel, for eksempel "hvis valgte sykepleier ikke svarer på forespørsel innen 30 sekunder, finn neste sykepleier". Disse reglene

kan også defineres slik at ressurser kan deles mellom avdelinger. Dette vil for eksempel være spesielt aktuelt nattetid med mindre bemanning på avdelingene.

4.2.4 Geografisk lokalisering av sykepleiere

Ved større avdelinger vil det være en fordel hvis systemet har oversikt over den geografiske plasseringen til de forskjellige sykepleierne. Da har systemet mulighet til å velge den sykepleieren som har kortest avstand til en pasient, og dermed effektivisere arbeidet ved en avdeling. Teknologi som muliggjør en slik lokalisering leveres blant annet av firmaet Radionor. Disse har utviklet et ”radioøye” [2] som kan ”se” hvor brukeren av et trådløst nett befinner seg.

4.2.5 Kryptering

Meldinger som sendes over systemet vil kunne inneholde sensitive opplysninger, for eksempel om pasienter. Datatrafikken mellom håndholdte enheter og serveren bør derfor være kryptert.

5 Diskusjon og konklusjon

Systemer for tilkalling av sykepleiere på sykehus har tradisjonelt basert seg på varslingsknapper ved hver sykehusseng og lamper ute i gangen. Vi har i denne rapporten sett på ulike sider ved å innføre et nytt, alternativt system for slik varsling. Dette systemet er basert på datateknologi gjennom bruk av små, håndholdte enheter (PDAer) med trådløs nettilgang. Brukt på riktig måte skulle alt ligge til rette for at denne teknologien vil gi økt effektivitet og en bedre utnyttelse av personellet ved et sykehus.

Det nye sykehuset som er under bygging i Trondheim, vil i en mye større grad benytte seg av IKT-teknologi enn dagens sykehus. Blant annet skal hver seng utstyres med en egen pasientterminal. En utførlig liste over krav til kommunikasjon i det nye sykehuset finnes i [5]. Cardiac Medical A/S skal levere terminaler, som skal kunne vise pasientinformasjon, data fra ulike måleapparater, samt være et underholdningssenter for pasienter [4].

Vi vil her se på noen aspekter ved innføring av et slikt system:

5.1 Sikkerhet og konfidensialitet

I et sykehusmiljø befinner det seg mye sensitiv informasjon. Derfor er det viktig å grundig vurdere hvordan denne informasjonen best kan sikres slik at den ikke kommer på avveie. Måter å sikre informasjonen på kan for eksempel være kryptering av trafikk og benytte seg av teknologi som sikrer at brukere av trådløst nettverk må befinne seg inne i selve sykehusbygningen.

Aspekter rundt det å ha et system som til enhver tid har oversikt over hvor hver sykepleier er, er diskutert i [3]. En av problemstillingene som tas opp her er at selv om ikke alle andre brukere har tilgang til informasjon om hvor en bruker befinner seg, kan slik informasjon misbrukes av en administrator av systemet.

5.2 Skepsis mot nye systemer

En ting vi la merke til ved våre befaringer på St.Olavs Hospital, var skepsisen mot å innføre et helt nytt system for tilkalling av sykepleiere. I de tilfellene hvor de ansatte kunne tenke seg et nytt system, ønsket de som regel kun å oppgradere til nye systemer som var i bruk ved andre avdelinger på sykehuset.

6 Referanser

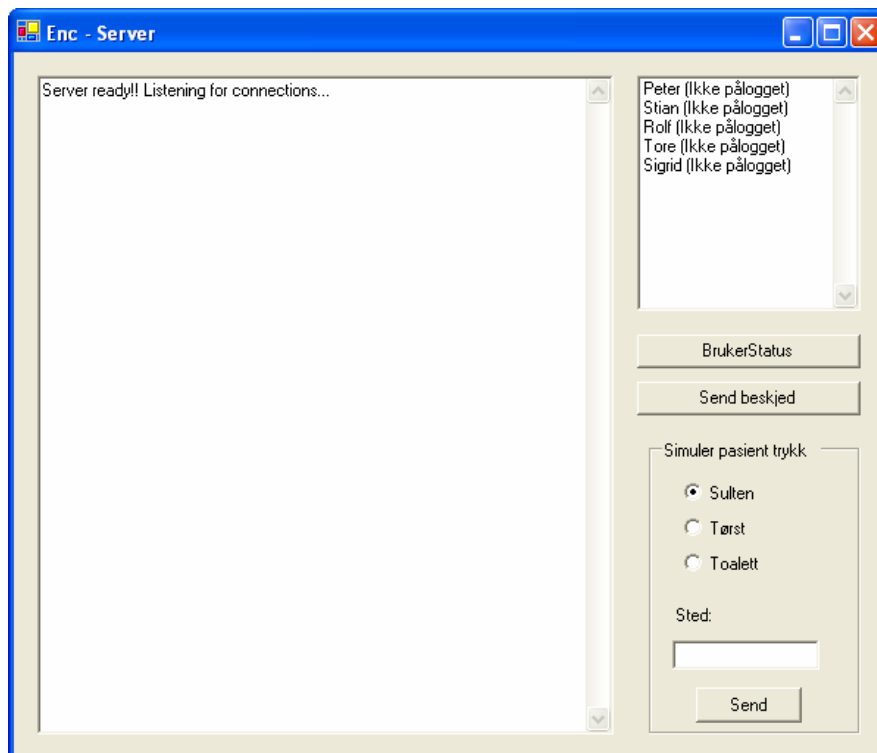
- [1] Miller, E., Deets, C. & Miller, R., (1997): "Nurse Call Systems: Impact on Nursing Performance", Aspen Publisher, Inc.
- [2] Radionor: Technology, www.radionor.no, aksessert 20.04.2004.
- [3] Marte Forthun (2003): "Group communication for healthcare workers designed in ServiceFrame", Prosjekt, ITEM/NTNU
- [4] Cardiac Medical: Medical IT, www.cardiac.no, aksessert 20.04.2004
- [5] Interconsult (2003): "Tilbudsbeskrivelse for Kommunikasjonsanlegg / infrastruktur for IKT", Helsebygg Midt-Norge – nye St. Olavs Hospital

Vedlegg A - Brukerveiledning for prototypen

A.1 Server

Serveren kan kjøres på en vanlig PC med nettilgang som har installert .NET framework.

Når man starter opp serveren får man opp ett skjermbilde likt det i figur A-1. Til venstre har man status vinduet som viser hvilke beskjeder serveren til en hver tid mottar. Øverst til høyre har man listen over brukerne og deres status. Nedenfor denne listen finner man funksjonene som man kan bruke



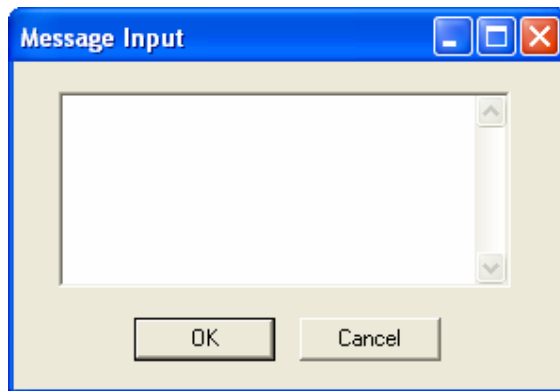
Figur A-1: Skjermbilde av oppstart av serveren

A.1.1 Bruker Status

Denne funksjonen er først og fremst beregnet som en hjelpefunksjon under utviklingen av programmet. Når man trykker på denne knappen skrives all informasjon om brukerne i systemet ut til status vinduet.

A.1.2 Sende beskjed

Brukeren kan sende beskjeder til klientene som er logget på serveren. Man velger er bruker i brukerlisten som har en status som tilsier at han er pålogget systemet. Dvs. de med annen status enn ”Ikke pålogget ” Man trykker så på knappen Send Beskjed og en dialog lik figur A-2 dukker opp. Brukeren skriver inn beskjeden og trykker OK for å sende den, eller Cancel for å avbryte.



Figur A-2: Skjerm bilde av Send Beskjed dialog

A.1.3 Simulere pasient tilkalling

Denne funksjonen er laget for å simulere at en pasient har trykket på pasient terminalen og trenger hjelp til noe. Man velger hva pasienten trenger hjelp til ved hjelp av radioknappene, og skriver til slutt inn lokasjonen til pasienten. Til slutt trykker man på Send knappen for å sende forespørselen om hjelp til valgt bruker. Man må på samme måte som når man skal sende en beskjed til en bruker, også her velge hvem man skal sende forespørselen til.

A.2 Klient

Klienten kan kjøres på en PDA med trådløs nettilgang eller i en emulator på en vanlig PC.

A.2.1 Logge inn

Når klienten startes opp vises skjermbildet påloggingsvinduet - se figur A-3:



Figur A-3: Skjermbilde som viser påloggingsvindu

Bruker angir brukeravn og passord og velger "Logg inn". Disse opplysningene kontrolleres opp mot server. Er brukernavn og/eller passord feil vil det komme en dialogboks med beskjed om dette, og brukeren får mulighet til å prøve på nytt.

A.2.2 Hovedvindu

Er brukernavn og passord korrekte, vil skjermbildet som vist i figur A-4 under, dukke opp: Øverst i skjermbildet kan vi se at brukeren "ROLF" er logget inn. Hans nåværende status er "LEDIG". Midt i skjermbildet vises en liste over andre brukere, i dette tilfellet 4 brukere.

Brukeren har nå flere valg:

- Endre egen status (se avsnitt A.1.3)
- Sende melding til andre brukere (se avsnitt A.1.4)
- Logge av (se avsnitt A.1.5)
- Avslutte programmet (se avsnitt A.1.6)

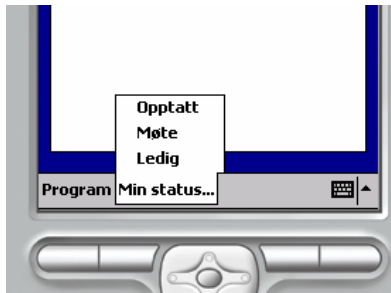
I tillegg kan bruker svare på pasientforespørsler (se avsnitt A.1.7)



Figur A-4: Skjerm bilde som viser hovedvinduet

A.2.3 Endre egen status

For å endre sin egen status, kan dette gjøres ved å klikke på “MIN STATUS...” nederst på skjermen. En meny som vist i figur A-5 vil da dukke opp:

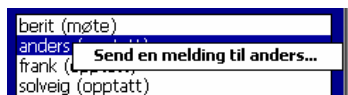


Figur A-5: Utsnitt av skjermbilde som viser statusmenyen

Når bruker har valgt ny status sendes en melding om dette til server samt alle andre brukere.

A.2.4 Sende melding til andre brukere

For å sende melding til en annen bruker kreves det at denne brukeren er pålogget og vises i listen midt i hovedvinduet som vist i figur A-4. For å sende melding, klikk på en bruker og velg ”Send melding til bruker...” som vist i figur A-6:



Figur A-6: Skjermbilde av send melding til andre brukere

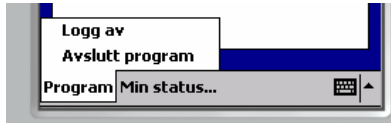
Skjermbildet som vist i figur A-7 vil da dukke opp:



Figur A-7: Skjermbilde som viser samtale med annen bruker

A.2.5 Logge av

For å logge av systemet, velg ”LOGG AV” i menyen under ”PROGRAM” nederst på skjermen. Dette er vist i figur A-8:



Figur A-8: Utsnitt av skjermbilde som viser Programmenyen

Programmet vil da returnere til påloggingsvinduet.

A.1.7 Avslutte programmet

For å avslutte, velg ”AVLUTT PROGRAM” i menyen under ”PROGRAM” nederst på skjermen (se figur A-8).

A.2.8 Svare på pasientforespørsler

Ved forespørsler fra pasienter vil skjermbildet som vist i figur A-9 dukke opp:

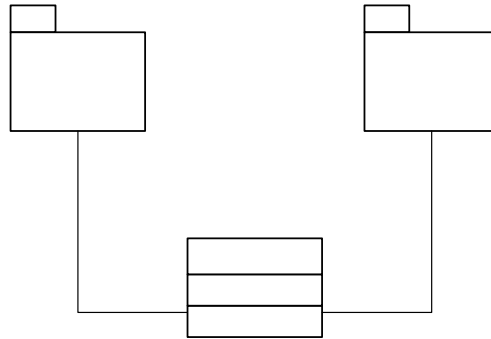


Figur A-9: Skjermbilde som viser forespørsel fra pasient

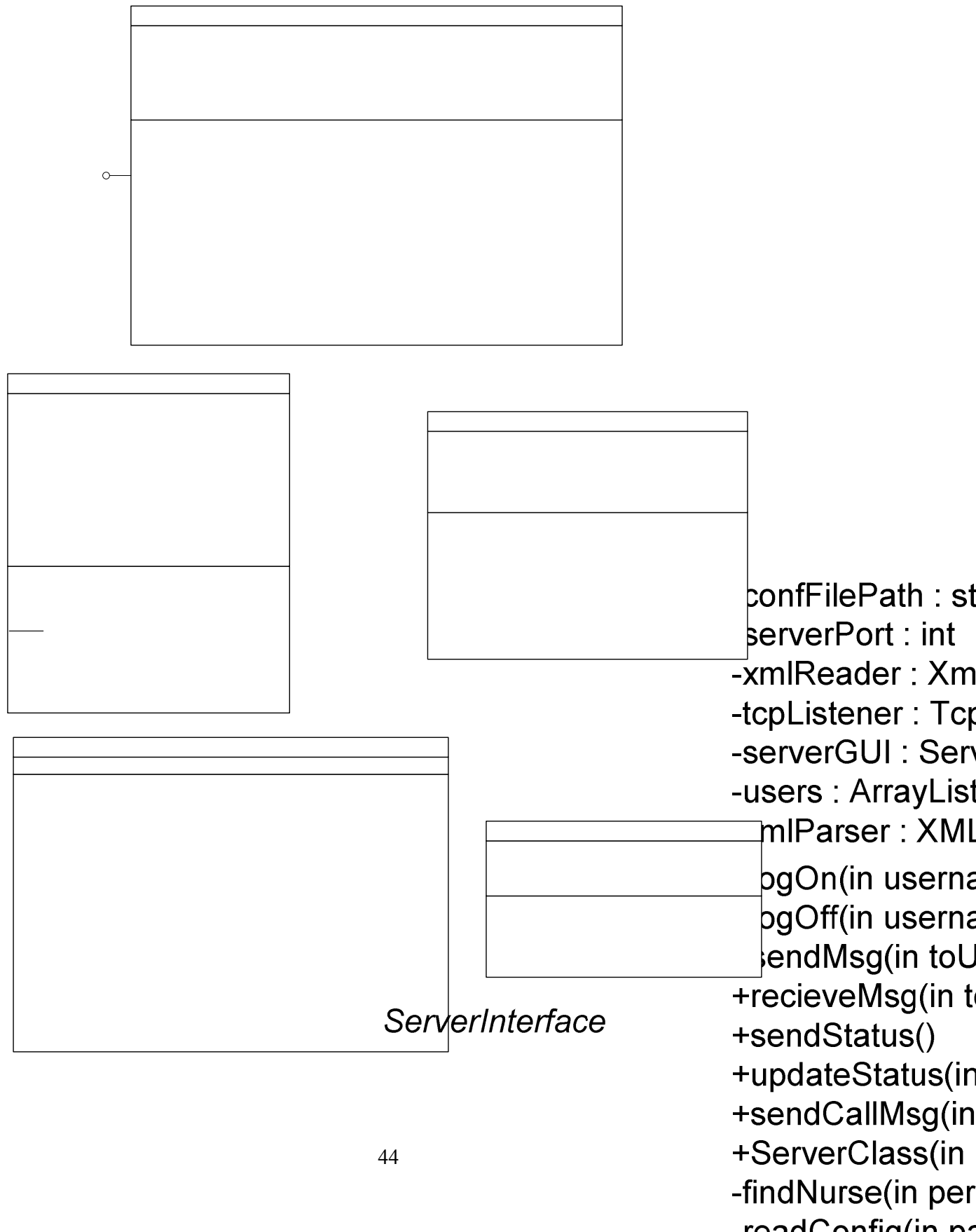
Velg ”Godta oppdrag” eller ”Avslå oppdrag” og vinduet vil lukkes.

Vedlegg B – UML Klassediagram prototyp

Topp-nivå



Pakke: Server



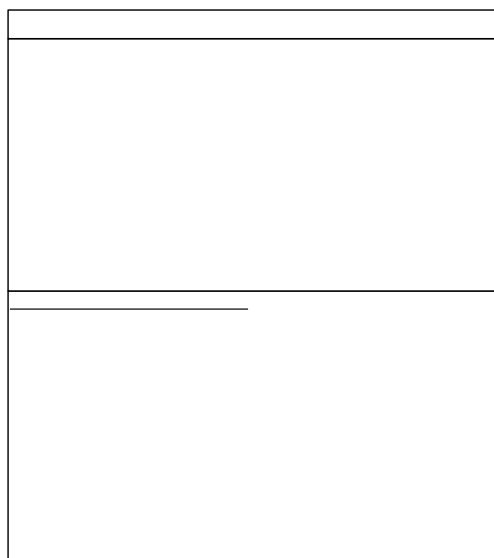
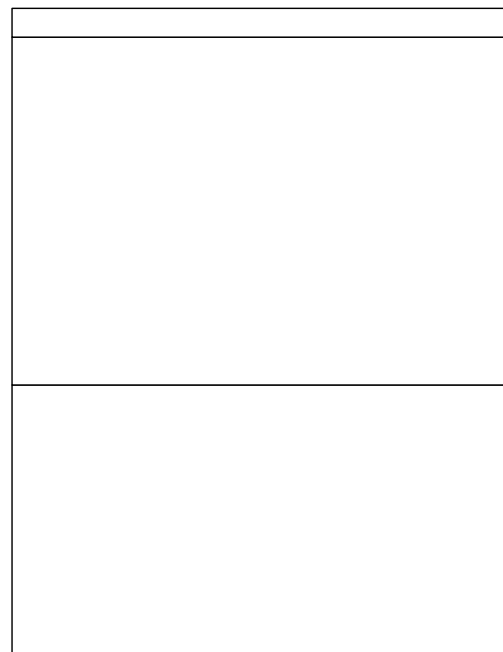
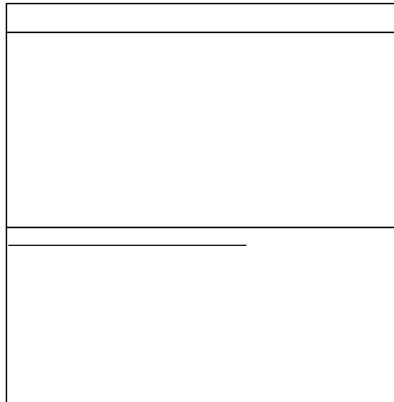
Pakke: Client



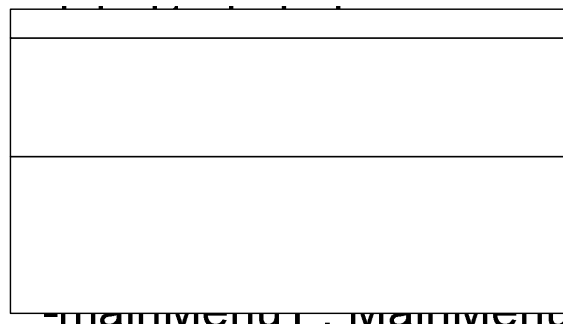
Client

+CONNECT : bool = true
 +DISPLAY_MESSAGES : bool = false
 -socket : TcpClient
 -networkStream : NetworkStream
 -streamWriter : StreamWriter
 -streamReader : StreamReader
 -listeningThread : Thread
 -xmlParser : XMLParser
 -loginForm : LoginForm
 -mainForm : MainForm
 -taskForm : TaskForm
 -msgForm : SendMessageForm
 -myStatus : string = ""
 -myUserName : string = ""

Pakke: ClientGUI



ClientGUI::LoginForm



-mainMenu : MainMenu
-loginButton : Button
-menuItem1 : MenuItem
-menuItem2 : MenuItem
-client : Client

-ShowIM(in dwFlag : int) : bool

+LoginForm(in client : Client)

+displayUserName(in userName : string)

+displayKeyboard()

⁴⁶+hideKeyboard()

+LoginClick(in sender : object, in e : EventArgs)

Vedlegg C – XML-meldinger og konfigurasjonsfiler

I prototypen er alle meldinger som sendes mellom klient og server på xml-format. I tillegg så er en enkel konfigurasjonsfil for serveren implementert i xml og også den midlertidige brukerdatabase. Dette vedlegger viser og beskriver hvordan filene og meldingene er bygd opp.

C.1 XML-filer

C.1.1 Konfigurasjonsfil for server

Serverens konfigurasjonsfil er veldig enkel. Den starter med tag'en serverconfig og inneholder 1 parameter: port. Denne parameteren angir hvilken port serveren skal lytte etter innkommende forbindelser på.

```
<serverconfig>
  <port>3333</port>
</serverconfig>
```

Figur C-1 XML-fil - Konfigurasjonsfil for server

C.1.2 Brukerdatabase

Brukerdatabasen starter med tag'en Userdatabase og for hver bruker som er lagt til begynner med en tag: user. User har 2 argumenter: username og password som er brukernavnet og passordet til brukeren. Så kommer det en ny tag: name som er navnet til brukeren.

```
<Userdatabase>
  <user username="brukernavn" password="passord">
    <name>navn</name>
  </user>
  <user username="brukernavn" password="passord">
    <name>navn</name>
  </user>
  <user username="brukernavn" password="passord">
    <name>navn</name>
  </Userdatabase>
```

Figur C-2 XML-fil - Brukerdatabase

C.2 XML-meldinger

Oppbyggingen av alle xml-meldinger er alle sammen på en standard form som vist under. Denne standarden er også brukt av Gruppe3 som lager pasientdelen av systemet. På denne måten vil det være lett og koble de to systemene sammen.

```
<msg type="meldingstype">  
  <parameter1>verdi</parameter1>  
  <parameter2>verdi</parameter2>  
</msg>
```

Figur C-1 Mal for XML-beskjed

Første argument: `<msg type="meldingstype">` angir hvilken type melding som kommer. Dette feltet brukes av xml-parseren for å finne ut hva som skal gjøres når en melding av en bestemt type er mottatt. Innenfor `<msg>` kan man så spesifisere så mange forskjellige parametere som man måtte ønske.

C.2.1 Logon

En logon beskjed er på formatet som er vist under. Type beskjed er satt til logon, og etter det kommer parametrene username, password og IP. Verdiene på de respektive feltene er brukernavnet, passordet og IP til klienten som brukeren forsøker å logge seg på fra

```
<msg type="logon">  
  <username>brukernavn</username>  
  <password>passord</password>  
  <ip>ipadresse</ip>  
</msg>
```

Figur C-2 XML-beskjed – Logon

C.2.2 Logon respons

Når serveren har mottatt en beskjed av typen logon sjekker den om brukernavn og passord stemmer mot brukerdata-basen og sender en beskjed tilbake. Dersom brukernavn og/eller passord var ugyldig sendes en beskjed av type logon_fail til klienten. Hvis riktig brukernavn og passord ble presentert i logon beskjeden sendes en logon_ok beskjed tilbake til klienten

```
<msg type="logon_fail" />
```

Figur C-3 XML-beskjed - logon_fail

```
<msg type="logon_ok" />
```

Figur C-4 XML-beskjed - logon_ok

C.2.3 Status

Når er bruker skifter status sender han en beskjed til serveren. Serveren oppdaterer brukerens status og sender statusbeskjeder til alle klientene som er logget på slik at de kan oppdatere statusen til brukerne. Meldingstype er satt til "status" og meldingen inneholder 2 parametere: username og status. Disse angir henholdsvis hvilken bruker som har oppdatert sin status, og hvilken status brukeren nå har. Det er den samme typen som sendes fra klient til server som det som sendes fra server til klient.

```
<msg type="status">  
  <username>"brukernavn"</username>  
  <status>status</status>  
</msg>
```

Figur C-5 XML-beskjed – status

C.2.4 Tekstbeskjed

Systemet tillater at brukerne sender tekstbeskjeder til hverandre. Melding typen er da satt til "text" og den ser ut som på figur C-8. Beskjeden inneholder de 3 parametrene to, from og text. To feltet angir brukernavnet til personen som meldingen skal til, from feltet brukernavnet til personen som melding kommer fra, og text feltet inneholder selve tekst beskjeden.

```
<msg type="text">
  <to>sykepleier1</to>
  <from>sykepleier2</from>
  <text>meldingstekst</text>
</msg>
```

Figur C-6 XML-beskjed – tekstbeskjed

C.2.5 Pasient tilkalling

På serveren kan man simulere at en pasient trenger hjelp til ett eller annet. Det sendes da en beskjed av typen "call" til sykepleieren. Meldingen inneholder parametrene pasient, location og call_type. Disse feltene gir pasientens id, lokasjon og informasjon om hva han trenger hjelp til

```
<msg type="call">
  <pasient>pasientid</pasient>
  <location>lokasjon</location>
  <call_type>tilkallingstype</call_type>
</msg>
```

Figur C-7 XML-beskjed - Pasient tilkalling

C.2.6 Svar på pasienttilkalling

Når sykepleieren mottar en tilkalling velger han om han godtar oppdraget eller avslår det. Det sendes da en melding av typen "call_respons" tilbake til serveren. Den inneholder parametrene pasient og respons. De respektive parametrene er pasientid til pasienten som han mottok meldingen fra og responsen på forespørselen om hjelp. Altså om han godtok oppdraget eller ikke.

```
<msg type="call_response">
  <pasient>pasientid</pasient>
  <response>respons</response>
</msg>
```

Figur C-8 XML-beskjed - Svar på pasienttilkalling

Vedlegg D - CD med kildekode