

## INSTITUTT FOR ELEKTRONISKE SYSTEMER

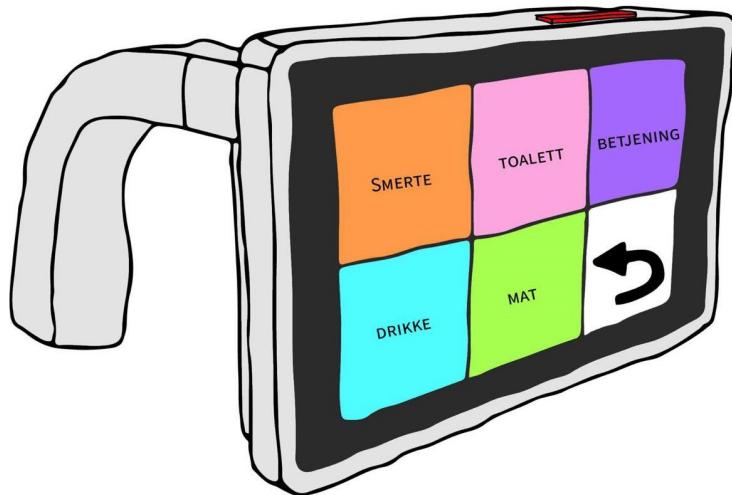
TTT4270 - ELEKTRONISK SYSTEMDESIGN, PROSJEKT

---

# Teknisk rapport - Gruppe 15

---

## VarselVenn



*Gruppe 15:*

Daniel Dalen, Nicole Maria Powell, Eivind Lillefosse, Palina Yermakova, Amalie Fridfeldt Hauge

Dato: 11.01.2024

---

# Forord

Studentene som studerer elektronisk systemdesign og innovasjon (ELSYS) ved NTNU har et fag som kalles; Elektronisk Systemdesign Prosjekt. I dette faget er hovedoppgaven et innovasjonsprosjekt. Innovasjonsprosjektet går ut på at studentene i samarbeid med en bedrift, skal løse en utfordring de har, på en innovativ måte. Årets innovasjonsprosjekt er i samarbeid med St. Olavs Hospital.

Denne sluttrapporten vil ta for seg et innovasjonsprosjekt i samarbeid med avdeling Ortopedisk Sykepleie.

## Sammendrag

Denne rapporten omhandler produktet *VarselVenn*. Problemstillingen som produktet skal tilfredsstille er:

*”Hvordan effektivisere og forenkle kommunikasjonen mellom sykepleiere og pasienten? ”*

Denne problemstillingen ble valgt på bakgrunn av en behovsanalyse som ble gjennomført på ortopedisk avdeling ved St. Olavs Hospital, der de ønsket en måte å differensiere varslingssignalene fra pasienten. Dagens varslingssystem består av å trekke i en rød snor eller trykke på en knapp, som tilkaller en sykepleier.

*VarselVenn* består av to systemer, en brukerenhet og en styreenhet. Brukerenheten er en bærbar enhet som kan utplasseres på rommene til pasientene og gir pasienten muligheten til å varsle sykepleierne igjennom en touch-skjerm med valg over spesifiserte varsel. Brukerenheten kan også også tilpasses til ulike grader av teknologiforståelse, den enkleste modulen har bare to varsel valg, mens den mest avanserte har 5 hovedvalg med flere undermenyer. Denne tilpasningen kan enkelt gjøres av sykepleierne igjennom en egen sykepleiermodul som aktiveres benyttelse av sykepleierne sine RFID-kort.

Det andre systemet, styreenheten, er plassert på tunet på avdelingen og er sykepleierne sin oversikts-platform over alle varsel som blir sendt fra brukerenhetene som er plassert på de ulike rommene. På styreenheten vil sykepleierne lett få en oversikt over varslene som er sendt fra de ulike rommene, både igjennom en liste over alle varslene som er sortert etter trafikklys inspirert hastegrad, og et kart som gir en rask oversikt over hvilke rom som trenger hjelp. Styreenheten gir også muligheten til å ha en liste med bare mat og drikke relaterte varsel, og mulighet utsending at matvarsel til alle brukerenheter som ikke er markert med fastende pasienter. Denne delen av styreenheten er designet for å avlaste sykepleierne med oppgaver kjøkkenpersonell kan utføre. Styreenheten og brukerenhetene vil kommunisere trådløst med hverandre over Bluetooth.

Til verifiseringen av produktet ble det først opprettet brukerkarav basert på behovsanalsisen for avdelingen, deretter ble det utarbeidet en rekke med systemkrav som

---

baserer seg på brukerkaravene og som setter klare grenser på hva som er forventet oppførsel fra *VarselVenn*. Disse systemkravene ble i hovedsak oppfylt eller delvis oppfylt under implementeringen.

Den implementerte brukerenheten har da tre ulike moduler som gir varierende valgmuligheter til sende varsel, en egen SOS-knapp som er fysisk og uavhengig av hva som er på touch-skjermen, en sykepleiermodul som åpnes ved å skanne et godkjent RFID-kort. I tillegg har den en batteritid ved aktiv bruk på minst 5 timer, og holder minst 24 timer ved vanlig bruk. De kravene som kun delvis ble oppnådd var; at brukerenheten kan ikke kan desinfiseres med alkohol ettersom at brukerenheten ble 3d-printet i PLA som ikke tåler alkohol i den grad som nyttes på sykehus. I tillegg ble ikke kravet om fall fra en meters høyde verifisert i full grad, da bare en halv meter ble testet.

For styrenheten ble følgende krav godkjent; sorterte varsel fra brukerenhetene, og toveis-kommunikasjon med brukerenheten er mulig. Styreenheten har også mulighet til å kun vise mat og drikke relaterte varsel, og kan sende matvarsel om bestilling av ulike måltider til brukerenhetene. Her tar den også hensyn til hvilke rom som er markert med fastende pasienter.

Det totale systemet er et velfungerende system med mange muligheter for differensiering av varsel allerede, men det er fortsatt ting som bør videreutvikles dersom produktet skal komme nærmere et kommersielt produkt. En stor del denne videreutviklingen hadde vært å koble *VarselVenn* sammen med det eksisterende varsel-systemet på sykehuset, samt at styreenheten kunne integreres på sykepleierne sine personlige digitale assistenter (PDA). Men prototypen har allerede fått mange gode tilbakemeldinger, da spesielt fra sykepleierne på ortopedisk avdeling som ser flere mulige besparelser for avdelingen ved integrasjon av produktet. I tillegg ønsket de at produkten skulle videreutvikles til produkt klar til integrasjon på sykehuset.

Hensikten med rapporten er da å gi fullstendig oversiktlig over designet til de to systemene og bakgrunnen for valgene som er tatt. I tillegg til en oversiktlig beskrivelse for implementasjonen av produktet. Det vil til slutt bli lagt opp til verifikasjon og videre utvikling.

# Innhold

<b>Forkortelser</b>	<b>1</b>
<b>1 Introduksjon</b>	<b>1</b>
1.1 Brukerkrav .....	2
<b>2 Konsept</b>	<b>4</b>
2.1 Mulige løsninger .....	4
2.2 Brukergrensesnitt for valgt løsning .....	5
2.3 Systemkrav .....	6
<b>3 Design</b>	<b>10</b>
3.1 Styreenhet .....	10
3.1.1 Varselliste .....	11
3.1.2 Kart over tunet .....	13
3.1.3 Kjøkkenmeny .....	14
3.2 Brukerenhet .....	17
3.2.1 Modul 1 - Enkel .....	18
3.2.2 Modul 2 - Middels .....	20
3.2.3 Modul 3 - Vanskelig .....	23
3.2.4 Sykepleier meny .....	26
3.2.5 Måltider .....	30
3.3 SOS .....	33
<b>4 Implementering</b>	<b>34</b>
4.1 Komponentbeskrivelse .....	34
4.2 Kretsskjema .....	35
4.3 Kommunikasjon .....	37
4.4 Kode .....	37
4.4.1 RFID-kode .....	37
4.4.2 Bluetooth-kode .....	38
4.4.3 UI .....	40
4.4.4 Innfatning .....	51
<b>5 Verifikasjon og Validering</b>	<b>54</b>
5.1 Verifikasjon av systemkravene .....	54
5.2 Validering av brukerkrav .....	58
<b>6 Diskusjon</b>	<b>62</b>
<b>7 Konklusjon</b>	<b>65</b>
<b>Referanser</b>	<b>66</b>
<b>Tillegg</b>	<b>67</b>
A Behovsanalyse .....	67
B Tilbakemelding fra sykepleiene .....	68
C Fusion360 Plantegninger .....	71

---

## Forkortelser

<b>BK</b>	Brukerkrav
<b>ELSYS</b>	Elektronisk Systemdesign og Innovasjon
<b>HTML</b>	Hyper Text Markup Language
<b>MCU</b>	Mikrokontroller
<b>NTNU</b>	Norges teknisk-naturvitenskapelige universitet
<b>PDA</b>	Personal Digital Assistant
<b>PETG</b>	Polyetylentereftalat
<b>PLA</b>	Polymelkesyre/ Polylactic acid
<b>PP</b>	Polypropen
<b>RPi</b>	Raspberry Pi
<b>SBC</b>	Single Board Computer
<b>SPI</b>	Serial Peripheral Interface Bus
<b>UI</b>	Brukergrensesnitt

## 1 Introduksjon

Helsejenesten står ovenfor store utfordringer i tiden som kommer. Mangelen på helsepersonell, kombinert med den økende andelen eldre og kronisk syke, vil gjøre det vanskelig å ivareta dagens tilbud innen helse- og omsorgstjenestene [1]. For å sikre at pasienter får nødvendig hjelp, er det viktig å utnytte tilgjengelige ressurser, i form av syke- og vernepleiere, på best mulig måte.

For å komme frem til en problemstilling og kartlegge de spesifikke behovene i helsevesenet, ble det arrangert et besøk hos ortopedisk avdeling på St.Olavs Hospital. Besøket ga verdifull innsikt i de daglige utfordringene sykepleierne møter, og hvor det er muligheter for teknologiske forbedringer. En utfyllende oversikt over utfordringene helsevesenet står foran, er presentert i Appendix A. Ut i fra denne behovsanalysen har det blitt valgt å fokusere på varslingssystemene som benyttes av dagens sykehus, eldrehjem og innen hjemmesykepleien, med et særlig fokus på hvordan pasientene varsler helsepersonell ved behov om hjelp.

For å kartlegge behovet for et slikt varslingssystem, ble helsepersonell ved ortopedisk sengepost og hjemmesykepleierne kontaktet. Per dags dato er det primære varslings-systemet en snor, ofte plassert bak sengen og på toalettet som pasientene på sykehus eller eldrehjem trekker i hvis de trenger assistanse. Dette kan være alt fra et akutt illebefinnende til ønske om et glass vann.

---

På besøket hos ortopedisk avdeling uttalte sykepleierne at dersom flere pasienter trekker i snoren samtidig, er det ingen prioritering på hvilke pasienter som får hjelp først. Mangelen på helsepersonell fører til at det er begrenset hvor mange pasienter som kan få hjelp samtidig. Da er det mulig at pasienter som trenger nødhjelp og trekker i snoren kan risikere å ikke bli prioritert. Dette er problematisk da pasientene risikerer å ikke få den hjelpen de har et akutt behov for, samt at sykepleiere og annet helsepersonell fort blir overarbeidet. Sykepleierne uttrykker videre at enkelte pasienter har en høy terskel for å trekke i snoren, da de er redde for å forstyrre sykepleierne slik at pasienter med større behov ikke får den hjelpen de trenger.

Med bakgrunn i behovsanalysen er den valgte problemstillingen:

*"Hvordan effektivisere og forenkle kommunikasjonen mellom sykepleiere og pasienten?"*

Et bedre varslingssystem vil kunne bidra til at færre overarbeides innen helsesektoren, og slik gjøre helsetjenesten til en mer attraktiv arbeidsplass. Videre vil et bedre varslingssystem kunne bidra til at pasienter med akutte behov mottar den hjelpen de trenger innen rimelig tid. Helsevesenet vil bli effektivisert og vil dermed være bedre rustet for den kommende eldrebølgen, en av de store utfordringene for den norske velferdsstaten i tiden som kommer, hentet fra sentralbyrå [1].

## 1.1 Brukerkrav

Med mål om å møte de identifiserte utfordringene fra behovsanalysen hos ortopedisk avdeling ved St.Olav Hospital, er det utformet en rekke brukerkrav som er delt inn i ulike kategorier: Tilgjengelighet, Kommunikasjon, Robusthet, Bærekraft og Informasjon.

Kravene skal sikre at løsningen er funksjonell og at den kan tilpasses et bredt spekter av brukere, i tillegg til at den forbedrer og effektiviserer kommunikasjonen mellom sykepleier og pasient. Brukerkravene er presentert i tabell 1. Til venste i tabellen kan en finne kravtype med et tilhørende navn for referansen og til slutt selve kravbeskrivelse.

### Beskrivelse av kravgrupper:

- **Tilgjengelighet:** Sikrer at produktet kan brukes av et bredt spekter pasienter med ulike behov og begrensninger.
- **Kommunikasjon:** Fremmer klar og prioriteringsvennlig kommunikasjon mellom pasienter og sykepleiere.
- **Bærekraft:** Vektlegger miljøvennlige og holdbare løsninger som reduserer miljøpåvirkning.
- **Robusthet:** Fokuserer på produktets fysiske styrke og sikkerhet, inkludert langvarig drift og beskyttelse mot tyveri og skade.

- 
- **Informasjon:** Behandler personlige og sensitive data på en ansvarlig måte og i tråd med lovverket.

**Hovedkrav:** Produktet skal effektivisere kommunikasjonen mellom sykepleiere og pasienter, det skal også tillate tilpasning av systemets kompleksitet etter den enkelte pasients behov.

Tabell 1: Oversikt over brukerkaravene til produktet.

Kravtype	Navn	Krav
A. Tilgjengelighet	A.1 A.2 A.3 A.4 A.5	Man skal til en hver tid kunne sende nødvarsle. Produktet skal være <b>alderstilpasset</b> . Produktet skal være tilpasset de med synsvansker. Produktet skal være tilpasset de med bevegelsesvansker. Produktet skal kunne benyttes av <b>flest</b> mulig, uavhengig av nasjonalitet, alder, kjønn eller funksjonsnedsettelse.
B. Kommunikasjon	B.1 B.2 B.3 B.4 B.5 B.6	Produktet skal synliggjøre eller formidle de forskjellige varslene; nødhjelp, drikke eller hjelp. Produktet skal differensiere pasientens varsler. Produktet skal formidle om en sykepleier responderer på et varsel. Produktet skal ha unik lyd basert på varsel og hastegrad. Produktet skal avlaste sykepleierne fra matbestillinger. Produktet skal vise hvilke rom som har fastende pasienter.
C.Bærekraft	C.1 C.2	Produktet skal kunne resirkuleres. Produktet skal være kompatibelt med det allerede eksisterende systemet.
D. Robusthet	D.1 D.2 D.3	Produktet skal være beskyttet mot stjeling. Produktet skal være beskyttet mot deformering og ødeleggelse. Produktet skal tåle å bli brukt over lengre perioder.
E. Informasjon	E.1	Produktet skal opprettholde alle taushetspliktene.

---

## 2 Konsept

Basert på behovsanalysen og de definerte brukerkavene, se tabell 1, skal systemet tilby en helhetlig løsning som forbedrer kommunikasjon mellom sykepleiere og pasienter gjennom mer dynamiske og tilgjengelige varslingstilnærmingar enn tradisjonelle alarmknapper og snorer.

Slik det er beskrevet i seksjon 1, er dagens varslingssystemer med snor og knapp den eneste formen for kommunikasjon mellom pasienter og sykepleiere. Dette gjør det vanskelig for sykepleierene å skille mellom akutte og ikke akutte behov, og derfor hvilke rom de bør prioritere.

Med et varslingssystem som separerer behovet til pasienten kan helsepersonell arbeide mer effektivt og prioritere akutt hjelp til de som trenger det. For å muliggjøre en slik differensiering, som vil oppfylle brukerkravene B.1 og B.2, er det ønskelig å skissere en mulig applikasjonsprogramvare som kan installeres på sykehøstet. I henhold til brukerkrav C.2, skal ikke det tenkte produktet erstatte varslingssystemet som allerede eksisterer, man skal kunne benytte seg av begge deler.

### 2.1 Mulige løsninger

Ved å tildele pasientene et nytt varslingssystem som tillater varsling av ulike behov, vil problemstillingen i stor grad være løst. Så fremt det nye varslingssystemet er designet og utformet på en effektiv og fordelaktig måte.

Et system bestående av to enheter, en brukerenhet og en styreenhet som kommuniserer via trådløs kommunikasjon, vil kunne fungere som en løsning på problemstillingen. Pasienten sender inn et varsel om spesifikt behov fra brukerenheten og dette kommer inn på styreenheten hvor sykepleieren kan se hva som trengs. Styreenheten kan prioritere hvilket rom som skal betjenes først, ut i fra hvilke type varslere pasientene sender, slik at brukerkravene B.1 og B.2 blir møtt.

Designet av brukerenheten må være lett å håndkes med og følge brukerkravene til kravtypen A. Det er flere potensielle løsninger på designet av brukerenheten, mens styreenheten har gjerne ikke like mange gode løsninger. Styreenheten skal ta inn varslene til pasientene og sortere de på en ryddig måte slik at sykepleier enkelt kan lese hvilke varsel som er sendt og hvor det kommer fra. Det er gunstig dersom systemet kan prioritere varslene ut ifra hvilke varsel som er sendt, og at det har et kart over pasientrommene.

For brukerenheten er det som sagt mange mulige løsninger for å få til et godt design. Valget faller derfor på en brukerenhet bestående av en touch-skjerm og en programvare man kan endre. På denne måten vil det være enkelt for sykepleierne å endre valgmulighetene på brukerenheten, slik at løsningen møter hovedkravet.

## 2.2 Brukergrensesnitt for valgt løsning

Applikasjonen er tenkt å være enkel og intuitiv å anvende, både for pasienter og helsepersonell, for å oppfylle brukerkravene innen kravtypene A og B. Figur 1 illustrerer hvordan dette kan oppnås gjennom et enkelt brukergrensesnitt med figurer og farger som tydelig viser hvilket behov pasienten ønsker å varsle om.



Figur 1: Brukergrensesnitt for pasientene med figurer som indikerer de ulike varslene.

Tanken bak det enkle designet er å gjøre applikasjonen mer intuitiv for de eldre og kognitivt svakere pasientene, som av ulike grunner kan slite med teknologien. Designet skal ha som mål å oppfylle brukerkarav A.2 om tilpasning for alle aldre.

Selve systemet er tenkt å bestå av en touchskjerm med en ramme slik at den kan installeres ved pasientens seng, for enkel tilgang i samsvar med brukerkaravene A.1 for nødvarsle og A.4 om vansker med bevegelse.

Pasienten kan velge hvilken type assistanse de trenger og hastegraden for henvendelsen. Dette møter kravet om at produktet skal synliggjøre og differensierte mellom forskjellige Varsler og hastegrader, brukerkarav B.1, B.2 og B.4. I tillegg vil valgene bli sendt til en sentral datamaskin som overvåkes av sykepleierne, som viser en prioritert liste over henvendelser med detaljer som romnummer og spesifikk assistansebehov.

Idet en pasient sender et varsel, dirigeres også varselet automatisk til helsepersonalets personlige digitale assistenter, en mobil enhet helepersonalet alltid bærer på hver vakt, gjerne kalt PDA. Gjennom disse PDA-ene, vil personalet ha anledning til å markere pasientvarsler for å indikere til andre ansatte at de skal til å hjelpe pasienten, i tråd med brukerkarav B.3. Det vil også være mulig å be om assistanse fra andre ansatte ved behov for dette, eksempelvis under pasientløft. Følgelig vil helsepersonalet kunne jobbe mer effektivt; Sykepleiere vil kunne ta med seg nødvendig utstyr på vei til pasienten, da de allerede kjenner til pasientens behov, og de vil kunne prioritere pasienter som oppholder seg i nærheten av der de selv befinner seg.

Figur 2 er et eksempel på hvordan brukergrensesnittet til varslingsssystemet kan se ut fra helsepersonalet sitt perspektiv. Her vises det en liste over varsler fra pasienter, med informasjon om hva varselet gjelder og hvilket rom pasienten oppholder seg på. I statusfeltet vises initialene til sykepleieren som tar hånd om varselet, og nederst til venstre er en knapp for å tilkalle hjelp fra annet helsepersonell.

Varslinger	Status
Rom 1: AKUTT	MDB
Rom 3: Vann	RF
Rom 5: Snor	TPR
Rom 2: Ønsker gjerne en ekstra pute	
Rom 5: TPR trenger assistanse	
Tilkall hjelp	

Figur 2: Et eksempel på hvordan brukergrensesnittet til varslingssystemet. (Ventre) Varsler fra pasienter med tilhørende rom. (Høyre) Indikerer hvorvidt varselet er svart på, med sykepleiers initialer.

For å avlaste sykepleierne fra håndteringen av matbestillinger, i tråd med brukerkrev B.5, skal systemet sende matforespørser direkte til kjøkkenet. Kjøkkenet skal da kun se de eventuelle bestillingene og tilhørende romnummer for å opprettholde personvern, ifølge brukerkrev E.1. Pasienter som faster skal ikke motta varslinger om mat fra kjøkkenet, for å møte brukerkrev B.6, og sykepleierne skal ha mulighet til å legge inn hvilke rom dette gjelder.

## 2.3 Systemkrav

Basert på systemets funksjonalitet, vil det være nødvendig å oppfylle flere tekniske og operasjonelle krav for å sikre at systemet fungerer effektivt og pålitelig. Utifra den foreslalte løsningen og de satte brukerkrevene, er det utarbeidet flere systemkrav. Systemkravene vil være delt inn etter ulike kategorier: Brukervennlighet, Kommunikasjon, Robusthet, Informasjon, Bærekraft og Matbestilling. Kravene vil inkludere, men ikke være begrenset til; brukergrensesnitt design, dataintegritet, systemets oppetid og sikkerhetsstandarder for å beskytte pasientdata.

Systemkravene er organisert og angitt i Tabell 2, 3, 4, 5, 6 og 7. Tabellene inneholder har et eget nummer for hvert systemkrav for å forenkle refereringen og holde god oversikt. Selve beskrivelsen av systemkravet vil være i den midterste kolonnen og hvilke brukerkrev som systemkravene baserer seg på i kolonnene til høyre.

Det er besluttet å fokusere mindre på brukerkrev B.4, unik lyd basert på varsle, og brukerkrev D.1, beskyttelse mot tyveri. Grunnet at prosjektet er tidsbegrenset og man må prioritere hvilke løsnigninger som er mest fundamentale. Funksjoner som lyd og eventuell sporing vil kreve mindre å legge til etterhvert.

---

## Brukervennlighet

Systemkravene for brukervennlighet er vist i tabell 2. De er laget ut i fra tanken om at produktet skal være lett tilgjengelig for folk med forskjellig alder, kjønn, utfordringer osv.

For at produktet skal være lett å anvende er det skal det være tre ulike moduler og ulike hastegrader for de ulike varslene.

Tabell 2: Systemkrav for brukervennlighet. Tabellen presenterer funksjonaliteter i brukerenheten for å sikre tilgjengelighet og tilpasning til varierte evner og behov.

Nr.	Systemkrav	Tilhørende Brukerkrav
1.1	Systemet skal ha en egen nødknapp, denne skal være tilgjengelig slik at den <b>alltid</b> kan anvendes.	A.1
1.2	Systemet skal ha tre forskjellige moduler for pasienter på brukerenheten: Enkel, middels og vanskelig.	A.2, A.3, A.4, A.5
1.3	Modul 1 på systemet skal ha et begrenset antall valg: Hjelp nå og hjelp senere.	A.2, A.3, A.4, A.5
1.4	Modul 2 på systemet skal ha et mindre begrenset antall valg: Hjelp nå, hjelp senere, mat og drikke.	A.2, A.3, A.4, A.5
1.5	Modul 3 på systemet skal minst begrensning i antall valg: Smerte, toalett, betjening, drikke og mat.	A.2, A.3, A.4, A.5

## Kommunikasjon

Systemkravene for kommunikasjon omhandler hovedsakelig **hvordan** informasjon skal være delt i de forskjellige hastegradene, prioritering, og differensiering. Det omhandler også hvordan systemet skal respondere avhengig av om det brukes av pasient eller sykepleier. Tabell 3 viser til alle kommunikasjons systemkrav.

Tabell 3: Systemkrav for kommunikasjon. Tabellen viser til funksjonaliteten i styreneheten og brukerenheten som forenkler kommunikasjon mellom sykepleier og pasient, mellom sykepleierne.

Nr.	Systemkrav	Tilhørende Brukerkrav
2.1	Varslene som blir sendt fra brukerenhet til styreenhet skal være differensiert.	B.1, B.2
2.2	Varslene på styreenheten skal være rangert etter hastegrad.	B.2
2.3	Hvis et av varslene er tatt skal det fjernes fra både styreenhet og brukerenhet om det velges igjen	B.3
2.4	Systemet skal kunne lese RFID-kort.	B.3
2.5	Systemet skal ha en administrasjonsmeny på brukerenheten som kun sykepleierne har tilgang til. Sykepleierne skal få adgang til denne ved å bruke et RFID-kort.	B.3, C.2

---

## Robusthet

Systemkravene for robusthet er hovedsakelig tiltenkt brukerenheten, grunnet at styreenheten bare er en programvare som kan installeres på det fleste moderne enheter. For at et slikt produkt skal kunne brukes et sykehus må brukerenheten tåle ulik bruk. Det må tåle desinfisering, å utsettes lettere fall eller trykk og det skal kunne brukes over lengre tid uten ekstreme slitasjer. Alle disse systemkravene er organisert i tabell 4.

Tabell 4: Systemkrav for robusthet. Denne tabellen viser til funksjonaliteten i brukerenheten for å sikre trygg brukt og gjenbruk av komponenter.

Nr.	Systemkrav	Tilhørende Brukerkrav
3.1	Produktet skal tåle desinfeksjon.	D.3
3.2	Produktet skal tåle å bli sluppet fra 1 meters høyde uten at funksjonaliteten påvirkes.	A.2,A.4,D.2
3.3	Systemet skal kunne være i tilstand ”på”, med aktiv bruk, i 5 timer.	D.3
3.4	Systemet skal kunne være i tilstand ”av”, hvilemodus, 24 timer.	D.3
3.5	Systemet skal kunne kommunisere mellom brukerenhet og styreenhet via trådløs kommunikasjon med en rekkevidde på opptil 10 meter.	B.1
3.6	Produktet skal opprettholde funksjonaliteten over minst 3 år.	D.3

## Informasjon

For å opprettholde bruksikkerheten og effektivisere kommunikasjon skal produktet kunne koble seg til sikkert nettverk, som for eksempel EDUROAM. Systemkravene for informasjon er organisert i tabell 5.

Tabell 5: Systemkrav for informasjon. Denne tabellen beskriver tekniske krav for å sikre trygg og privat kommunikasjon mellom brukerenhet og styreenhet.

Nr.	Systemkrav	Tilhørende Brukerkrav
4.1	Systemet skal ikke være koblet til felles internett for å beskytte mot informasjons-lekkasjer.	E.1
4.2	Systemet skal ha en styrenhet med visuelt overblikk over hvilke varsler som kommer fra hvilke brukerenheter.	E.1
4.3	Brukerenhet og styreenhet kan kommunisere med hverandre gjennom Bluetooth	E.1

---

## Bærekraft

Et bærekraftig produkt skal redusere klima- og miljøbelastningen med tanke på produksjon og bruk av produktet [2]. Produktet skal ta hensyn til miljøet ved bruk, demontering og gjenbruk, se systemkravene gitt i tabell 6.

Tabell 6: Systemkrav for bærekraft. Denne tabellen presenterer kravene som systemet må oppfylle for å begrenses som bærekraftig, i følge [2].

Nr.	Systemkrav	Tilhørende Brukerkrav
5.1	Minst 90% av alle materialene som brukes under design og implementering skal kunne gjenbrukes.	C.1
5.2	Brukerenheten skal kunne demonteres til enkelte komponenter.	C.1
5.3	Produktet skal være integrerbar med allerede eksisterende system.	C.2

## Matbestilling

En stor del av sykepleierens hverdagen går på å finne ut om pasientene vil spise og hvilken mat de ønsker. Ifølge brukerkrav B.5, se tabell 1.1, skal produktet avlaste sykepleiene fra matbestillinger, og gjøre det enklere for pasienten å bestille mat. Tabell 7 viser systemkravene for matbestilling.

Tabell 7: Systemkrav for matbestilling. Denne tabellen presenterer matbestillingsfunksjonen i systemet.

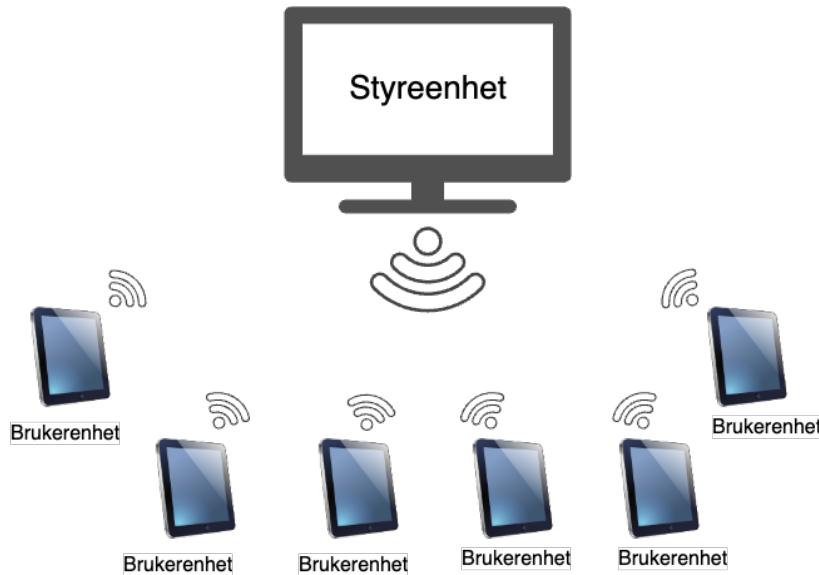
Nr.	Systemkrav	Tilhørende Brukerkrav
6.1	Systemet skal ha mulighet til å separere matbestillinger og sende dem til direkte til kjøkkenet.	B.5, B.6
6.2	Det kan sendes ut matbestillinger fra styreenheten til brukerenheten for frokost og kveldsmat.	B.5, B.6
6.3	Brukerenhetene som faster skal ikke ha mulighet til å bestille eller motta forespørslar om mat.	B.6

---

### 3 Design

Som beskrevet i Konseptet, seksjon 2, er det designet et system, *VarselVenn*, som vist i figur 3, bestående av følgende to delsystem:

- Styreenhet - Samlepunkt for informasjon/varslinger til sykepleierene fra de ulike brukerenhetene som plasseres ut på ulike rom.
- Brukerenheter - Kommunikasjon/varslingssystem for pasient på rom til styreenhet for sykepleierene.

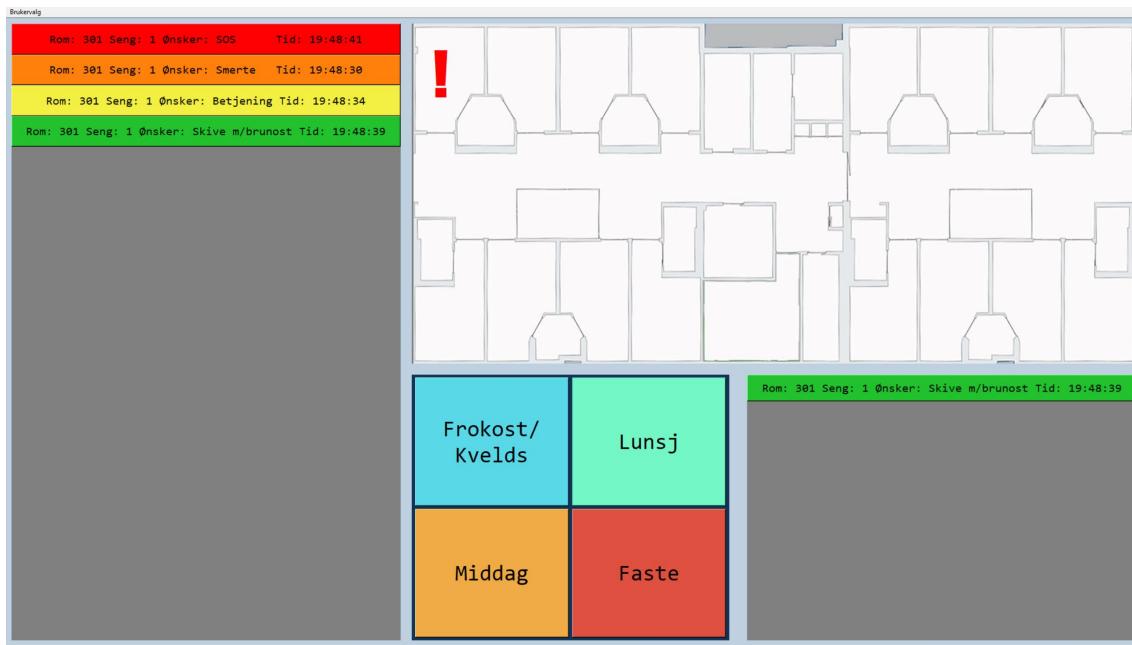


Figur 3: Enkel representasjon av et totalt system med en styreenhet og seks brukerenheter. De to delsystemene skal kommunisere med trådløs kommunikasjon gitt av systemkrav 2.7.

#### 3.1 Styreenhet

Styreenheten skal fungere som en informasjonbank og oversiktsplattform for sykepleierene, som gir mulighet for differensiering av varsler fra pasientene. Utformingen av styreenheten er vist i figur 4 og den består av følgende delsystem:

- Varselliste
- Kart over tunet
- Kjøkkenmeny



Figur 4: Utforming av styreenhet: (*Høyre*) Liste over alle varslinger fra brukerenhetene (*Venstretopp*) Kart over tunet, (*Venstrebunn*), Kjøkkenmeny.

### 3.1.1 Varselliste

Første delsystem av styreenheten er en liste over alle varsler fra brukerenhetene, design av listen er vist i Figur 5. Hvert varsel gir oversikt over romnummer og hvilken seng på rommet varselet kommer fra. Varselet er markert med hva som er årsaken/ønsket fra pasienten og hvilken tid det ble sendt. Varslene er fargekodet med et trafikklyssystem som gjør det enkelt å differensiere hastegraden på varset. Liste med forklaring av farge og tilsvarende hastegrad er presentert under:

- Rødt - Haster mest (Forbeholdt SOS/Stansalarm)
- Oransje - Haster mye
- Gul - Haster litt
- Grønn - Haster ikke

Rom: 301 Seng: 1 Ønsker: SOS	Tid: 10:25:23
Rom: 301 Seng: 1 Ønsker: Smerte	Tid: 12:46:09
Rom: 301 Seng: 1 Ønsker: Betjening Tid: 13:35:09	
Fjern	↑ (Hastegrad)
Legg til tilstedeværelse	↓ (Hastegrad)
Rom: 301 Seng: 1 Ønsker: Skive m/brunost Tid: 12:23:23	

Figur 5: Liste over varsler fra de ulike brukerenhetene og en skjult meny i grøn farge for å gjøre endringer på varslene. Varslene innholder romnummer, info om seng, spesifisert ønske og tid for når varselen ble sendt. I skjult meny for endringer kan en fjerne varselet, endre på hastegraden og vise at det er sett.

Figur 5 viser en skjult meny for sykepleierne i varsellisten. Dersom en sykepleier trykker på et varsel har de mulighet til å gjøre følgende med varselet:

- Fjerne varselet
- Øke hastegraden
- Senke hastegraden
- Legge til tilstedeværelse av sykepleier i rommet.

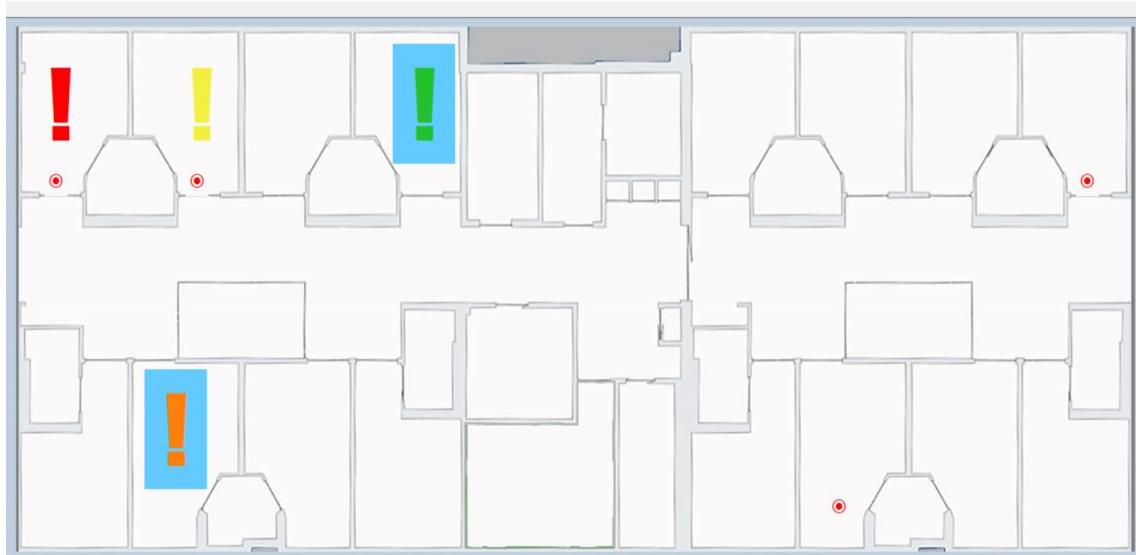
Valgmulighetene i menyen er til for at sykepleiere manuelt skal kunne gjøre endringer av hastegrad ved behov, fjerne varsler som er løst dersom det ikke er gjort på brukerenheten, eller gjøre det synlig at en sykepleier er i rommet og responderer på varselet.

Dette designet av varsellisten tilfredsstiller systemkrav 2.1 ved å differensiere varslene, 2.2 ved å prioritere dem etter hastegrad og 2.3 ved å fjerne varslene som er løst.

---

### 3.1.2 Kart over tunet

Andre delsystem av styreenheten er et kart over tunet som vist i Figur 6. Kartet gjør det enkelt for sykepleiere å få et visuelt overblikk om det er sendt ut ett eller flere varsel, og hvilke rom varselene kommer fra.



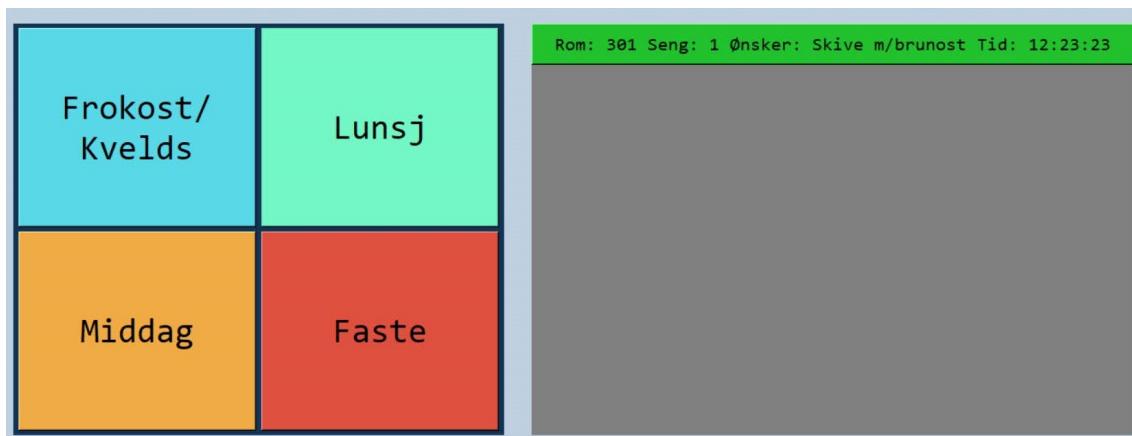
Figur 6: Kart over tunet på styreenheten. På kartet er det flere utropstegn som viser til varselene fra de tilhørende rom. Fargene på utropstegnene viser til hastegraden og blå boks rundt viser til at en sykepleier alledere hjelper til. Røde prikk på rommet viser til at pasienten faster.

Varslene blir illustrert gjennom et utropstegn på rommet som er i tilsvarende farge som hastegraden til varselet. Slik som beskrevet i seksjon 3.1.1. Dersom det er betjening på et rom, altså en sykpleier som responderer på et varsel, som nevnt i seksjon 3.1.1, er tilstedeværelse registrert gjennom å markere varselet med en blå boks. De røde prikkene er til for å indikere hvilke rom som har fastende pasienter slik at sykepleiere lett skal kunne se hvem som ikke skal få beskjed mat og drikke. Dypere forklaring om faste kommer i neste seksjon 3.1.3 under **Faste**.

---

### 3.1.3 Kjøkkenmeny

Kjøkkenmenyen er tredje delsystem av styreenheten, og er vist i figur 7. Kjøkkenmenyen består av to delsystem: Første delsystem er en meny for å sende ut matbestillinger og registrere rom med fastende pasienter. Det andre delsystemet er en liste hvor alle mat- og drikke varsler fra varsellisten er samlet.

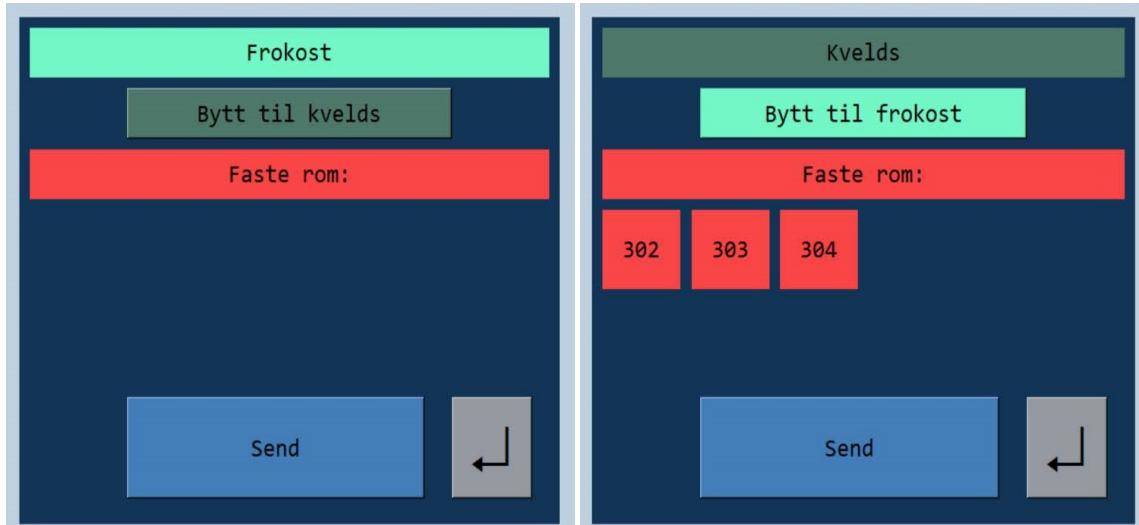


Figur 7: En visuell representasjon av kjøkkenmeny på styreenheten. To delsystem: (Venstre) Meny for utsending av matbestilling til brukerenheten og registrere rom med fastende pasienter. (Høyre) Liste over kun mat- og drikke varsler med tilhørende romnummer og tid.

Kjøkkenmenyen er en del av styreenheten som sykepleierene har på tunet, men den kan også brukes separat på en annen styreenhet på kjøkkenet. På den måten vil de som jobber på kjøkkenet kunne bistå sykepleierene med å håndtere varsler kun relatert til mat og drikke. Dette gir mulighet for effektivisering av arbeidet til sykepleierene ved å avlaste dem fra oppgaver de som jobber på kjøkken kan gjøre, dersom det skulle være andre varsler av større hastegrad sykepleierene må prioritere først.

## Meny for utsending av matbestilling

Til venstre i kjøkkenmenyen i figur 7 er det fire valg. De tre valgene om frokost/kvelds, lunsj og middag, er for utsending av matbestilling fra styreenhet til brukerenhet, og er designet som vist i figur 8.



(a) Meny for utsending av frokostbestilling.

(b) Meny for utsending av kveldsbestilling.



(c) Meny for utsending av lunsjbestilling.

(d) Meny for utsending av middagsbestilling.

Figur 8: Design av meny for utsending av matbestilling fra styreenhet til brukerenhet. (a) viser til utsending av frokostbestilling, (b) viser kveldsbestilling, (c) lunsj-bestilling og (d) middagsbestilling.

Bestillingene kan, som nevnt tidligere, sendes fra enten styreenheten til sykepleierene på tunet, eller fra en egen styreenhet med kun kjøkkenmenyen plassert på kjøkkenet på tunet. De røde feltene markert med faste, er for å indikere hvilke rom som har fastende pasienter. Brukerenhetne på disse rommene vil ikke få varsle om matbestilling, av hensynsmessige årsaker. Mer om dette kommer senere i denne seksjonen under **Faste**.

## Frokost og kveldsmat

Figur 8a og 8b viser det første valget om frokost eller kvelds fra figur 7. Her kan man bytte på hvilket varsel om bestilling som skal sendes, avhengig av om det er morgen eller kveld. Disse to menyene har et større utvalg av mat som pasienten kan velge, og har derfor ikke et felt for å skrive inn dagens meny. Designet av slik varselet for frokost og kvelds vil se ut på brukerenheten, samt tilhørende meny, blir presentert i seksjon 3.2.5.

## Lunsj og middag

Figur 8c og 8d viser de andre to valgene om lunsj og middag fra figur 7. Her fylles det inn i tekstboksen, enten dagens lunsj eller middag. Deretter kan man trykke på send og et varsel om bestilling sendes til brukerenhetene. Designet med tekstboks er valgt ut i fra at sykehusene ofte har faste menyer for disse to måltidene hver dag, som betyr at pasienten ikke må ta egne valg. Designet av slik varselet for lunsj og middag vil se ut på brukerenheten, samt tilhørende meny, blir presentert i seksjon 3.2.5.

## Faste

Som vist i figur 7 er det en egen knapp på kjøkkenmenyen kalt faste. Designet av denne knappen er vist i 9.



Figur 9: Design av meny for å registrere/fjerne rom med fastende pasienter.

Designet gir en oversikt over alle rom med en aktiv brukerenhet. Dersom man velger å legge til et rom på faste, vil dette rommet bli markert med rødt. Når pasienten er ferdig med å faste, går man enkelt tilbake og trykker på rommet igjen. Ved å legge til et rom på faste blir det tatt hensyn til pasienten ved at når et varsel om

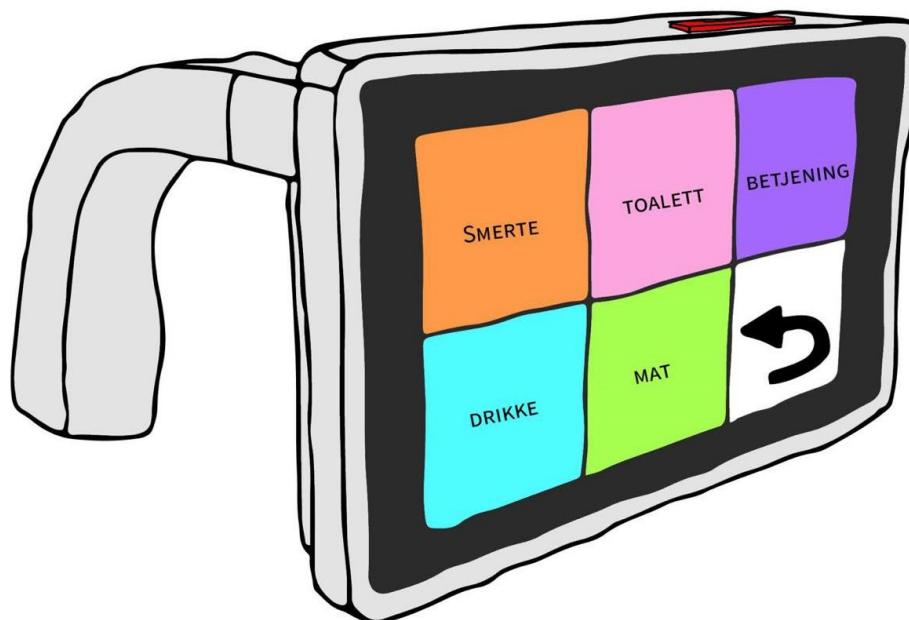
---

matbestilling sendes, vil brukerenheten på de markerte rommene bli ekskludert fra å motta et varsel.

Hvilke rom som er registrert med faste og som ikke vil motta varsel om matbestilling er vist på styreenheten i figur 8. Det er også vist på styreenheten i figur 6 som en rød prikk på kartet over tunet slik at sykepleierene får god oversikt. Design og forklaring på hvordan faste blir tatt hensyn til på brukerenheten, blir presentert i seksjon 3.2.5.

## 3.2 Brukerenhet

Brukerenheten skal være en portabel enhet som pasienter kan bruke til å lett sende ut differensierte varsel om hva de trenger til sykepleierene. Et mulig design av brukerenheten er vist i Figur 10.



Figur 10: Mulig design av brukerenheten for *VarselVenn*. Designet viser til en bærbart enhet med touchskjerm. På touchskjermen er det flere knapper med valgene: Smerte, Toalett, Betjening, Drikke, Mat og en "Tilbake" knapp. Valgene på skjermen representerer valgene i den vanskeligste modulen på brukenheten.

---

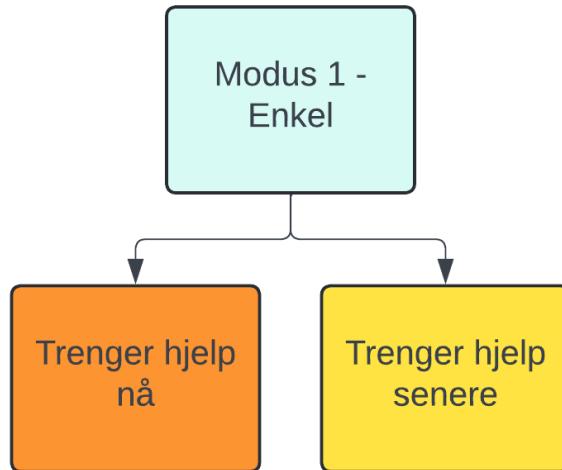
Brukerenheten består av følgende delsystem:

- Modul 1 - Enkel
- Modul 2 - Middels
- Modul 3 - Vanskelig
- Sykepleiermeny

Dette designet av brukerenheten tilfredsstiller systemkrav 1.2 om forskjellige moduler.

### 3.2.1 Modul 1 - Enkel

Det første delsystemet av brukenheten er en enkel modul. Her vil pasienter kun få to enkle valg: Trenger hjelp nå eller trenger hjelp senere. Dette er en modul med minimale valg, men som likevel kan differensiere varselet fra pasienten til en viss grad. Valgene en pasient vil ha i modul 1 er vist i flytskjema i figur 11.



Figur 11: Flytskjema over valg for modul 1 - Enkel.

---

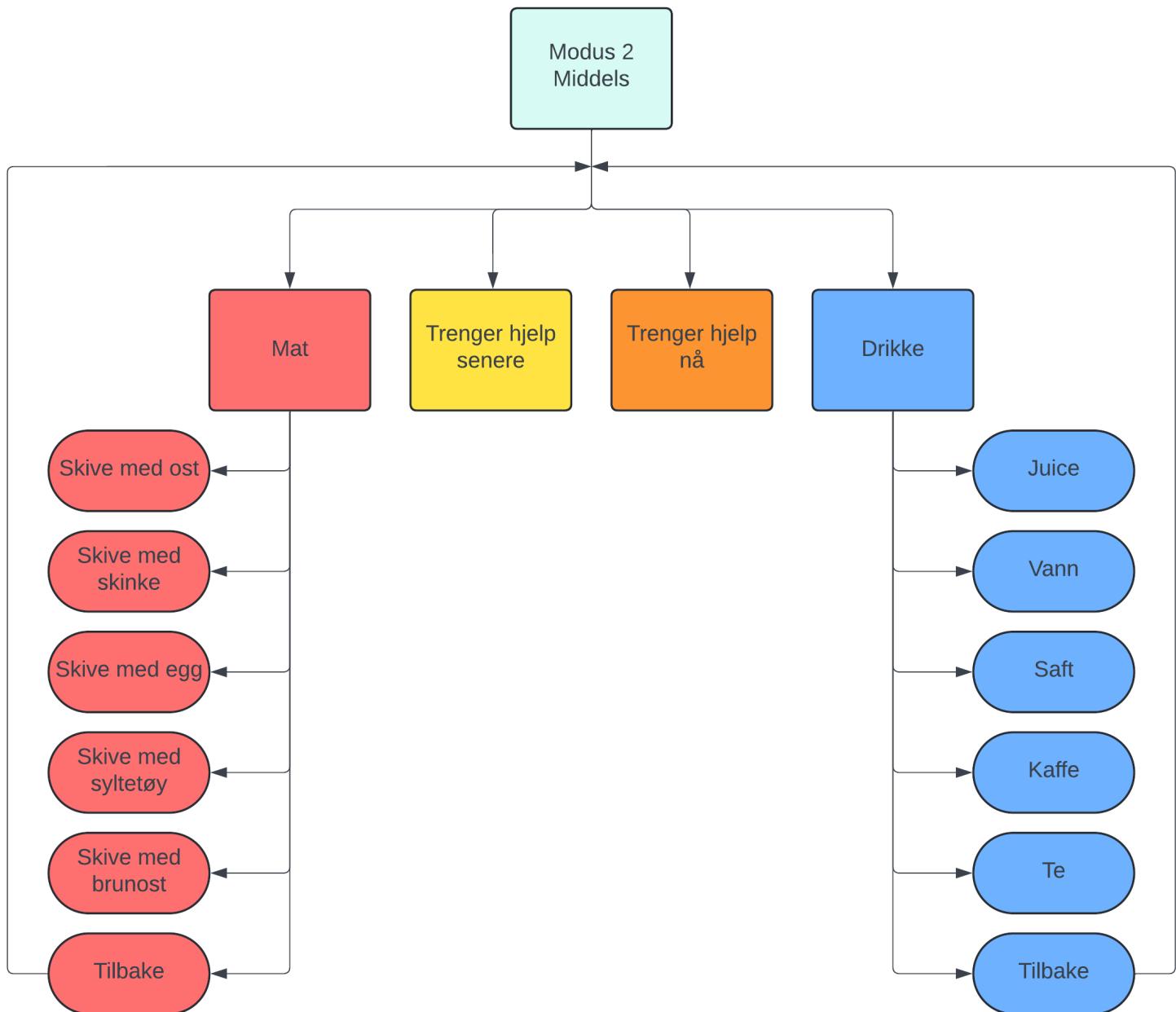
Figur 12 viser hvordan designet av modul 1 vil se ut på brukerenheten. Modul 1 er designet med større knapper og tekst, slik at denne modulen kan brukes av de mer svaksynte og eldre som ikke nødvendigvis trenger flere valg. Designet tilfredsstiller systemkrav 1.3 og 1.6.



Figur 12: Design av modul 1 på brukerenhet. Her har man to valg: om man trenger hjelp nå eller trenger hjelp senere.

### 3.2.2 Modul 2 - Middels

Det andre delsystemet av brukenheten er en middels modul. Her vil pasienter få fire valg: Trenger hjelp nå, trenger hjelp senere, mat eller drikke. Dette er en modul med litt flere valg som differensierer varselet fra pasienten til større grad enn Modul 1. Valgene en pasient vil ha i modul 2 er vist i flytskjema i figur 13.



Figur 13: Flytskjema over valg for modul 2 - Middels.

---

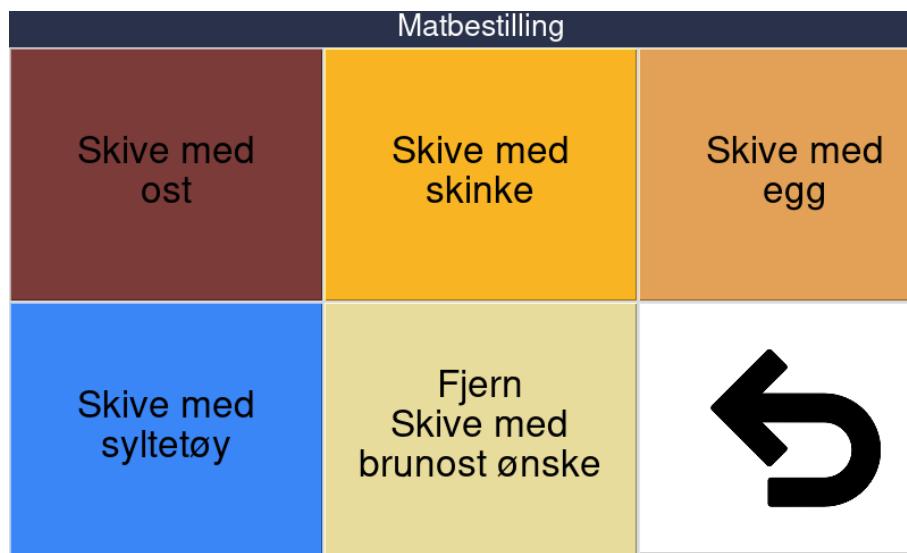
Figur 14 viser hvordan designet av modul 2 vil se ut på brukerenheten. Designet tilfredsstiller systemkrav 1.4.



Figur 14: Design av modul 2 på brukerenhet. Her er det fire valg totalt : Hjelp nå, Hjelp senere, Mat og Drikke.

### Mat

Et eksempel på mulig design av mat knappen på brukerenheten er vist i figur 15. Mat knappen består av et delsystem som gir pasienten mulighet til å bestille mat utenom de faste måltidene. Det vil være en mindre meny som kan tilpasses det tilbudet sykehuset har. Pasienten kan også fjerne ønsket om mat dersom de har trykket feil og skulle ønske annen mat.



Figur 15: Design av matbestilling på brukerenhet.

Mat knappen vil effektivisere arbeidet som oppstår når pasienten vil spise utenom

---

---

faste måltider, ved at sykepleierene slipper å bruke tid på å høre om at pasienten ønsker mat, presentere menyen, lage maten og levere den. Som nevnt i seksjon 3.1.3, vil også alle matvarsler komme på en skjerm på kjøkkenet, som da gir mulighet for å avlaste sykepleierene ved at de ansatte der tar matbestillingen.

## Drikke

Et eksempel på mulig design av drikke knappen på brukerenheten er vist i figur 16. Drikke knappen består av et delsystem som gir pasienten mulighet til å velge hva de ønsker å drikke. Dette kan tilpasses det tilbudet sykehuset har. Pasienten kan også fjerne ønsket om drikke dersom de har trykket feil og skulle ønske annen drikke.

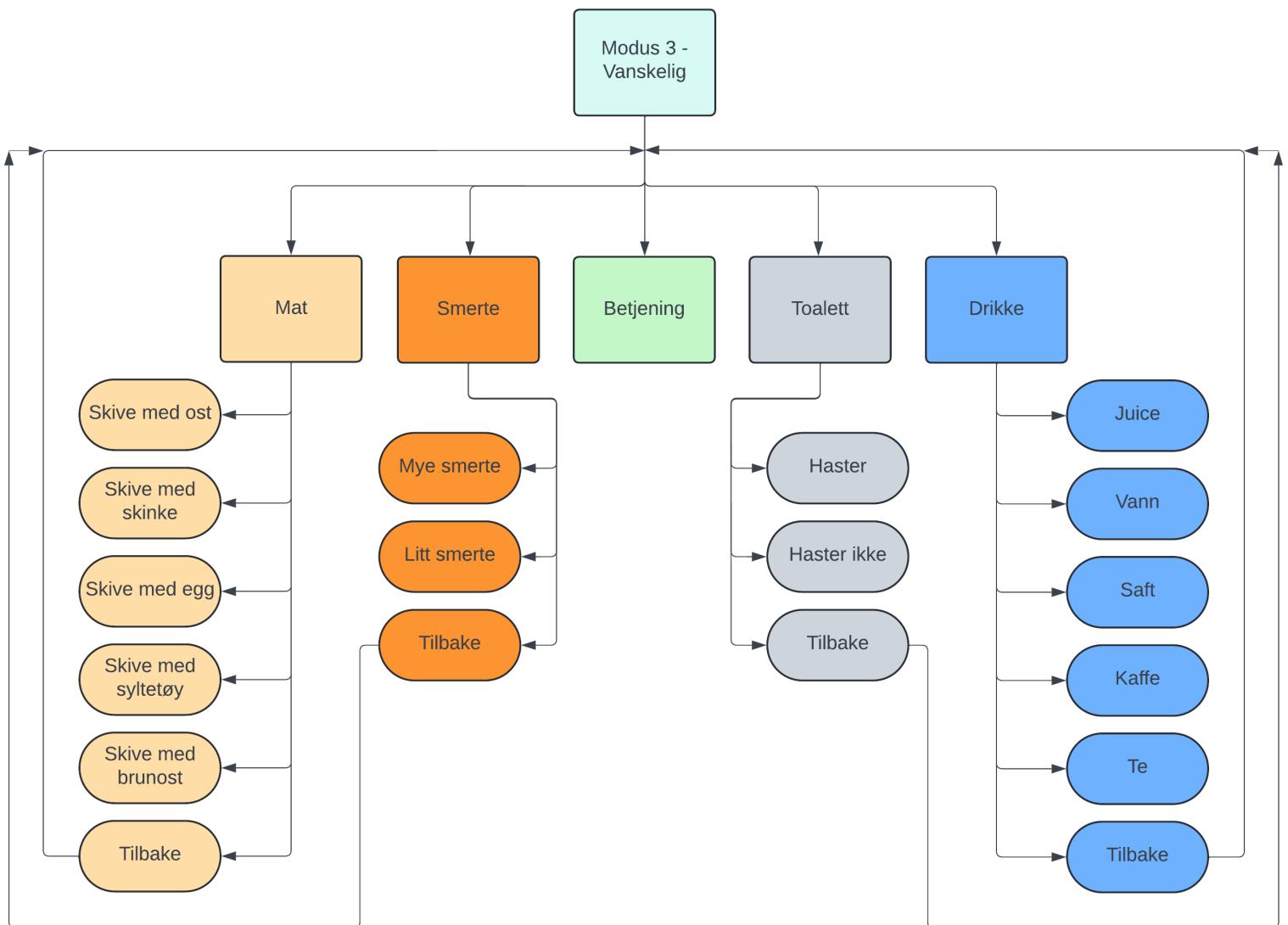


Figur 16: Design av drikkebestilling på brukerenhet.

Drikke knappen vil effektivisere arbeidet ved at sykepleierene slipper å bruke tid på å høre om at pasienten ønsker drikke, presentere valgene, hente drikken og levere den.

### 3.2.3 Modul 3 - Vanskelig

Det tredje delsystemet av brukenheten er en vanskelig modul. Her vil pasienter få fem valg: Smerte, toalett, mat, drikke og betjening. Dett er en modul med flere valg som differensierer varselet fra pasienten i en stor grad. Valgene en pasient vil ha i modul 3 er vist i flytskjema i figur 17.



Figur 17: Flytskjema over valg for modul 3 - Vanskelig.

---

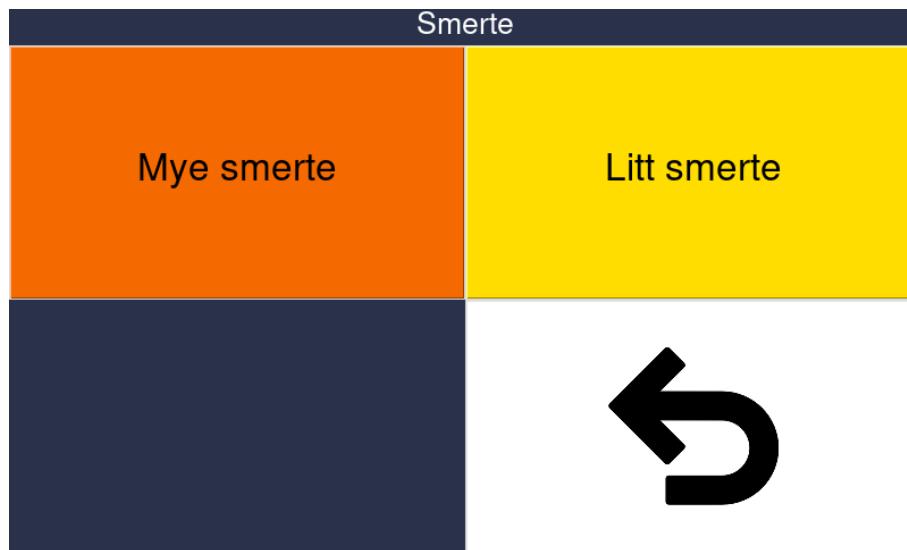
Figur 18 viser hvordan designet av modul 3 vil se ut på brukerenheten. Designet tilfredsstiller systemkrav 1.5.



Figur 18: Design av Modul 3 på brukerenhet. Her er de fem første valg: Smerte, Toalett, Betjening, Drikke og Mat.

### Smerte

Smerte knappen består av et delsystem som gir pasienten mulighet til å velge hvor mye smerte de har: mye eller litt. Valget vil ha betydning for hastegraden til varselet. De to valgene er vist i figur 19.

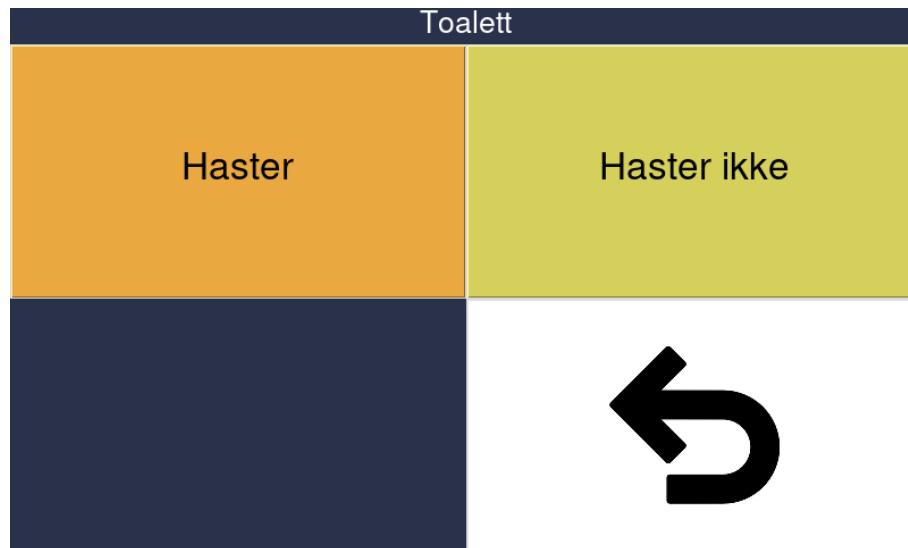


Figur 19: Design av smertemeny på brukerenhet. Ved å velge Smerte på Modul 3 vil denne menyen komme opp. En pasient kan velge om det er mye eller lite smerte, som vil resultere i forskjellige hastegrad på styreenheten.

---

## Toalett

Toalett knappen består av et delsystem som gir pasienten mulighet til å velge dersom det haster eller ikke at de må på toalettet. Valget vil ha betydning for hastegraden til varselet. De to valgene er vist i figur 20.



Figur 20: Design av toalett meny på brukenhet. Ved å velge Toalett på Modul 3 vil denne menyen komme opp. Pasienten kan velge om det haster eller ikke, som vil resultere i tilhørende hastegrad, gult og oransje på styreenheten.

## Betjening

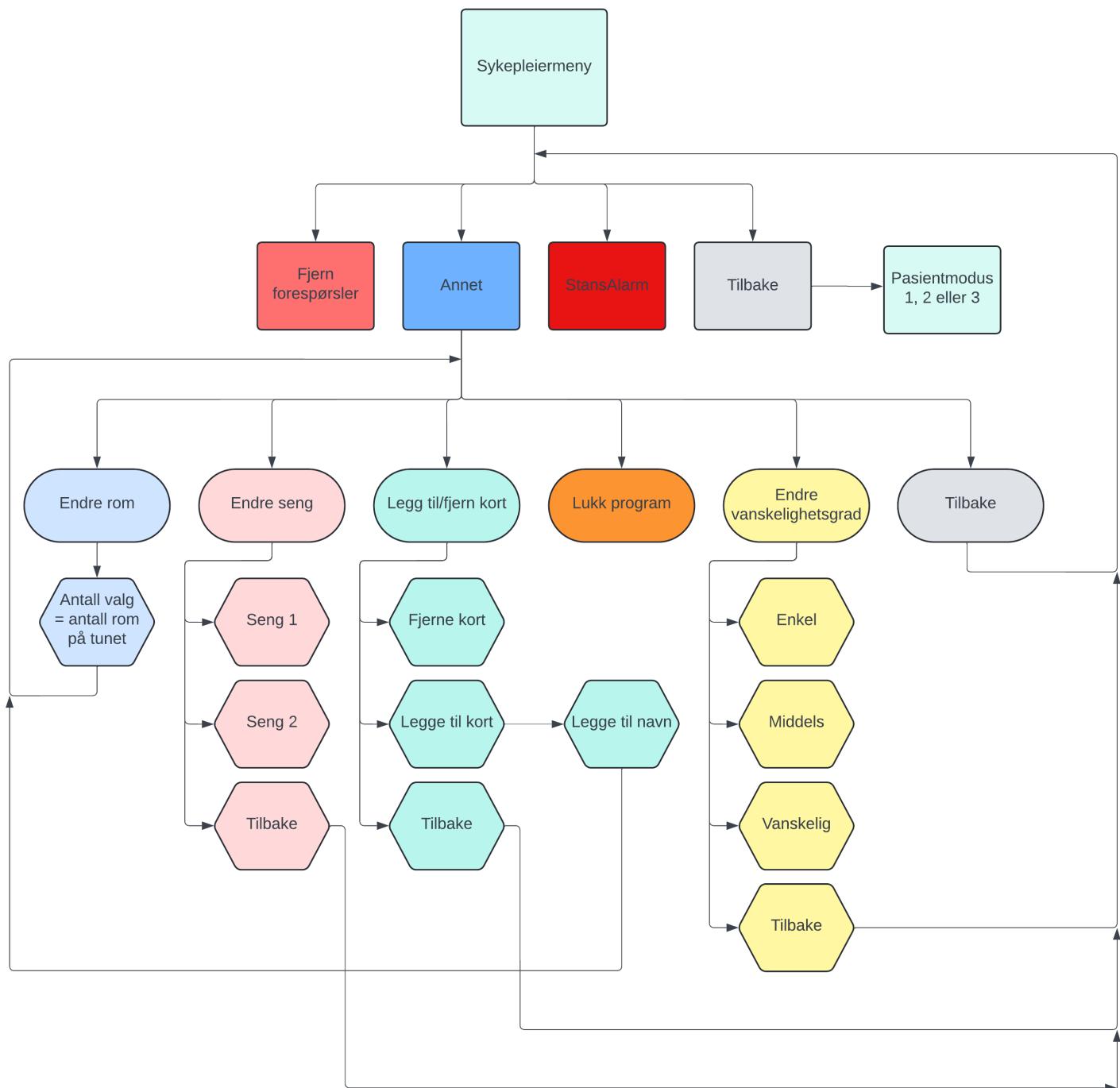
Knappen for betjening er veldig enkel og består ikke av et underliggende delsystem. Den fungerer slik at dersom en pasient trykker på denne, vil det bli sendt et varsle om at pasienten ønsker betjening. Betjening er til for generelle spørsmål eller annet pasienten skulle trenge hjelp med som ikke er en av de andre knappene.

## Mat og drikke

Design av mat- og drikke knappene er presentert og forklart i seksjon 3.2.2.

### 3.2.4 Sykepleiermenu

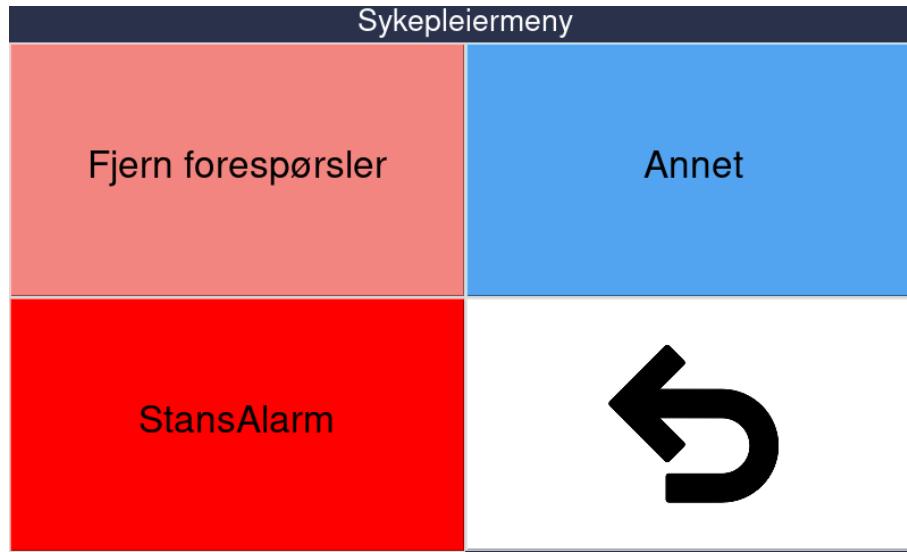
Det siste delsystemet av brukenheten er en sykepleiermenu. Denne har kun sykepleierene tilgang til, og vil dukke opp dersom de skanner ID-kortet sitt på siden av brukerenheten. Den består av tre valg: Fjern forespørsler (varsler som er gjort), stansalarm og annet. Annnet vil bestå av en ny delmeny med flere valg. Valgene sykepleierene vil ha i denne menyen er vist i flytskjema i figur 21.



Figur 21: Flytskjema over valg for sykepleiermenu.

---

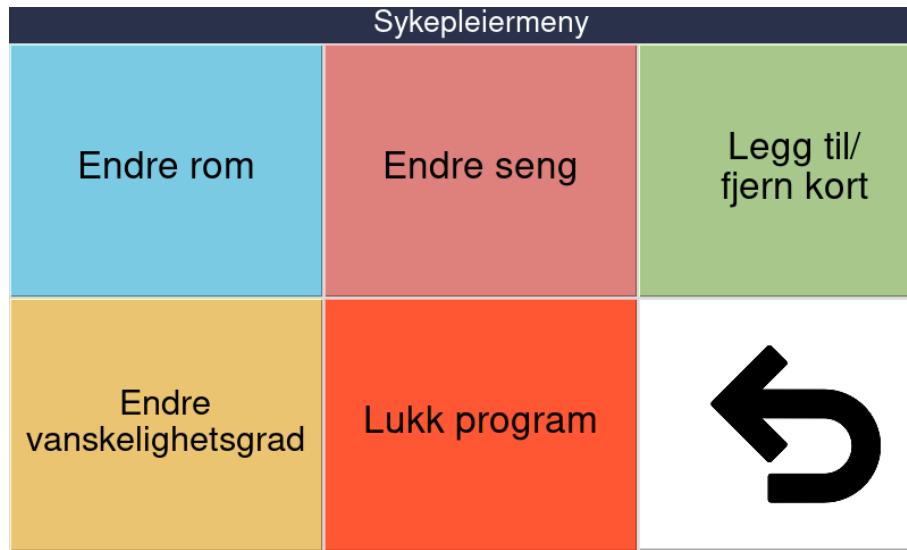
Figur 22 viser hvordan designet av sykepleiermenyen vil se ut på brukerenheten. Designet tilfredsstiller systemkrav 2.3, 2.4 og 2.5.



Figur 22: Design av sykepleiermodul på brukerenhet. Sykepleieren kan velge mellom å fjerne alle forespørslar, sette på StansAllarm eller Annet.

### Annet

Figur 23 viser design av delsystemet 'annet' i sykepleiermenyen. 'Annet' knappen gir sykepleierne mulighet til å sette opp brukerenheten: De kan endre rom, hvilken seng på det aktuelle rommet og endre vanskelighetsgrad tilpasset pasienten som skal ta i bruk brukerenheten. Sykepleierne kan legge til nye ID-kort dersom det er nye sykepleiere på tunet, eller fjerne ID-kort til sykepleiere som ikke lenger er på tunet.

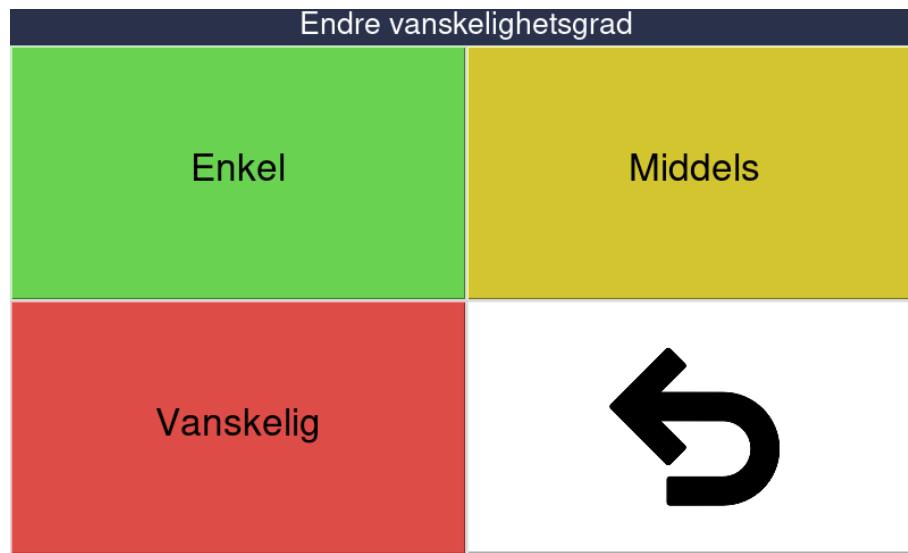


Figur 23: Design av Annet meny i sykepleiermenyen. Ved å velge Annet på sykepleiermodulen vil denne menyen komme opp. Her har sykepleieren flere alternativer å velge mellom.

---

## Endre vanskelighetsgrad

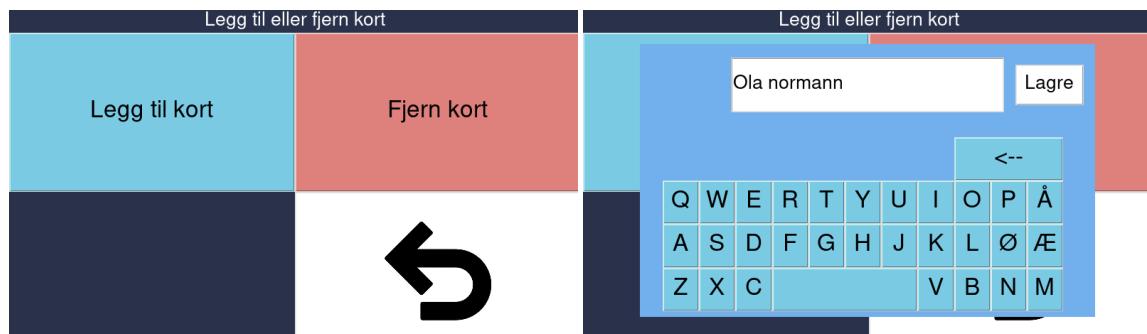
Figur 24 viser design av delsystemet for valg av vanskelighetsgrad i sykepleiermenyen. Dette delsystemet gir sykepleieren mulighet til å sette opp brukerenheten i en modul tilpasset den nye pasienten som skal ta i bruk den. Her er valgene: Enkel, middels og vanskelig. Henholdsvis modul 1, 2 og 3, som er presentert og forklart i seksjon 3.2.1, 3.2.2 og 3.2.3.



Figur 24: Design av meny to valg av vanskelighetsgraden (Modul nr.) brukerenhet skal være satt opp etter.

## Legge til eller fjern kort

Figur 25 viser delsystemet for å legge til eller fjerne ID-kort. For å legge til nytt eller fjerne et ID-kort, skannes kortet på siden av brukerenheten. Dersom et nytt kort legges til, dukker det opp en meny for å registrere navnet til sykepleieren som kortet tilhører.



(a) Legge til og fjerne ID-kort.

(b) Registrere navn tilhørende nytt ID-kort.

Figur 25: Design av delsystem for å legge til eller fjerne ID-kort på brukerenheten i sykepleiermodul.

---

## Endre rom og seng

Figur 26 viser delsystemet for å sette opp brukerenheten til et utvalgt rom og hvilken seng på rommet den skal være tilknyttet til. Varsler som blir sendt fra brukerenheten vil alltid være markert med romnummer og enten seng 1 eller 2. De to valgene, endre rom og endre seng, er derfor til slik at når en pasient er ferdig med brukerenheten, kan sykepleierne enkelt gå inn på menyen og sette opp brukerenheten klar for ny pasient. I tillegg til å velge rom og seng velges da også en tilpasset vanskelighetsgrad som forklart tidligere i seksjon 3.2.4 om endring av vanskelighetsgrad.

Rom 301	Rom 302	Rom 303	Rom 304	Velg Seng:	
Rom 305	Rom 306	Rom 307	Rom 308	Seng 1	Seng 2
Rom 309	Rom 311	Rom 313	Rom 315		

- (a) Velge av rom brukerenheten skal være plassert på.  
(b) Velge av seng på rommet brukerenheten skal være plassert ved.

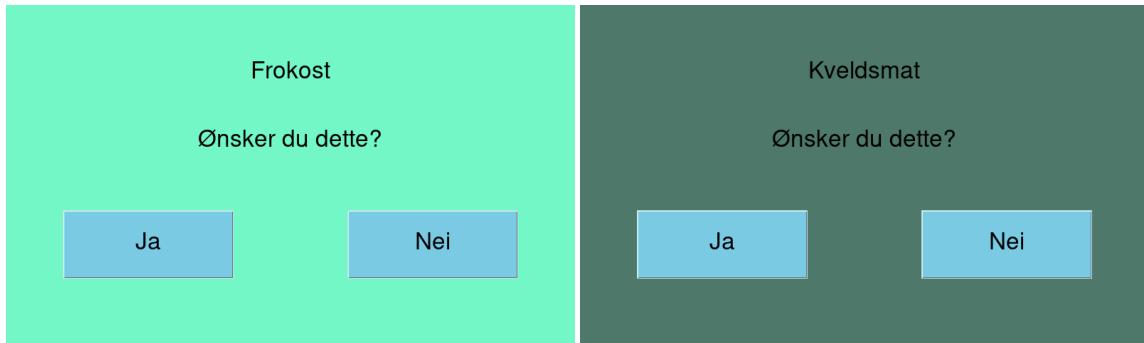
Figur 26: Design av delsystem for å sette opp brukerenheten på utvalgt rom og seng. Denne menyen kommer opp om sykepleieren velger å Endre rom eller seng i sykepleiermodul.

### 3.2.5 Måltider

I seksjon 3.1.3 blir det presentert designet av delsystem for kjøkkenmeny på styrenheten. Her kan sykepleierene og kjøkkenpersonale sende ut matbestilling til brukerenheten, noe som effektiviserer arbeidet ved at de slipper å gå inn til pasienter og høre om de skal ha mat, presentere menyen og så gå å lage maten.

#### Frokost og kvelds

Varselet om matbestilling for frokost og kveldsmat som sendes fra styreenhet til brukerenheten, er designet til å komme opp på brukerenheten som vist i figur 27.



(a) Forespørsel om frokost.

(b) Forespørsel om kveldsmat.

Figur 27: Design av delsystem for å motta forespørsel om frokost og kveldsmat fra styreenheten. Her kan pasienten velge om han ønsker maten eller ikke.

Dersom pasienten ikke ønsker frokost eller kveldsmat trykker de enkelt **nei** og varselet forsvinner. Pasienter vil alltid ha mulighet til å likevel bestille mat selv senere i modul 2 og 3, eller ved å kontakte sykepleier i modul 1.

Dersom pasienten ønsker å spise frokost eller kveldsmat ved de faste tidene, trykker de på **ja** i varselet og vil deretter bli sendt til menyen for matbestilling som vist i figur 28.

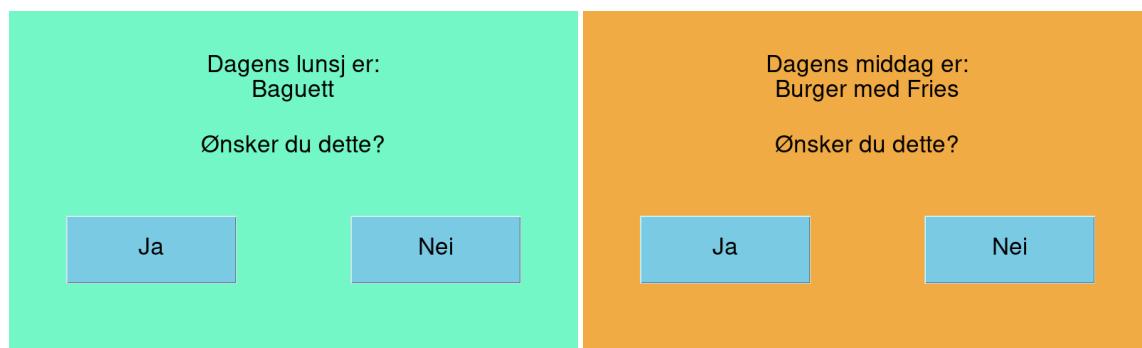
	Matbestilling	Trykk for å fjerne
Brødkive	Fjern Majones	Brødkive,Ost,Skinke
	Smør	Knekkebrød,Brunost,Syltetøy
Knekkebrød	Ost	
	Brunost	
Fjern Rundstykke	Skinke	
	Fjern Egg	Legg til
	Syltetøy	Send bestilling

Figur 28: Meny for matbestilling ved frokost og kveldsmat. Denne menyen kommer opp om pasienten trykker ”Ja” på forespørsel om mat i figur 27.

Menyen for matbestilling i figur 28 er eksklusiv for frokost og kveldsmat og kommer kun opp dersom pasienter trykker at de ønsker det. Menyen er designet på måten at pasientene først velger et type brød, de kan deretter ta på/fjerne de ulike påleggene de skulle ønske og trykker **legg til**. Pasientene kan legge til flere like/ulike kombinasjoner ut ifra hvor sultne de er. Når de har lagt til det de ønsker trykker de enkelt på **send bestilling**. Bestillingen sendes da til styreenheten både på varsellisten som vist i figur 5, og på listen over matbestillinger på kjøkkenmenyen vist i figur 7.

## Lunsj og middag

Varselet om matbestilling for lunsj og middag som sendes fra styreenhet til brukerenhet, er designet til å komme opp på brukerenheten som vist i figur 29.



(a) Forespørsel om lunsj.

(b) Forespørsel om middag.

Figur 29: Design av delsystem for å motta forespørsel om lunsj og middag. Her kan pasienten velge å si ”Ja” eller ”Nei” til maten.

---

Varselet vil vise hva dagens lunsje eller middag er og dersom pasienten ikke ønsker å spise trykker de enkelt **nei** og varselet forsvinner. Pasienter vi alltid ha mulighet til å likevel bestille mat selv senere i modul 2 og 3, eller ved å kontakte sykepleier i modul 1.

Dersom pasienten ønsker å spise lunsj eller middag ved de faste tidene, trykker de på **ja** i varselet og bestillingen blir sendt til styreenheten på samme måte som for frokost og kveldsmat.

### Endret modul ved faste

Dersom en pasient er markert med at de skal faste, vil designet på modulene på brukerenheten ta hensyn til dette slik at pasienten ikke skal bli påminnet om mat og drikke når de ikke har mulighet til å velge dette selv. Siden modul 1 ikke inneholder en knapp for mat eller drikke er det kun modul 2 og modul 3 som endres. Det nye designet av modulene under fasteperioden er vist i figur 30.



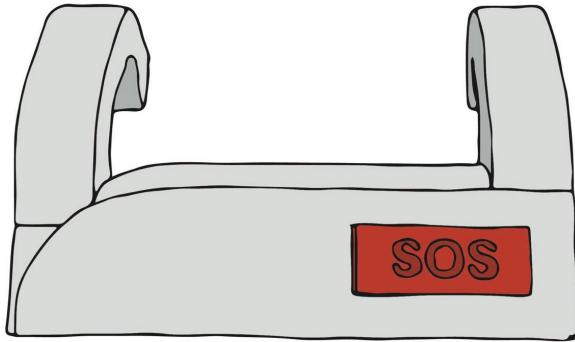
(a) Modul 2 sine valg dersom en pasient skal faste.  
(b) Modul 3 sine valg dersom en pasient skal faste.

Figur 30: Design av delsystem for å tilpasse modul 2 og 3 for å ta hensyn til fastende pasienter. Hvis sykepleien markerer rommet eller seng med ”Faster” vil ikke pasienten ha mulighet til å spør om Mat eller Drikke.

---

### 3.3 SOS

Brukerenheten har et delsystem som er uavhengig fra de andre delsystemene: SOS knappen. SOS knappen skal være en fysisk knapp plassert på utsiden av brukerenheten og er vist i figur 31.



Figur 31: Design av SOS knapp på brukerenhet. Knappen skal være plassert på toppen av brukerenheten, med SOS tekst på.

Ved å plassere SOS knappen på toppen av brukerenheten, sikres det at pasienter alltid skal ha mulighet til å enkelt trykke på denne, uten å måtte trykke på en av valgene på menyen på skjermen. Dette tilfredsstiller systemkrav 1.1.

Dersom SOS knappen blir trykket på vil det komme opp et varsel på brukenheten som vist i figur 32, i tillegg til på varsellisten på styrenheten som vist i figur 5. Dette er slik at pasienten får beskjed om at alarm er sendt, men er også designet som en feilsikring, slik at dersom pasienten har utløst knappen ved en feil, vil de kunne avbryte varselet.



Figur 32: Design av varsel om SOS på brukerenhet. Ved å trykke på SOS knappen i figur 31, vil pasienten få dette på skjærmen. Pasienten kan velge å trykke på AVBRYT.

---

## 4 Implementering

### 4.1 Komponentbeskrivelse

#### Raspberry Pi

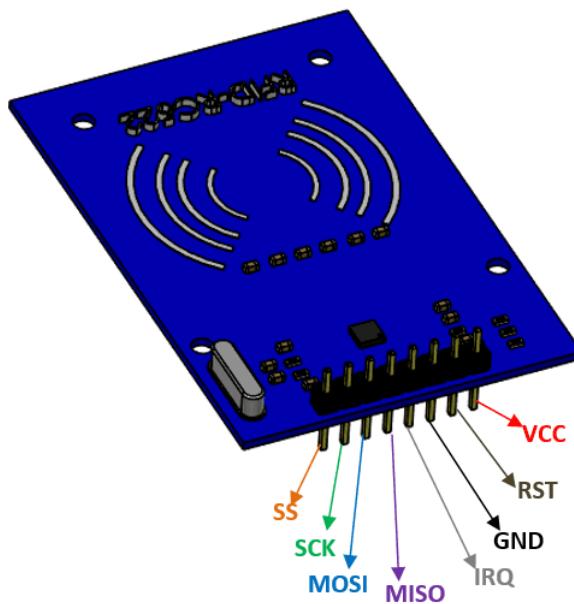
Raspberry Pi (RPi), er en liten datamaskin på et enkelt kretskort som blant annet kan brukes til å motta, behandle og sende data. Systemet bruker en Raspberry Pi 3 model B+. I det realiserte designet brukes RPi-en til å motta og behandle sensordata og lagre det i en database, for å så visualisere den relevante dataen til brukeren på en oversiktlig måte.

#### Raspberry Pi Screen

Raspberry Pi Screen, en 7touch-skjerm med display, blir brukt til realiseringen av produktet, prototypen. Skjermen kobles til Raspberry Pi via et adapterkort som håndterer strøm og signalomforming. Bare to tilkoblinger er nødvendige: strøm fra GPIO-porten og en båndkabel som kobles til DSI-porten.

#### RFID-skanner

En RFID-skanner er et digitalt gjenkjenningssystem som kommuniserer trådløst mellom datasystemer og mikrobrikker plassert i gjenstander. Dermed kan den gjenkjenne de vanligste ID-kort og lese av data fra dem. Dermed i dette prosjektet blir det nyttet typen RC522 [3], ettersom den er lett tilgjengelig og fungerer bra med Raspberry Pi. En modell av RC522 er vist i figur 33.



Figur 33: RFID-skanner av typen RC522 med pinout.

---

## Batteri

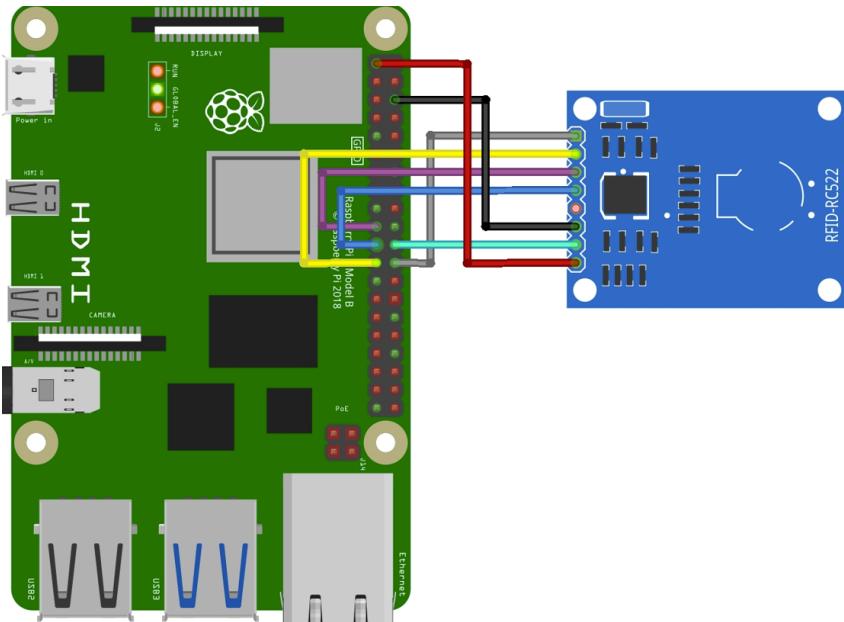
Får at brukerenheten skal kunne være portabel, blir det nytet ett batteri som strømforsyning for alle de øvrige komponentene. Batteriet som nyttes er i form av en 'powerbank', ettersom det er en allerede eksisterende komponent som lett kan implementeres til prototyping. Samt at moderne powerbanker har høy kapasitet og mulighet til å levere de 15W med strøm som Raspberry Pi-en krever. Derfor ble powerbanken fra GP med modellnr. MP10MA [4] nytet, ettersom den har en kapasitet på 10000mAh og muligheten til å levere 15W.

## 4.2 Kretsskjema

I denne seksjonen presenteres alle fysiske oppkoblinger av komponenter med blokk-skjema, kretsskjema og pinlout.

For å implementere muligheten til å lese av RFID-kort, og dermed oppfylle systemkrav 2.4, nyttes det en RFID-skanner av typen RC522 som skal kunne koples til Raspberry Pi-en som vist i Figur 34. Fra figuren kan man se at RFID-skanneren er koplet sammen til Raspberry Pi-en igjenom SPI-pinsene, 3.3V, og Reset (RST). For mer informasjon om SPI, se seksjonen om SPI. Denne koblingen er også vist i tabell 8. For å kunne implementere funksjonaliteten til RC522-en i et python program på Raspberry Pi-en utnyttes biblioteket MFRC522, for mer informasjon om dette biblioteket se [5].

### RFID og RPi



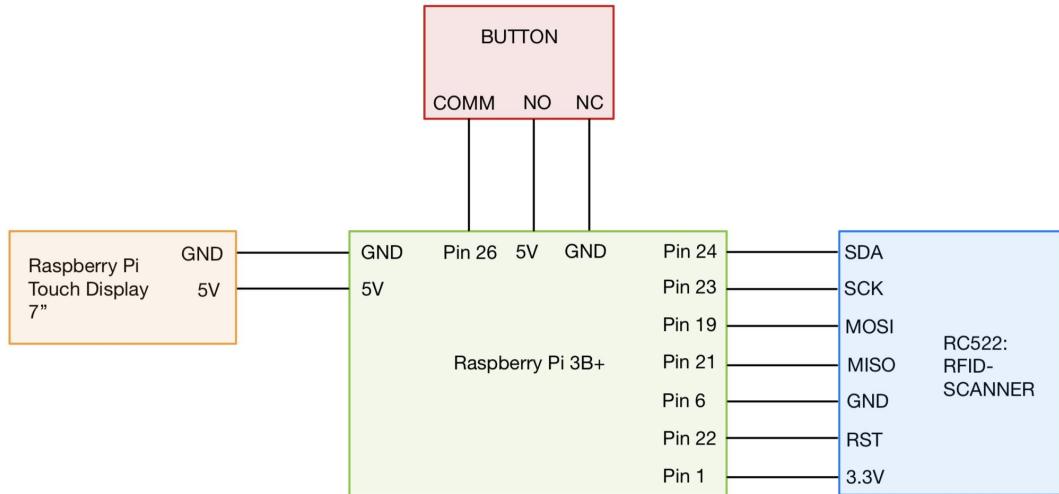
Figur 34: Fysisk representasjon av kobling mellom RFID-skanner, av typen RC522, og Raspberry Pi modell 3B+.

Tabell 8 viser koblingen mellom pinsene på RC522 RFID-skanneren og de tilsvarende pinsene på en Raspberry Pi. Tabellen er viktig for forståelse av den tekniske hardwaren som ble brukt, og inkluderer nødvendig informasjon om spenningforsyning, dataoverføring og tilbakestillings-signaler.

Tabell 8: Oppkobling mellom RC522 RFID-skanner og Raspberry Pi (RPi).

RC522 PIN	RPi PIN
SDA	Pin 24
SCK	Pin 23
MOSI	Pin 19
MISO	Pin 21
GND	Pin 6
RST	Pin 22
3.3V	Pin 1

Figur 35 viser et detaljert kretsskjema for integrering av brukerenheten, med RPi, skjermen og RFID-skanneren. I tillegg kan en SOS knapp også kobles til systemet via GPIO-pin 26 og jord, som vist i figuren. Kretsskjemaet illustrerer hvordan de forskjellige komponentene er koblet sammen med de numrerte pinsene.



Figur 35: Kretsskjema for kobling av det totale systemet, med Raspberry Pi (RPi) modell 3B+, RPi touch skjerm, SOS knapp og RFID-skanner.

---

## 4.3 Kommunikasjon

### SPI

SPI står for Serial Peripheral Interface Bus. SPI en databuss som sørger for synkron seriell kommunikasjon mellom komponenter i ett system. SPI bruker en master-slave arkitektur, der en enhet styrer dataoverføring og klokkesignal til en eller flere slave-enheter.

Databussen bruker fire ledninger, **CS** eller chip select som brukes til å skru på og av kommunikasjonen mellom slaven og master. **SCLK** som er klokkesignalet. De to gjen-værende ledningene **MOSI** og **MISO** til å overføre data henholdsvis fra og til master. For mer informasjon om dette se [6].

### Bluetooth

Kommunikasjonen mellom styreenheten og brukerenheten gjøres igjennom Bluetooth standarden ettersom den har en rekkevidde på opptil 14-16m [7]. Dermed tilfredsstiller systemkrav 2.6 om rekkevidde og 4.3 om å bruke Bluetooth som kommunikasjonsprotokoll.

For å kople sammen de ulike enhetene igjennom Bluetooth brukes Python standardbiblioteket **Socket** som støtter de fleste nettverkstandarder på Linux, MacOS og Windows. For mer informasjon om **Socket** se [8].

## 4.4 Kode

I denne seksjonen vil de viktigeste delene av koden presenteres.

### 4.4.1 RFID-kode

Koden til RFID-skanneren består i hovedsak **read()** funksjonen, som skal lese av både kort-id og data fra RFID-kortet og returnere dette. Denne funksjonen skal kjøre kontinuerlig i en løkke, for å alltid søke etter RFID-kort i nærheten.

Den implementerte koden for **read()** er vist i listing 1. **SimpleMFRC522** er en klasse som gjør det enklere å lese av RFID-kortet, og kommer fra **mfrc522** biblioteket. **RPi.GPIO** blir brukt til å tilbakestille tilstanden til alle GPIO-pins som blir brukt til å lese.

---

```

1 import RPi.GPIO as GPIO
2 from mfrc522 import SimpleMFRC522
3
4 def read():
5     # Prøver først å lese av id og data fra ett RFID-kort
6     try:
7         reader = SimpleMFRC522() # Lager ett objekt av klassen
    ↳ SimpleMFRC522, som gjør klar RFID-scanneren til
    ↳ kortavlesning
8         id, data = reader.read() # Leser av kort ID og data
9         return id, data # Returnerer data og ID
10
11     # Dersom ett kort ikke eksisterer eller klarer å bli lest,
    ↳ eller andre feil skjer skal det returneres tom data og ID
12 except Exception as e:
13     return None, None
14 finally:
15     # Rydder opp tilstanden til GPIO pinnene som ble brukt
16     GPIO.cleanup()

```

Listing 1: Kode til kortavlesningsfunksjon.

#### 4.4.2 Bluetooth-kode

Som forklart i seksjonen **Bluetooth** brukes biblioteket **Socket** i dette prosjektet til å kommunisere igjennom Bluetooth. Listing 2 - 4 viser koden som er ansvarlig for kommunikasjonen mellom brukerenhet og styreenhet. Derav nyttet funksjonene:

- `receive_data()` til å opprette en kobling mellom brukerenhet og styreenhet, og deretter lytte etter data overføringer. For deretter å bruke den mottatte dataen til å oppdatere informasjon på brukerenhet og styreenhet. Koden til denne er vist i listing 2.
- `sendData(Data)` til å sende data til den tilkoblede styrenheten/brukerenheten. Koden er vist i listing 3.
- `sendRequestBT(request,function)` brukes til å sende en request, eller varsel på rett format til den tilkoblede styrenheten/brukerenheten. Koden er vist i listing 4.

---

```

1 # Funksjonen som skal motta data fra tilkoplet enhet
2 def receive_data():
3     global client
4     server = socket.socket(socket.AF_BLUETOOTH, socket.SOCK_STREAM,
5                             → socket.BTPROTO_RFCOMM)
6     server.bind(("38:d5:7a:7d:5d:2e", 4))
7     # Lytter etter enheter som ønsker å kople til
8     server.listen(99)
9     # Godtar enheter som ønsker å koble til
10    client, address = server.accept()
11    try:
12        # Prøver å hente data som blir sendt, men dersom der ikke
13        → er noe, slutter den løkka
14        while True:
15            data = client.recv(1024)
16            if not data:
17                break
18            data = data.decode('utf-8')
19            # Bruker dataen til legge til en "request" i "request
20            → listen"
21            useData(data)
22        except OSError as e:
23            pass
24        client.close()
25        server.close()

```

Listing 2: Koden som er ansvarlig for kommunikasjonen mellom brukerenhet og styreenhet.

```

1 # Sender data til tilkoplet enhet
2 def sendData(data):
3     # Prøver å sende data via client objektet som er definert på
4     → linje #3.
5     try:
6         client.send(data.encode('utf-8'))
7     except OSError as e:
8         pass
9     return

```

Listing 3: Kode for sending av data via Bluetooth.

---

```

1 # Sender en "request" på rett format over Bluetooth
2 def sendRequestBT(request,function):
3     # Formaterer meldingen som skal sendes, basert på "request"
4         ↳ dictionarien
5     message =
6         ↳ f"{function},{request.get('Rom')},{request.get('Seng')},{request.get('Hva')}"
7
8     # Dersom enheten er tilkoplet en annen enhet, skal messange
9         ↳ sendes
10    if clientEnabled:
11        sendData(message)

```

Listing 4: Kode for sending av varsel via Bluetooth.

I tillegg blir pythonbiblioteket `threading` [9] nyttet for å tillate flere funksjoner å kjøre samtidig, da spesielt viktig er muligheten for å alltid søke etter mottatt data samtidig som man oppdaterer UI. Dette blir implementert i funksjonen `start_thread()` i listing 5.

```

1 # Oppretter en ny tråd for receive_data() hvis clientEnabled = true
2 def start_thread():
3     if clientEnabled:
4         receive_thread = threading.Thread(target=receive_data)
5         receive_thread.start()

```

Listing 5: Kode opprettning av tråd for å motta data fra Bluetooth.

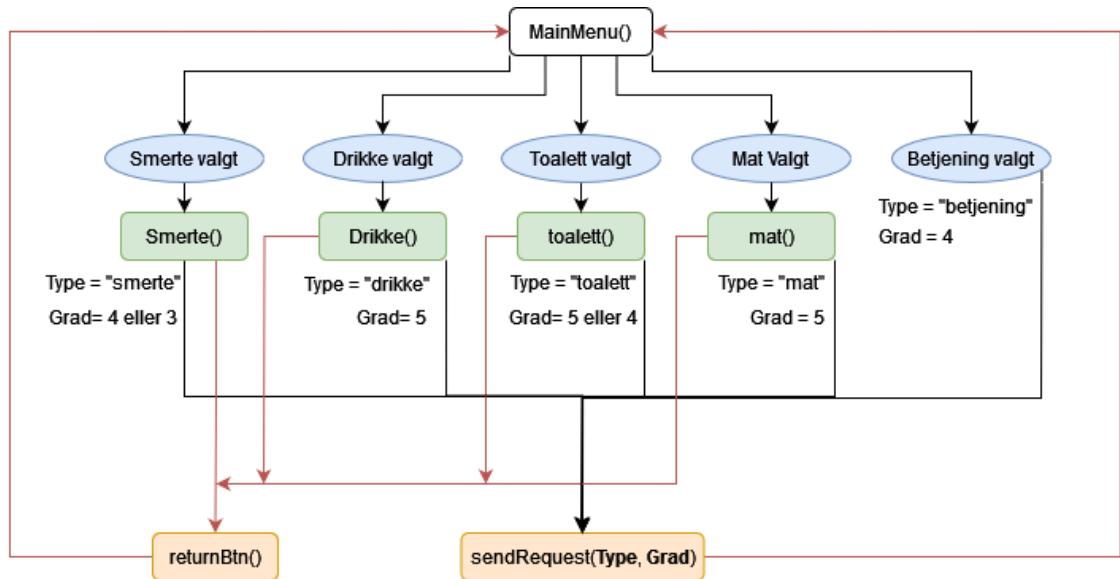
#### 4.4.3 UI

For design av UI blir pythonbiblioteket `Tkinter` [10] nyttet, ettersom det er ett simpelt bibliotek som oppyller våre krav til UI design og funksjonalitet. Der finnes to ulike UI design i prosjektet, ett til styreenheten og ett til brukerenheten.

##### **UI-brukerenhet**

Menyen som kjøres ved oppstart er Moduler, og er menyen som gir tilgang til alle undermenyer og funksjoner. Det finnes fire ulike Moduler som forklart i seksjon 3.2. I tillegg er funksjonaliteten til disse modulene allerede beskrevet i Design 3.

Et flytdiagram over funksjonene til hovedmenyen er vist i Figur 36, og er basert på designet Modul 3 (vanskelig) vist i seksjon 3.2.3 med tilhørende undermenyer. Dette flytdiagrammet vil også kunne anvendes lett og middels modusene, ved kun å implementere valgene presentert i 3.2.1 og 3.2.2.



Figur 36: Flytdiagram som viser funksjonene som blir nyttet i hovedmenyen.

Hovedfunksjonen som oppretter denne menyen er `MainMenu()`. Når denne funksjonen kjøres vil først alle elementer på skjermen slettes og noen variabler blir satt som vist i listing 6.

```

1 def MainMenu():
2     # Sier at ikke sykepleier menyen er åpen
3     inNurseMenu = False
4     # Sletter alle element fra skjermen
5     for widget in root.winfo_children():
6         widget.destroy()
7     # Forteller hvilken menu vi er inne i
8     currentMenu = "MainMenu"
9     # Hvilkmeny som er valgt
10    currentMainMenu = "MainMenu"

```

Listing 6: Kode som setter opp layouten til Styreenheten.

Deretter tegnes de ulike knappene på hovedmenyen utifra skjematikken vist i Figur 36. Hver av de ulike tegnes på samme måte, bare med ulik funksjon som blir kjørt ved trykk. Listing 7 viser hvordan Smerteknappen tegnes.

---

```

1 b1 = tk.Button(
2     root, # Velger området det skal tegnes på
3     text = "Smerte", # Tekst på knapp
4     bg = "#CD6A1F", # Farve på knapp
5     fg = buttonTextColor, # Tekst farve
6     command= Smerte, # funksjonen som skal kjøre ved trykk, i dette
    ↵ tilfellet Smerte()
7     font = button_font, # Font til tekst på knapp
8     height = button_height, # Høyde på knapp
9     width = button_width # Bredde på knapp
10 )

```

Listing 7: Kode til tegning av knapp, som i dette tilfellet fører til underkategorien smerte.

Ved hjelp av koden i listing 6 og 7 er dermed mulig å tegne opp alle hovedmenyene og undermenyene som vist i diagrammet på figur 36. Men for å kunne sende en forespørrelse fra brukerenheten til styreenheten nyttet funksjonen `sendRequest(request, hastegrad)`, denne funksjonen tar inn hvilket type varsel og hastegrad som skal sendes til styreenheten. Listing 8 viser hva variabler som blir initialisert i funksjonen `sendRequest`.

```

1 def sendRequest(request, hastegrad):
2     removeRequest = False # Om varselet skal fjernes fra
    ↵ styreenheten
3     requestExists = False # Om et tilsvarende varsel eksisterer
4     showUserMessage = "" # Beskjed som skal bli sendt på skjerm til
    ↵ pasient
5
6     # Lager melding som skal sendes til styreenhet på rett format
7     message = f"{rom},{seng},{request},{hastegrad},{user[lastUser]}"
8     # Gjør om message til en dictionary
9     messageDict = {
10         "Rom": int(receivedData[0]),
11         "Seng": int(receivedData[1]),
12         "Hva": receivedData[2],
13         "Hastegrad": int(receivedData[3])}

```

Listing 8: Initialisering av varable til funksjonen `sendRequest()`.

Deretter kjøres det ulike sjekker for å finne ut om varselet har blitt sendt før, om det skal fjernes og om det er ett SOS”varsel. Dette er vist i listing 9 under.

---

```

1 if request == "SOS" and hastegrad != 2:
2     remove = True
3     hastegrad = 2
4 if hastegrad != 0:
5     # itererer over alle varsel i listen requestRoom
6     for i in requestsRoom:
7         #sjekker om der finst et varsel fra tidligere som er likt
8         if (i.get('Rom') == Rom and i.get('Seng') == Seng and
9             i.get('Hva') == Hva):
10            # Endrer det gamle varselet dersom det nye varselet har
11            # lavere hastegrad.
12            if Hastegrad < i.get('Hastegrad'):
13                #setter brukerbeskjed til sendt
14                showUserMessage = "sent"
15                i['Hastegrad'] = Hastegrad #oppdaterer hastegrad
16                #dersom brukerenheten er tilkoblet sytreenhet
17                if not clientEnable:
18                    showSent() #vis beskjed til bruker
19                    returnButton() #returner til hovedmeny
20                    requestExists = True #varsel eksisterer allerede
21                # Dersom varsel eksisterer legg til nytt varsel i
22                # requestRoom listen
23                if requestExists:
24                    requestsRoom.append(messageDict)
25                    showUserMessage = "sent"
26                    # Viser beskjed til bruker og returnerer til hovedmeny
27                    if not clientEnable:
28                        showSent()
29                        returnButton()
30                # Fjerner varsel fra requestRoom listen
31                elif removeRequest:
32                    requestsRoom.removeRequest(messageDict)
33                    message = f"{rom},{seng},{request},-1,{user[lastUser]}"
34                    showUserMessage = "removed"
35                    # Beskjed til bruker om at varsel er fjernet og
36                    # returnerer til hovedmeny
37                    if not clientEnable:
38                        showRemoved()
39                        returnButton()
40
41 # Dersom varselgrad er lik 0, betyr det at varselet skal fjernes
42 else:
43     useData(f"{request},{rom},{seng}")
44     show = "removed"
45     remove =True

```

Listing 9: Koden i `sendRequest()` som avgjør hva som skal gjøres med varselet.

---

Etter det kan varselet sendes til styreenheten med funksjonen `sendData(message)`, og tilbakemelding er blir gitt til bruker på at varsel er sendt. Denne koden er vist i listing 10.

```
1 # Dersom brukerenheten er koblet til en styreenhet
2 if clientEnable:
3     # Send varsel til styreenhet
4     sendData(message)
5     # Gir pasient informasjon om at varsel er sendt
6     if show == "sent" and request != "SOS":
7         showSent()
8         returnButton() # Returnerer til hovedmeny
9     # Gir pasient informasjon om at varsel er fjernet
10    elif show == "removed" and request != "SOS":
11        showRemoved()
12        returnButton() # Returnerer til hovedmeny
13    # Kjøres dersom varselet "SOS" skal fjernes
14    else show == "removed":
15        returnButton() # Returnerer til hovedmeny
```

Listing 10: Kode for `sendRequest()` for sending av data.

---

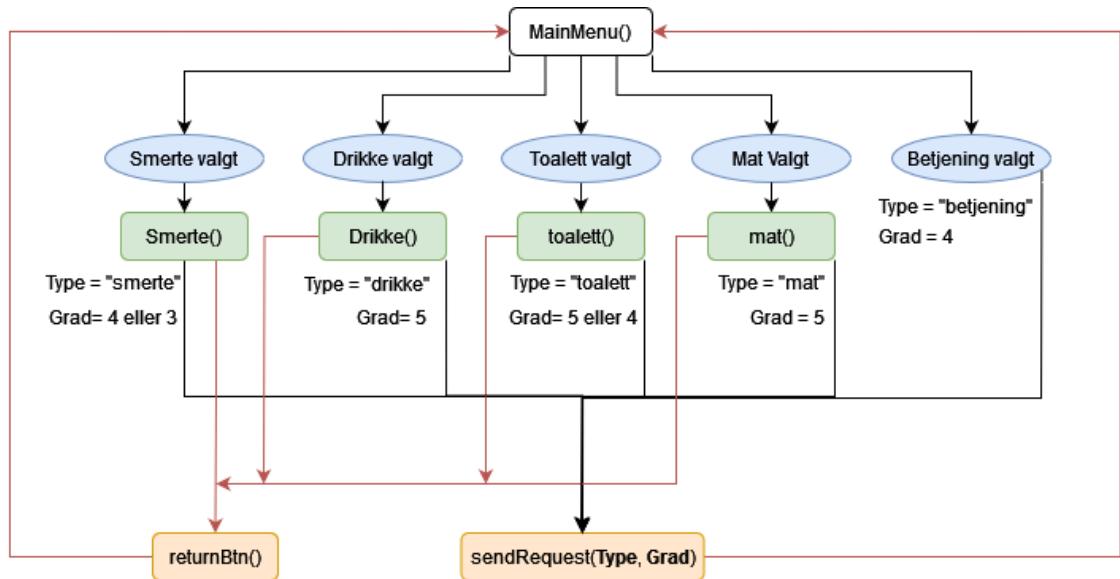
## UI- sykepleiermeny

For at sykepleierne skal kunne administrere brukerenheten, implementeres det en sykepleiermeny som beskrevet i seksjon 3.2.4. Denne menyen lages av funksjonen `MainNurseMenu()` og skal kun kjøre dersom ett gjenkjent RFID-kort har blitt scannet. Til denne gjenkjenningen nytes funksjonen `useRFID` som blir opprettet i en ny tråd som gjør det mulig å kontinuerlig søke etter RFID-kort mens andre ting skjer på skjermen. Koden for dette er vist i listing 11.

```
1 #Oppretter ny tråd for søking av RFID-kort
2 RFIDthread = threading.Thread(target=useRFID)
3 RFIDthread.start()
4
5 def useRFID():
6     # variabler for lagring av siste kort og bruker
7     global lastCardScanned, lastUser
8     read = True
9     start_time = time.time() #start tid
10    card = 1
11    while True:
12        elapsed_time = time.time() - start_time
13        # Sjekker etter RFID-kort hvert 0.5s
14        if elapsed_time < 0.5:
15            read = False
16        else:
17            read = True
18        if read:
19            card = RFID.getCardId() #Skaffer kort-ID
20            lastCardScanned = card #lagrer siste kort
21            start_time = time.time() #oppdaterer tid
22            #sjekker om kortet er lagret fra før av
23            if card in acceptedCards:
24                lastUser = acceptedCards.index(lastCardScanned)
25                mainNurseMenu() #opner sykepleiermeny
26
```

Listing 11: Kode for `useRFID()` for søking av RFID-kort.

Etter et RFID-kort som eksisterer i listen `acceptedCards`, kjøres funksjonen `mainNurseMenu()` og sykepleiermenyen opprettes. Struktureringen av sykpleiermenyen er vist i figur 37.



Figur 37: Flytdiagram som viser funksjonaliteten til sykepleiermenyen.

Ettersom mye av oppsettet og strukturen til sykepleiermenyen ligner på hovedmenyen, kan koden til hovedmenyen som vist i listing 36 og 6 gjenbrukes kun med endringer på knappene og fuknsjonen de kjører.

Som vist på diagrammet i Figur 37 kjøres funksjonen `MainAdminMenu` når knappen 'Annet' trykkes, denne funksjonen åpner en undermeny som skal gi sykepleierne alternativene til å endre følgende:

- `changeRoom()` - rommet bukerenheten befinner seg skal kunne velges
- `changeBed()` - dersom det er flere sengerposter på et rom skal det kunne velges hvilken av de brukerenheten tilhører.
- `addOrRemoveCard()` - sykpleierne skal kunne legge til RFID-kort til listen for godkjente RFID-kort.
- `changeDifficulty()` - nivået på hovedmenyen skal kunne velges.
- `quitFunc()` - lukker programmet

Menyene som lages av funksjonene `changeRoom()` og `changeBed()` tegnes på samme måte som `MainMenu()` med koden som vist i listing 36 og 6. Med knapper plassert utifra designet vist i Figur 26. Når en knapp blir trykket i en av disse menyene vil variablene `room` og `bed` oppdateres med ny verdi fra knappen.

Funksjonen `changeDifficulty()` oppretter en undermeny som med tre knapper; lett, middels og vanskelig. Som henholdsvis kjører funksjonene til `mainEasyMenu()`, `mainMediumMenu()` og `MainMenu()`. Slik at hovedmenyen som er valgt til blir endret.

Funksjonen `quitFunc()` kjører `root.quit()` som lukker programmet.

---

## UI-SOS

Siden SOS knappen er en fysisk knapp som alltid skal være tilgjengelig for brukeren, blir det opprettet en ny tråd med `threading` som kontinuerlig kjører funksjonen `SOSbuttonFunction()`. Denne funksjonen skal lytte etter når SOS-knappen trykkes ned, koden til denne er vist i listing 12.

```
1 def SOSbuttonFunction():
2     #Setter rette tall på GPIO-pinene
3     GPIO.setmode(GPIO.BCM)
4     #Setter GPIO-pin 26 til input
5     GPIO.setup(26, GPIO.IN)
6     if GPIO.input(26) == GPIO.HIGH:
7         #Sender varsel til styreenhet
8         sendRequest("SOS",2)
9         #Tegner et SOS-varsel på skjermen til brukerenheten
10        SOSpopup()
11        time.sleep(1) # Venter ett sekund
12    else:
13        time.sleep(0.1) #Venter 0.1s
```

Listing 12: Kode for `SOSbuttonFunction()`. Denne funksjonen skal lytte etter når SOS-knappen trykkes ned.

---

## UI styreenhet

Styreenheten tar utgangspunkt i designet i seksjon 3.1. Der hovedlayouten blir satt opp av koden i listing 13 på følgende måte:

1. Først opprettes det ett Tkinter vindu ved kodesnutten fra linje 1-6.
2. Deretter opprettes hver et tomt område for hver av delsystemene definert i 3.1, linje 8-21. I listing 13 blir bare plasseringen til varsellisten vist, plasseringen til resten kan finnes på Github [11].

```
1 root = tk.Tk()
2 root.title(pageTitle)
3 root.attributes('-fullscreen',True)
4 root.geometry(str(max_width) + "x" + str(max_height))
5 root["background"] = windowBackgroundColor
6 # Setter opp tomme rammer for hvert av de ulike UI elementene
7 # Varsel liste
8 left_frame = Frame(
9     root, #vinduet som det skal tegnes til
10    width=int(max_width*0.4 - 2*max_height*0.01), #bredde
11    height=int(max_height - 2*max_height*0.01), #høyde
12    bg='grey') # bakgrunnsfarve
13 left_frame.grid(
14     row=0, #Rad på root vinduet
15     column=0, # Kollone på root vinduet
16     rowspan=2, #Antall rader elementet skal strekke over
17     padx=int(max_height*0.01), #padding på side
18     pady=int(max_height*0.01)) #padding på topp og bunn
19 # Kart over rom
20 rightTop_frame = Frame(...)
21 rightTop_frame.grid(...)
22
23 # Varsel til kjøkken
24 rightBottom_frame = Frame(...)
25 rightBottom_frame.grid(...)
26
27 # Matvarsel meny
28 rightBottomRight_frame = Frame(...)
29 rightBottomLeft_frame = Frame(...)
```

Listing 13: Kode som setter opp layouten til Styreenheten.

---

Deretter kan grensesnittet tegnes for de ulike seksjonene:

- **Varselliste** - liste over varsel som må bli løst
- **Kart** - kart over alle rom på tunet, mulighet for å se varsel på de ulike rommene.
- **Varsel til kjøkken** - varsel som kan gå til kjøkkenet på avdelingen
- **Matvarsel** meny - til å sende varsel om måltid til pasientene

En liste med dictionary-element inneholder alle aktive forespørsler fra brukerenhetene som er koplet til styreeheten. Disse lagret på formatet vist under:

```
1   requests = {  
2     "Rom":305, #romnummer  
3     "Seng":1, #Hvilken seng  
4     "Hva":"Do", #Hva varselet handler om  
5     "Hastegrad":2, #Hastegrad til varselet fra 2 til 5  
6     "Tid":"12:46:09", #Når varselet ble sendt  
7     "Occupied":True, #Om det er en sykepleier på rommet  
8     "ID":2} # ID til varselet
```

Denne listen blir iterert over, og hvert element i listen brukes til å lage en knapp i varsellisten, og ett varsel på romkartet. Dette gjøres ved funksjonen `updateButtons()` og koden til denne er vist i listing 14. Se Figur 5 og 6 for eksempel på varselliste og kart.

---

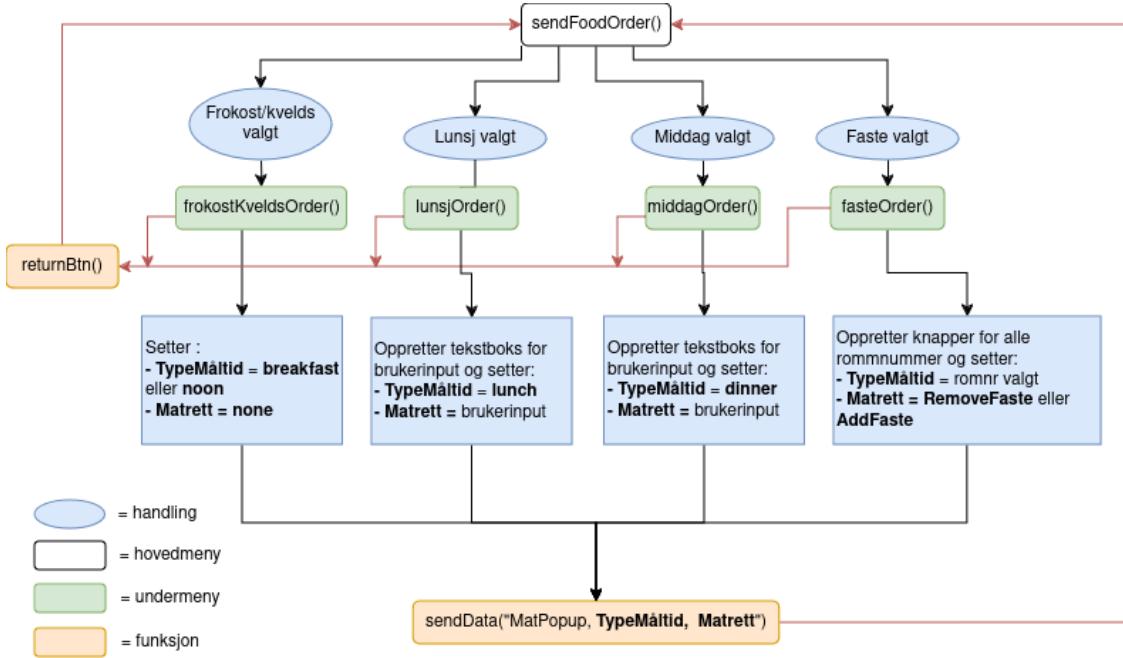
```

1 def updateButtons():
2     #Fjerner knappene som eksisterer
3     buttons.clear()
4     romMerking.clear()
5     #Iterer over listen med varsler og legger de til i varsellisten
6     for index, i in enumerate(requests):
7         buttons.append(tk.Button(left_frame,
8             command=lambda index=index: buttonFunction(index),
9                 ← #funksjon som kjører ved trykk
10                #Tekst på knappen
11                text=f"Rom {i.get('Rom')[:3]},{i.get('Seng')[:1]} Ønske:
12                    ← {i.get('Hva')[:8]} {i.get('Tid')[:8]}",
13                    font=("Consolas",16), #Fonttype
14                    bg = cf.color(i.get("Hastegrad")), #Farve etter
15                        ← hastegraden
16                        pady=10,padx=10))
17     #Iterer over de varselene med lavest hastegrad, og plasserer de
18     ← på kart og varselliste
19     for index,i in
20         ← enumerate(get_lowest_hastegrad_requests(requests)):
21         romMerking.append(tk.Label(rightTop_frame,
22             text="!", #Legger til utropstegn på kart
23             font=romMerkingFont,
24             bg=cf.occupiedColor(i), #Legger til bakgrunnsfarge
25             fg = cf.color(i.get('Hastegrad')), #Farger "!" etter
26                 ← hastegrad
27                 padx=romMerkingPadx, pady=romMerkingPady))
28     #Plasserer varsel på kartet etter plasseringen gitt av
29     ← listen roomPosition
30     romMerking[index].place(x=cf.roomPosition(i.get('Rom'))[0],
31                             y=cf.roomPosition(i.get('Rom'))[1])
32     for index,button in enumerate(buttons):
33         button.place(relx=0.0, rely=index*0.05, relwidth=1,
34             ← relheight=0.05)

```

Listing 14: Kode for `updateButtons`. Funksjonen iterer over listen med varlene på styreenheten og oppdater den.

For implementeringen av 3.1.3, mat og faste, tegnes først fire knapper med: Frokost/kvelds, lunsj, Middag og Faste. Knappene blir tegnet ved koden som vist i listing 7. Deretter velges det en funksjon som skal kjøre ved trykk på de ulike knappene. Når en ett valg er tatt, slettes de andre alternativene, og den henholdsvis valgte menyen i blir tegnet, dette er vist i flytdiagrammet i Figur 38.



Figur 38: Diagram over funksjonaliteten til mat og faste menyen på styreenheten.

Når 'Send' blir valgt, blir innfyllete data-en formatert som en string som brukerenheten kjenner igjen, og blir deretter sendt over bluetooth ved bruk av `sendData()` funksjonen som vist i listing 3.

For å se hele koden se [11] på GitHub

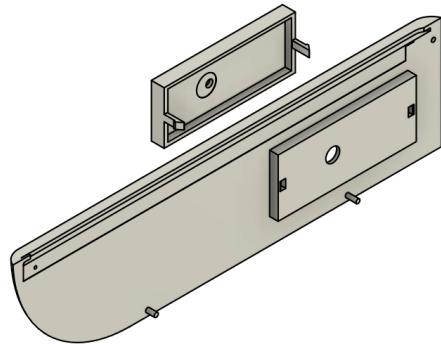
#### 4.4.4 Innfatning

Innfatningen til skjermen og RaspberryPi-en ble designet i modelleringsprogrammet Fusion360 [12], og består av fire ulike komponenter:

- Topplokk - som holder knappen, og låser skjerm på plass
- SOS knapp - Ett forstørret knapphode som festes til stammen på på knappen som er festet til topplokket. Dette gir brukeren en mye større overflate å tykke ned SOS-knappen.
- Hovedelementet - som holder skjermen, RaspberryPi-en og RFID-skanneren på plass.
- Håndtak høyre/venstre - som sammen med høyre håndtak er designet slik at brukerenheten skal kunne henge over rammen på en sykehusseng. Samt at de skal fungere som et hjelpemiddel for å holde brukerenheten i ved løfting.

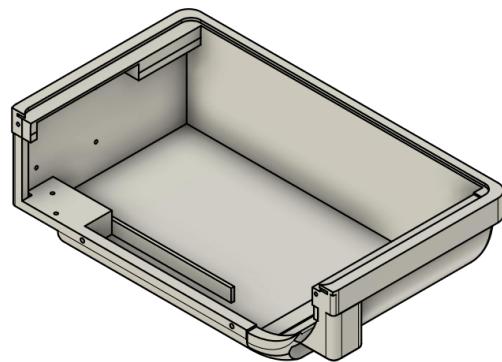
---

Alle komponentene ble designet slik at de kan monteres med kun skruer, og uten åpninger på utsiden av innfatningen. Det endelige designet av topplokket og SOS-knappen er vist i Figur 39.



Figur 39: (Øverst til venstre) SOS-knapp,(Nederst til høyre) topplokk med montering til knapp.

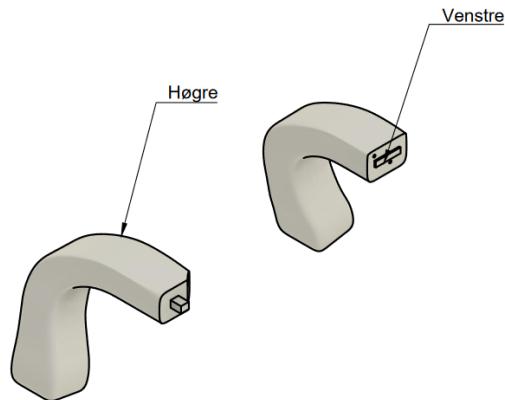
Designet for hovedelementet er vist i Figur 40.



Figur 40: Hovedelementet som holder på plass skjerm, RPI, RFID-skanner og powerbank.

---

Designet til håndtakene er vist i Figur 41.



Figur 41: Venstre og høyre håndtak.

Plantegningene for den ferdige modellen ligger i vedlagt som vedlegg A, samt ligger STL filene med designet på GitHub [11]

Videre ble modellen eksportert fra Fusion 360, til 3d-print-slicer-en PrusaSlicer [13] for klargjøring til 3d-printing. Modellen ble deretter printet på en *Prusa i3 MK3S+* [14] 3d-printer med en 0.4mm dyse i hvit materiale av typen PLA. Dette materiet ble valgt ettersom det er ett bionedbrytbart materiale, som er brytesterkt nok til vår dimensjonerte bruk [15]. En ulempe PLA har er ett lavt smeltpunkt, men denne er på ca. 180-200 °C altså langt over forventet brukstemperatur for vår situasjon.

Under i Figur 42 vises bilde av fullstendig prototyp med alle komponenter



Figur 42: Bilde av fullstendig implementert prototype.

---

## 5 Verifikasjon og Validering

Verifikasjonsplanen er laget for å kontrollere alle systemkrav. Utførelse av tester, hvilke systemkrav som testes og om testen er godkjent blir presentert i tabell 9, 10, 11, 12, 13 og 14. Alle bemerkningene følger med som kommentarer under tabellene. Refleksjon over funksjonalitet av det samlede systemet kommer til slutt.

Validering handler om å sikre at produktet møter brukerens behov og forventninger. Valideringsprosessen innebærer å teste systemet med minst seks deltagere. Testdeltakerne ble vikt ut i fra målgruppen til produktet.

### 5.1 Verifikasjon av systemkravene

I denne seksjonen skal systemkravene gitt i seksjon 2.3 verifiseres.

#### Brukervennlighet

Tabellen under inneholder tester for systemkravene til Brukervennlighet. Disse testene ble utført av gruppemedlemmer.

Tabell 9: Tester av systemkrav for brukervennlighet. Tabellen viser testnummer, utføring, tilhørende systemkrav og resultat.

Testnr.	Utførelsen	Testede Systemkrav	Godkjent?
I	En funksjonell nød-knapp på toppen.	1.1	Ja
II	Tre forskjellige vanskelighetsgrader: Enkel, middels og vanskelig.	1.2, 1.3, 1.4, 1.5	Ja
III	Modus 1 (Enkelt) har to valgalternativer: Hjelp nå og Hjelp senere.	1.3	Ja
IV	Modus 2 (Middels) har fire valgalternativer: Hjelp nå, Hjelp senere, mat og drikke.	1.4	Ja
V	Modus 3 (Vanskelig) har fire valgalternativer: Smerte, toalett, betjening, drikke og mat.	1.5	Ja

**Kommentarer:** Alle systemkravene er godkjent uten noen bemerkninger fra gruppa.

---

## Kommunikasjon

Tabell 10 inneholder utførelse og resultater av tester for systemkravene for Kommunikasjon. Disse testene ble utført av gruppemedlemmer.

Tabell 10: Tester av systemkrav for kommunikasjon. Tabellen viser testnummer, utføring, tilhørende systemkrav og resultat.

Testnr.	Utførelsen	Testede Systemkrav	Godkjent?
VI	Man kan differensiere følgende varsel på styreenheten; nød-situasjon, smerte, toalett, betjening, drikke og vann	2.1, 2.2	Ja
VII	Ved å skanne et godkjent RFID-kort gis det mulighet til å fjerne alle de eksisterende varslene	2.3	Ja
VIII	Det er forskjell på hastegradene ved valgene: mat, smerte og nød.	2.1, 2.2	Ja
IX	Brukerenheten kan lese RFID-kort	2.4	Ja
X	Det er en egen sykepleier modul på brukerenheten.	2.5	Ja

**Kommentarer:** Alle systemravene for Kommunikasjon har blitt testet og resultatarene er vist i 10. Ingen bemerkninger.

## Robusthet

Tabell 11 inneholder utførelse og resultater av tester for systemkravene til Robusthet. Disse testene ble utført av gruppemedlemmer.

Tabell 11: Tester av systemkrav for Robusthet. Tabellen viser testnummer, utføring, tilhørende systemkrav og resultat.

Testnr.	Utførelsen	Testede Systemkrav	Godkjent?
XI	Brukerenheten kan desinfiseres ved bruk av aloholbasert desinfeksjon.	3.1	Delvis
XII	Brukeenheten slippes fra en meters høyde uten å knuse.	3.2	Delvis/ikke testet
XIII	Brukeenheten trenger ikke å lades på 5 timer med aktiv bruk.	3.3	Ja
XIV	Brukeenheten trenger ikke å lades på 24 timer med vanlig bruk.	3.4	Ja
XV	Styreenhet og brukerenhet kan kommunisere på 10 meters avstand.	3.5	Ja
XVI	Brukerenhet skal opprettholde sin funksjonalitet over 3 år.	3.6	Ikke testet

---

**Kommentarer:** Den første bemerkningen gjelder systemkravet 3.1 om desinfeksjon. Selv om brukerenheten kan desinfiseres med vanlig antibak serviet, vil ikke plastikken på brukerenheten tåle store mengder med alkohol/antibak som er brukt på sykehuset. Produktet er heller ikke vanntett som kan føre til kortslutninger.

Siden produktet skal framvises på konferansen, kan ikke testen gjennomføres uten fare for å ødelegge produktet. Det har derimot skjedd uhell der brukerenheten har falt ned fra rundt en halv meter høyde uten å ta skade.

Siste bemerkningen er operasjonstiden for produktet. Det er uoppnåelig å teste hvor lenge produktet kommer til å holde, på grunn av tiden det tar.

## Informasjon

Tabell 12 inneholder utførelse og resultater av tester av systemkravene til Informasjon. Disse testene ble utført av gruppemedlemmer.

Tabell 12: Tester av systemkrav for Informasjon. Tabellen viser testnummer, utføring, tilhørende systemkrav og resultat.

Testnr.	Utførelsen	Testede Systemkrav	Godkjent?
XVII	Systemet kobles kunne koples til et sikkert trådløst nett.	4.1	Nei
XVIII	Styreenhet kan skille mellom de forskjellige brukerenhetene	4.2	Ikke testet
XIX	Brukerenhet og styreenhet kan kommunisere med hverandre over Bluetooth	4.3	Ja

**Kommentarer:** På grunn av vansker med å koble til EDUROAM, som er konto-basert internett den man logger inn med personlig bruker, er det besluttet å bruke Bluetooth istedenfor. På grunn av det er systemkravet 4.1 om internett og testen XVIII, ikke godkjent.

På grunn av begrenset budsjett, er det kun implementert én brukerenhet. På grunn av mangelen på flere brukerenheter, har ikke systemkrav 4.2 blitt testet.

---

## Bærekraft

Tabell 13 inneholder utførelse og resultater av tester for systemkravene til Bærekraft. Disse testene ble utført av gruppemedlemmer.

Tabell 13: Tester av systemkrav for Bærekraft. Tabellen viser testnummer, utføring, tilhørende systemkrav og resultat.

Testnr.	Utførelsen	Testede Systemkrav	Godkjent?
xx	Brukerenheten kan demonteres til enkelte deler, og alle deler kan lett byttes ut.	5.1, 5.2	Ja
xxI	Systemet er koblet på det allerede eksisterende varselsystemet.	5.3	Nei/Ikke testet

**Kommentarer:** Alle delene i produktet kan demonteres med skrujern som verifiserer systemkrav 5.1 og 5.2 om bærekraft. På grunn av manglende internett tilkobling kan ikke systemet integreres inn i det allerede eksisterende systemet på sykehuset, slik at test xxii ikke kan godkjennes. Likevel kan systemet brukes som separat system ved siden av, uten å ta erstattet det gamle.

## Matbestilling

Tabell 14 inneholder utførelse og resultater av tester for systemkravene til matbestilling. Disse testene ble utført av gruppemedlemmer.

Tabell 14: Tester systemkrav for Matbestilling. Tabellen viser testnummer, utføring, tilhørende systemkrav og resultat.

Testnr.	Utførelsen	Testede Systemkrav	Godkjent?
XXII	Kan man se matbestillingsvarslene fra kjøkkenet?	6.1	Ja
XXIII	Kan matbestillinga sendes fra styreenheten til brukerenheten for frokost og/eller kveldsmat ?	6.2	Ja
XXIV	Kan brukerenheter som er merket med ”faste” se matvarsel?	6.3	Ja

**Kommentarer:** Ingen bemerkninger.

## Refleksjon om verifikasi

Gjennom testing av systemets fysiske og brukskravene er de fleste systemkravene godkjent, med unntak av noen få. Systemkravene 3.2, 3.6 og 4.1 er ikke blitt testet, på grunn av maglende implementering eller tidsbegrensning.

---

## 5.2 Validering av brukerkrav

Brukertesten tar for seg test av hele produktet og tar utgangspunkt i brukerkravene som er utredet i seksjon 1.1.

I tabellen under er det resultatene for de gjennomførte brukertestene presentert. Testdeltakerne ble valgt ut for å ha mest mulig like forutsetninger som de faktiske brukerne av systemet, ut i fra alder, erfaring med sykehus og kompetanse. Hvert brukerkrav er validert fra 0 til 5, der null er ikke oppfylt og fem er oppfylt uten noen kommentarer. I tillegg kan man se ”-“ som betyr at det er ikke tatt hensyn til under testing. Testdeltakerne fikk teste systemet under observasjon og svarte deretter på spørsmål ut ifra brukerkrav og opplevelsen av produktet. Nøkkelinformasjon fra testene er beskrevet i følgende seksjon.

### A. Tilgjengelighet

For å validere følgende brukerkrav har testdeltagerne blitt spurtt om:

- A.1 - Hvor lett er det å sende nødvarsel til en hvert tid ?
- A.2 - I hvor stor grad er løsningen alderstilpasset?
- A.3 - I hvor stor grad er løsningen tilpasset til de som har vansker med syn?
- A.4 - I hvor stor grad er løsningen tilpasset til de som har vansker med bevegelse?
- A.5 - Hvor universal løsning er?

Tabell 15: Brukerkravverifikasjon av tilgjengelighet med gitt kvantitativt vurdering.

Brukerkrav	Test nummer						Gjennomsnitt
	1	2	3	4	5	6	
A.1	4	4	4	4	5	5	4.3
A.2	5	5	5	4	4	5	4.7
A.3	4	5	3	2	3	3	3.3
A.4	5	5	4	4	3	4	4.2
A.5	5	3	5	4	4	5	4.3

#### Kommentarer:

Ved testing av A.1 har ikke knappen være verdig installert, dermed har testdeltakerne påpekt at knappen ikke var til stede. I tillegg til det har det blitt nevnt at nød-knapper burde ha et beskytelses lokk for å ikke komme bort i knappen med uhell.

For brukerkravet A.2, er brukerenheten kun tilgjengelig i et språk, som gjør det mindre universal i følge at testdeltakerne.

---

## B. Kommunikasjon

I denne delen har deltagerne svart på følgende spørsmål:

- B.1 - Hvor godt formidler produktet (brukerenhet) om det forskjellige varslene?
- B.2 - Hvor lett er det å se forskjell på varslene (styrenehett)?
- B.3 - Hvor lett er det å se om noen av varslene er tatt?
- B.4 - Kan man høre forskjell på de forskjellige varslene?
- B.5 - I hvor stor grad avlaster sykepleiene fra matbestilling?
- B.6 - Hvor synlig er det at noen faster?

Tabell 16: Brukerkravverifikasjon av kommunikasjon med gitt kvantitativt vurdering.

Brukerkrav	Test nummer						Gjennomsnitt
	1	2	3	4	5	6	
B.1	4	4	5	5	4	5	4.5
B.2	4	4	5	5	4	4	4.3
B.3	5	5	5	4	4	4	4.5
B.4(ikke implementert)	0	0	0	0	0	0	0
B.5	-	-	4	4	4	5	4.3
B.6	-	-	2	4	4	5	3.75

### Kommentarer:

Først og fremst har det blitt nevnt at det er intuitivt å finne fram til ting men informasjonsarkitekturen kunne vært bedre, dette gjelder B.1 og B.2. Brukerenheten mangler lyd, B.4, for varslig og synliggjøring av bestilinger.

## C. Bærekraft, D. Robusthet, E. Informasjon

- C.1 - I hvor stor grad kan produktet resirkuleres?
- C.2 - Hvor kompatibel systemet er med allerede eksisterende system?
- D.1 - Hvor lett er det å stjele produktet ( brukerenhet)?
- D.2 - Hvor vanskelig er det å knuse produktet (1-lett, 5 -vanskelig)?
- D.3 - Hvor lenge kan en slik produkt være? ( 1-kort, 5-langt periode)
- E - I hvor stor grad opprettholder produktet alle taushetspliktene?

Tabell 17: Brukerkravverifikasjon av bærekraft, robusthet og informasjon med gitt kvantitativt vurdering.

Brukerkrav	Test nummer						Gjennomsnitt
	1	2	3	4	5	6	
C.1	5	5	5	5	3	5	4.7
C.2	3	3	3	3	4	2	3
D.1	3	3	3	3	4	3	3.2
D.2	4	4	4	4	2	4	3.7
D.3	5	5	5	5	3	5	4.7
E	5	5	5	5	4	5	4.8

Etter testene nr. 1 og 2 ble gjennomført har systemet blir forbedret for å ta hensyn til problemene som har oppstått under testing. Dette ble gjort ved og teste alle funksjonene og ulike kombinasjoner av varslinger på systemet. Programvarefeil som dukket under første testene bestevet i tabell 18 sammen med løsningene som ble implementert.

Tabell 18: Problemene som har blitt løst under eller etter brukertester.

Problem	Løsning
Skriftstørrelse på varsel styreenhet passer ikke innenfor rammene.	Skriftstørrelse byttet fra 18 til 16 i left-Frame i ControllerGUI.
Posisjonen på kartvarsling er litt skjevt i forhold til kart.	Posisjonen returnert av roomPosition i controllerFunctions.py er justert til å passe med kartet.
Man kan bruke andre knapper i sidekanlene selv om SOS popup er fremme.	SOS varsel ble satt fra å dekke 80% til 100% slik at knappene på siden ikke er tilgjengelige.
Rom med faste har mulighet til å sende mat ønsker.	Når faste blir sendt til brukerenhet blir knappene for mat og drikke fjernet fra menyen.
Frokos og kvelds bestilling forsvinner etter 1 sekund i enkel og medium versjon, samt bestilling sender personen tilbake til vanskelig versjon.	Returnbutton funksjonen lå inne i popup-funksjonen og ble nå fjernet. I tillegg ble det lagt en ny funksjon for mat med navn mat-FrokostKvelds som ikke inneholder endring av currentMenu variabel.

---

## Tilbakemelding fra ortopedisk avdeling

Det ble også gjennomført et møte med ortopedisk avdeling når prototypen var tilnærmet ferdigstilt. Under møtet var sykepleierene positive til produktets funksjonalitet, og kunne se for seg flere områder produktet kunne hjelpe avdelingen med hverdagen. I tillegg kan produktet hjelpe med å fordele kompetansen til sykepleierne bedre.

Det ble også foreslått fra sykepleierne at de gjerne ville at prototypen skulle videreføres til et produkt som kunne implementeres på sykehuset. Ettersom at de ikke har noe lignende system tilgjengelig og at produktet har potensialet til å redusere kostnader på avdelingen, samt redusere stresspotensialet blant sykepleierne. Et samendrag av tilbakemeldingen som ble sendt på e-post fra sykepleierne ligger i vedlegg B under seksjonen **23. april**.

---

## 6 Diskusjon

Prosjektet er tidsbegrenset, det er derfor naturlig at ikke alle bruker- og systemkra-vene blir verifisert. I underseksjonen nedenfor vises diskusjon rundt videreutvikling av produktet, og diskusjon om hvordan brukerkarav og systemkrav er oppfylt og eventuelle forbedringer rundt de ulike kravene.

### Videreutvikling

Det er flere muligheter for videreutvikling av produktet, da det er både systemkrav og brukerkarav som ikke er møtt i nåværende prototype. Ulike områder som kan videreutvikles diskutes i denne seksjonen.

#### Programvaren

Programvaren til systemet har flere muligheter for å utvikle bedre løsninger. Bruker-kravene A.3, A.4 og A.5, se tabell 1, er i stor grad oppfylt, men tilretteleggingen for blinde og synssvake mangler fortsatt. Ved å gjøre systemet talestyrt vil det være mulig for blinde, svaksynte og ekstremt bevegelseshemmde å utnytte av brukerenheten bedre. Dette vil også forenkle varslingen av behov fra pasienten i flere situasjoner, for eksempel dersom en pasient skulle falt på bakken og ikke får tak i brukerenheten.

I seksjon 2 ble det i Figur 1 foreslått et design for brukerenheten. Dette designet inneholdt bilder som er tenkt å gjøre det enda mer intuitivt for pasienten hva de ulike varslene beskriver. Dette vil være med på å fjerne eventuelle språkbarrierer for pasienter av andre nasjonaliteter eller med lesevansker, og som da vil oppfylle brukerkarav A.5 i større grad.

Alle sykepleiere er utstyrt med en egen PDA, personal digital assistant, som de har på seg til en hver tid under vaktene. Å få applikasjonen implementert på disse vil gjøre det mulig for hver PDA å operere som en egen styreenhet. Dette vil igjen effektivisere arbeidshverdagen og problemstillingen blir løst i enda større grad.

Slik prototypen er nå, blir ikke brukerkarav B.4 oppfylt. Det er ikke implementert komponenter som tillater systemet å gi fra seg lyd. Ved å legge til høyttaler hos styreenheten og endre programvaren slik at hver varsel har en egen lyd, vil produktet oppfylle brukerkaravene B.1, B.2 og B.4 i en større grad. Utfordringen med det kan være lydtilpasning. På ortopedisk avdeling og sykehuset generelt er det en del støy å lyder fra alt utstyr de tar i bruk. For å implementerer lydvarsling på styreenheten, må det gjøres en grundig analyse på hvilken lyd burde brukes. Dette betyr at lyden må være tydelig og skille seg godt ut, uten å være forstyrrende eller stressende for pasienter og personale. På brukerenheten trengs det derimot ikke like avansert lydteknologi siden den primært vil fungere i pasientens nærområdet, hvor det typisk er mindre bakgrunnsstøy.

#### Innfatningen

Systemets innfatning kan møte brukerkaravene A.1 og A.3, se tabell 1, i større grad. Dette kan gjøres ved integrere blindeskrift i SOS-knappen på brukerenheten, slik

---

at blinde og svaksynte pasienter har mulighet til å sende ut nødvarsel til enhver tid fra brukerenheten. Det er implementert uthevet skrift som kan gjenkjennes med fingerene, noe som gjør at dette ikke er en prioritering men et mulig tillegg om det lar seg gjøre.

Brukerenhetens innfatning oppfyller ikke brukerkaravene i kravtype D om robusthet i sin helhet. Brukerkrav D.1 fastsetter at brukerenheten skal være beskyttet mot stjeling, slik at det ikke er mulig for en pasient å få med seg brukerenheten hjem ubemerket. Dette brukerkaravet kan tilfredsstilles ved å sette inn en sporingsenhet i brukerenheten, da vil sykepleierne ha oversikt over alle brukerenhetene til en hver tid. Skal denne enheten implementeres er det viktig at brukerkarav E.1 om taushetspliktene fortsatt er bevart.

Brukerkrav D.2, at produktet skal beskyttes mot deformering og ødeleggelse, legger til rette for systemkaravene 3.1 og 3.2, se tabell 4. Ved utviklingen av prototypen for brukerenheten er det tatt noe hensyn til nevnte systemkarav, men grunnet fokus på bærekraft har det vært vanskelig å tilfredsstille. Brukerenhetens prototype er, som nevnt i underseksjon 4.4.4, 3D-printet i PLA (polymelkesyre), et bionedbrytbart materiale som er sensitivt for alkohol. Ved å lage innfatningen av et annet materiale som fortsatt er bærekraftig, men også tåler alkohol, vil det kunne desinfiseres uten å ta skade. Forslag til eventuelt materiale kan være PETG eller PP [16]. En annen alternativ kan være å bruke beskytelses lag over PLA-en, som for eksempel ett silikonlag eller akryllag. Det vil sikre at brukerenheten er vanntett og gjør den mer robust mot fall.

## Diskusjon av resultater

### Verifikasjon av systemkarav

Det har totalt sett blitt gjennomført 24 verifikasjonstester av systemkaravene. Alle systemkaravene for brukervennlighet som gjelder systemkarav 1.1-1.5 er godkjent uten noen bemerkning. Det samme gjelder for kommunikasjon systemkarav 2.1-2.5. Graderingen for testene har vært på en diskret skala fra 1-5.

Det har blitt diskurert tidligere at PLA, som har blitt brukt som hovedmaterialet i 3D-printingen, ikke er veldig godt egnet til disinfeksjon med alkoholbasert væske, ettersom PLA er mottakelig for smelting ved bruk av alkohol. Dermer har systemkaravet 3.1 ,for desinfeksjon, blitt delvis godkjent. Det er fremdeles mulig å rense enheten, men ikke i den graden det blir gjort på sykehuet. Av hensiktsmessige grunner er ikke test nr. 12, som går ut på å teste falltoleranse, gjennomført ”skikkelig”. Det vil si at brukerenheten fallt fra boret uten noen fysiske eller tekniske konsekvenser, men vi har ikke testet en meters fall. Det er også ikke mulig å teste levetiden for produktet, men ut i fra informasjonen i databladene til komponentene vil brukerenhet kunne bli brukt over 3 år, uten behov for å bytte komponenter.

Systemkaravene for informasjon 4.3, for Bluetooth kommunikasjon, har blitt godkjent. Systemkaravene 4.1 og 4.2 har ikke blitt implementert. Siden brukerenheten er laget av flere ferdige komponenter og satt sammen av skuer er det enkelt å bytte ut alle komponenter i systemet dersom det oppstår en defekt, noe som godkjerner karavene

---

5.1, 5.2 i bærekraft, se test 20. For test 21 derimot, har ikke systemet blitt koblett sammen med det allerede eksisterende varslingssystemet på sykehuset, dermed er ikke systemkravet 5.3 godkjent. Alle systemkravene for matbestillingen er godkjent.

Totalt sett er 18 av 24 tester godkjent og 2 er delvis godkjent. Dette tyder på at systemet fungerer som det har blitt satt krav til å fungerer i stor grad.

### **Validering av brukerkarav**

Andre delen av testingen var validering av brukerkarav. For brukerkaravet A. tilgjengelighet ligger gjennomsnittlig resultater over 4, utenom brukerkaravet for tilpasning for alle syn. Det gjelder det samme for brukerkaravet B. kommunikasjon. B.6 som spør om hvor synlig det er at et rom har fast har en gjennomsnittlig vurdering på 3.75 av 5. Det kan forklares med at testdeltagerne er kjent med styreenheten fra før av, og ikke blitt informert om interface. Brukerkaravverifikasjon av bærekraft, robusthet og informasjon viser tilstrekkelige gode resultater med tanke på implementeringen som har blitt diskuter tidligere.

Gjennomsnittlig vurdering av det helhetlige produktet er 4.1 av 5 som viser at produktet tilfredstiler de fleste system- og bruker-karavene.

---

## 7 Konklusjon

I dette prosjektet er det utviklet en prototype med fungerende system- og maskinvare, som skal bidra til å effektivisere og forenkle kommunikasjon mellom sykepleierne og pasientene ved St. Olavs hverdag ved å differensiere varslesignalene. Løsningen er utviklet fra problemstilling som ble formulert etter besøket på avdelingen ved St. Olavs der noen av utfordringene de hadde ble oppdaget.

På brukertestene med en gradert diskrett skalla fra 1 til 5, har produktet fått totalt snitt på 4.1 av 5. Ved verifikasjon av systemkravene har 18 av 24 testene blitt godkjent. Dette tyder på at produktet oppfyller alle system- og bruker-kravene i stor grad.

Men man ser også rom for tydelige forbedringer; for eksempel andre type materiale som PETG eller PP kunne blitt brukt, og tillegg av lydvarsling kunne blitt implementert. Man ser også at avdelingen produktet er laget til har vært svært positive til produktets funksjoner og har et ønske om at produktet blir implementert på avdelingen.

Produktet viser at konseptet er en god løsning på problemstillingen og at det er et produkt som er aktuelt å videreføre utvikling. På bakgrunn av system- og brukerkraftester, og tilbakemelding fra avdelingen på St. Olavs, er problemstillingen løst på en god måte med produktet *VarselVenn*.

---

## Referanser

- <sup>1</sup>P. Aursand, *Eldrebølgen skaper stor usikkerhet for helsesektoren*, (2023) <https://www.ssb.no/arbeid-og-lonn/sysselsetting/artikler/eldrebolgen-skaper-stor-usikkerhet-for-helsesektoren> (sjekket 11. apr. 2024).
- <sup>2</sup>E. Notatbasen, *Bærekraftige produkter*.
- <sup>3</sup>Components101, *RC522 RFID Module*, <https://components101.com/wireless/rc522-rfid-module>.
- <sup>4</sup>GP, *PowerBank MP10MA 10000mAh*, <https://international.gpbatteries.com/products/powerbank-mp10ma-10000mah?variant=32276551041123> (sjekket 4. mai 2024).
- <sup>5</sup>Aditya, *mfrc522-python*, <https://pypi.org/project/mfrc522-python/>.
- <sup>6</sup>P. Dhaker, *Introduction to SPI Interface*, <https://www.analog.com/en/resources/analog-dialogue/articles/introduction-to-spi-interface.html>.
- <sup>7</sup>N. J. Muller, *Networking A to Z* (McGraw-Hill Professional, 2002).
- <sup>8</sup>Socket, <https://docs.python.org/3/library/socket.html> (sjekket 9. apr. 2024).
- <sup>9</sup>Threading, <https://docs.python.org/3/library/threading.html> (sjekket 18. apr. 2024).
- <sup>10</sup>Tkinter, <https://docs.python.org/3/library/tkinter.html>.
- <sup>11</sup>G. 15, *ELSYS-Prosjekt Gruppe 15*, (2024) <https://github.com/PaulusKlaus/ELSYS-Prosjekt> (sjekket 4. mai 2024).
- <sup>12</sup>AutoDesks, *Fusion360*, <https://www.autodesk.com/products/fusion-360/overview?term=1-YEAR&tab=subscription>.
- <sup>13</sup>Prusa, *PrusaSlicer*, [https://www.prusa3d.com/en/page/prusaslicer\\_424/](https://www.prusa3d.com/en/page/prusaslicer_424/).
- <sup>14</sup>Prusa, *Prusa i3 MK3S+*, <https://www.prusa3d.com/category/original-prusa-i3-mk3s/>.
- <sup>15</sup>Xometry, *PLA (Polylactic Acid): Definition, Applications, and Different Types*, <https://www.xometry.com/resources/materials/what-is-pla/>.
- <sup>16</sup>P. P. Team, *Chemical resistance of 3D printing materials*, (2012) <https://prusament.com/chemical-resistance-of-3d-printing-materials/?fbclid=IwAR36ncgRh0z4-tr2aQaMiDdfHV-kgxL89Gu9hVJqExANXw-i1b4Tz-HjNE> (sjekket 1. mai 2024).

---

# Appendix

## A Behovsanalyse

Nedenfor, i tabell 19, er en oversikt over de viktigste utfordringene som ble identifisert ved besøk av St.Olavs Hospital:

Tabell 19: En oversikt over de forskjellige utfordringene hos Ortopedisk avdeling på St. Olav sykehus.

Utfording	Beskrivelse
Rekruttering	Vanskiligheter med å tiltrekke kvalifisert personale.
Sykefravær	Høyt fravær blant ansatte påvirker arbeidskapasiteten.
REGO 1	Kommunen strever med å følge opp pasienter etter sykehushusopphold.
Psykologisk trygghet	Utfordringer med samarbeid og ulik arbeidsmengde mellom avdelingene.
Stikkontakter	Mangel på nødvendige elektriske punkter for utstyr.
Skader	Økt risiko for fall blant pasienter som ikke ber om hjelp.
Informasjon	Pasienter mottar utilstrekkelig med informasjon.
Varslinger	Begrenset med varslingsalternativer for pasientene.
Trykkskader	Hyppig forekomst av trykkskader krever forebyggende tiltak.
Utstyr	Problemer med feilplassering og tilgjengelighet av utstyr.

Ut fra behovsanalysen i tabell 19 er det identifisert flere kritiske områder for forbedring. Ut i fra det har vi kommet fram til flere problemstillingen sammen med eventuelle løsningene. Liste over disse problemstilingene er gitt:

1. Optimalisering av sykepleiers tidsbruk: Effektivisering av arbeidsflyt for å redusere tidsbruk og stress.
2. Forhindring av feilplassering av utstyr: Implementering av et sporingssystem for utstyr for å sikre rask og korrekt tilgjengelighet.
3. Forbedring av varslingssystemet på sengepost: Utvikling av et mer avansert varslingssystem som kan prioritere pasienters behov basert på alvorlighetsgrad.
4. Forebygging av trykkskader: Implementering av bedre rutiner og teknologi for å forebygge trykkskader.
5. Forebygging av fallskader: Utnyttelse av sensorer eller annen teknologi for å detektere og forhindre fall blant pasienter.

Disse potensielle løsningene vil ikke bare forbedre pasientomsorgen, men også bidra til å gjøre helsevesenet mer attraktivt som arbeidsplass og effektivisere ressursbruk, noe som er kritisk i møte med den kommende eldrebølgen.

---

## B Tilbakemelding fra sykepleiene

Under design og implementering har vi hold kontinuerlig kontant med sykepleiene på ortopedisk avdeling. I denne seksjonen presenteres alle tilbakemeldingene vi har fått fra de gjennom prosjektet. Blå teksten representerer tilbakemeldingen vi har fått, mens svart er spørsmålne so ble stilt.

### 8. april

Vi har et ide som går ut på at tilkallingssystemet for pasienter er ikke optimalt. Det vil si at pasienten drar i en snor for å tilkalle hjelp. Denne snoren blir også brukt for alvorlige situasjoner. Ved å dra i en snor flere ganger kan gjøre at viktigheten med denne snoren blir svekket. Derfor ønskes det å lage et system hvor pasienten kan lettere melde fra til sykepleiere eller de ansatte hva behovet pasienten er. Med det menes at hvis pasienten kun ønsker vann, kan pasienten enkelt anvende systemet og informere at pasienten ønsker vann og det ikke er en alvorlig situasjon(altså de drar i snoren), dette systemet kan også anvendes slik at hva pasienten ønsker å spise den dagen kan bli loggført og innført i et datasystem istedenfor at en sykepleier eller en annen ansatt må manuelt inn å spørre og gå til hvert enkelt rom. Med et slikt system kan også pasienten få vite hvilke matretter som er på menyen og hvilke allergener, intoleranser etc som maten inneholder, trur du det hadde vært noe som hadde hjelpe dere?

Hvordan ville du tenkt at dette kan gjøres på best mulig måte? Hva syns du om vår løsning?

Som vi uttalte på den dagen da flere av dere studenter var hos oss på et informasjonsmøte, er det et ønske om å kunne skape et differensiert ringesignal. Dette kan være hensiktsmessig av flere grunner, både for pasientene og for helsepersonell. Så dette er en god ide om det lar seg løse. Hvordan dette skal løses er jeg ikke sikker på. Nå kan det se ut som om flere studentgrupper har fattet interesse for denne utfordringen og et par grupper har allerede vært i kontakt med oss for å vise løsningen de jobber med. Noen tenker at pasientene kan få et display/skjerm de kan trykke og gjøre valg på, andre har konstruert en boks med ulike knapper pasientene kan trykke på.

Hvilke funksjoner ville du hatt med i et slikt system?

Dette er et vanskelig spørsmål, men at pasientene kunne gradert hjelpen de ønsker er viktig. Dette kan man gjøre enkelt med for eksempel rødt/oransje/grønt signal. Hvis man begynner å analysere alt pasient ønsker, vil det potensielt bli mange knapper. Da er det enkle det beste. I tillegg tror jeg at en kommunikasjon med kjøkkenet hadde også løst masse. Der kunne pasientene lagt inn ønsker om hva slags mat og drikke de ønsker.

Er det noe spesielt, vi ikke har tenkt på selv?

Dere må tenke på at mange pasienter ikke er kognitivt friske. Skal de kunne få denne løsningen? I tillegg må dere se på hygieniske aspekter ved den fysiske løsningen –

---

den må kunne vaskes/desinfiseres uten å ta for mye skade. Dere må ta høyde for at pasienter kan ta med seg enheten hjem ved et uhell (eller stjele) og det kan bli kostbart for helsetjenesten.

Hvordan tror du at en pasient ville likt dette?

Jeg personlig tror pasientene ville likt en slik løsning. Mange pasienter føler de «fortyrrer» helsepersonell unødig og ved et differensiert ringesignal kan dette unngås.

Trur du denne produktet hadde påvirket kommunikasjon i positiv eller negativ retning?

Også et vanskelig spørsmål. Kommunikasjonen med pasientene er for sykepleiere ved sengepost av stor viktighet for behandlingen de får. Det er blant annet gjennom kommunikasjon vi identifisere problemområder hos pasientene. Så en slik løsning må ikke gå på bekostning av dette. Med ette som bakteppe, må man tenke at ønsket om hjelp fra pasientene ikke skal bli mindre, men heller mer målrettet.

Hvis det er greit så fører jeg deg inn kontakloggen som en spesialist innafor helsesektoren.

Det skal gå fint.J

Tusen takk på forhold

Bare hyggelig – og ta kontakt om noe er uklart.

## 23. april

Hei igjen Det er litt vanskelig å svare fra 0-5, da løsningen slik den er i dag har stort potensial, men likevel kan den trenge noen justeringer. Som for eksempel at den fungerer på flere språk, at informasjonsarkitekturen kunne vært noe annerledes Her er mine svar (0 = ikke bra/eksisterer ikke – 5 er veldig bra):

1. Det er enkelt å sende nødversell – 4 (det var en knapp som manglet eller var feilplassert, var det ikke?)
2. Det er tilpasset for alle alder – 5 (her uttaler jeg meg kun på vegene av voksne pasienter over 18 år, da dette er den pasientgruppen vi har)
3. Produktet skal være tilpasset til de som har vansker med syn – 5
4. Det er tilpasset for de som har vansker med bevegelse - 5
5. Produktet er tilpasset for flest mulig, nasjonalitetet –3 (hadde den flere språk?)
6. Produktet må synliggjøre eller formidle om de forskjellige oppgavene. –4 (her kunne informasjonsarkitekturen vært noe bedre)
7. Produktet skal differensielle mellom de forskjellige oppgavene. –4
8. Produktet skal formidle om noen har allerede tatt en oppgave. -5

- 
- 9. Produktet skal ha ulik lyd for foskjellige hastegrader ( Vi har ikke hatt tid til å implementere dette) - 0
  - 10. Produktet skal kunne resirkuleres. – dette husker jeg ikke om ble nevnt, men om produktet kan resirkuleres så: 5
  - 11. Produktet skal være beskyttet mot stjeling – dette husker jeg ikke om var håndtert. Den var jo løs, så den er nok lett å ta med seg? 3
  - 12. Produktet skal være beskyttet mot knusing – den virket solid -4
  - 13. Produktet skal kunne brukes over langt tidsperiode.- 5
  - 14. Produktet skal opprettholde alle taushetspliktene.-**dette er jeg også veldig usikker på, for her snakker man om taushetsbelagt informasjon både for pasient og sykepleier (ref logg).** Men om det bare er pasienten vi tenker på så: 4

Generelt inntrykk av produktet: I pleietjenesten på sengepost opplever vi tidvis en meget hektisk aktivitet, hvor det å prioritere oppgavene riktig kan være avgjørende for en effektiv organisering dagen. Et differensiert ringesignal vil hjelpe pleiepersonell i denne prioriteringen. Produktet som her er utviklet adresserer langt på vei de utfordringer man tidlig identifiserte i samarbeidet. Produktet gir pasientene mulighetene til å kommunisere målrettet med pleiepersonalet gjennom ulike valg for hvorfor man ønsker assistanse. Differensieringen av ringesignalet vil også være et godt verktøy med tanke på oppgavedeling blant pleiepersonalet. Riktig kompetanse på rett sted er en viktig faktor for kvalitet i pleie og behandling i helsetjenesten i dag. Produktet har også identifisert behovet for en samhandling med det lokale kjøkkenet, noe som ikke finnes i dagens løsninger. Dette vil ytterligere forbedre oppgavedelingen og kan ha potensial til å redusere kostnader i egen avdeling. Generelt sett mener vi at produktet i stor grad imøtekommmer de behov vi i utgangspunktet adresserte. Dersom det ikke finnes andre liknende kommersielle løsninger som vi kan ta i bruk, mener vi produktet har potensial som en innovasjonsløsning, og man bør vurdere å kontakte sentrale aktører/samarbeidspartnere for å se på muligheten for videre utvikling.

---

## C Fusion360 Plantegninger

I denne seksjonen presenteres alle plantegninger på alle modeller som ble 3D printet.

Planene ligger også i GitHub, som er linka til i implementering.

