

# Brukbarhet ved benyttelse av fri programvare i systemutvikling - en praktisk studie

**Tarjei Eriksen Ormestøyl**  
**Anders Kløvrud Rognstad**

Master i datateknikk  
Oppgaven levert: Juni 2010  
Hovedveileder: Dag Svanæs, IDI



# Oppgavetekst

MOBESITY er et forskningsprosjekt ved NTNU i samarbeid med overvektsklinikken ved St. Olavs Hospital i Trondheim. Målet med prosjektet er å utvikle en mobil/Internett-applikasjon for egenomsorg blant overvektspasienter. Som del av oppgaven skal det utvikles en applikasjon basert på åpen kildekode (open source). Denne skal så brukbarhetstestes med pasientgruppen. Problemstillingen i oppgaven er knyttet til hvordan man ivaretar brukervennlighet i prosjekter som baserer seg på åpen kildekode.

Oppgaven gitt: 15. januar 2010  
Hovedveileder: Dag Svanæs, IDI



# Sammendrag

Bruk av åpen kildekode fører med seg mange fordeler og blir mer og mer utbredt i programvareutvikling. Utviklingstiden til prosjekter som tar i bruk ferdigkomponenter blir relativt kort sammenlignet med systemer som utvikles fra grunnen av, og man vil raskt ha et system oppe å kjøre. En ser dog at brukbarhet ofte ikke har høyeste prioritet i åpen kildekode-prosjekter. I mange tilfeller er den tiltenkte brukergruppen ekspertbrukere eller utviklerne selv og prosjektenes natur fører til at tradisjonelle brukbarhetsmetoder blir vanskelige å gjennomføre.

I dette prosjektet har vi sett på hvilke brukbarhetsutfordringer som gjør seg gjeldende ved bruk av åpen kildekode-komponenter i systemutvikling, og hvordan slike utfordringer kan reduseres. Vi har utviklet et nettbasert egenomsorgssystem for overvektige og brukt dette som et systemutviklingscase. I forbindelse med denne casen ble det gjennomført to brukbarhetstester på systemet. Ut ifra resultatene fra disse testene, samt våre erfaringer fra utviklingsprosessen, kan vi si oss enige i at utbyttet er stort ved bruk av fri programvare kontra tradisjonell systemutvikling. Blant problemene vi fant var at komponentene vi benyttet oss av hadde svakheter og brudd innenfor brukbarhetsheuristikker som *Konsistens og standarder* og *Eстетisk og minimalistisk design*. Vi ser også at de fleste brukbarhetsproblemene var knyttet til enkeltkomponenter, og færre til selve integrasjonen av komponentene.

Likevel fant vi at tilfredsheten og anvendbarheten til systemet var tilstrekkelig, selv om utseendet på det grafiske brukergrensesnittet kunne holdt en høyere standard. Relevant litteratur underbygger våre erfaringer om at mye kan gjøres underveis i utviklingsprosessen for å redusere brukbarhetsproblemer. Spesielt bør en tidlig vurdering og integrering av aktuelle komponenter gjennomføres for å avdekke risikomomenter og legge til rette for brukbarhetsevaluering på et tidlig stadium.



# Forord

Denne masteroppgaven er gjennomført i siste semesteret av sivilingeniørstudiet innen studieretningen Program og Informasjonssystemer ved Institutt for Datateknikk og Informasjonsvitenskap (IDI) ved NTNU. Oppgaven har sett på hvilke brukbarhetsutfordringer som gjør seg gjeldende ved utvikling basert på fri programvare, og hvordan disse kan reduseres.

Oppgaven er et ledd i MOBESITY-prosjektet, som er et forskningsprosjektet i samarbeid med NSEP og Obesitasklinikken ved St.Olav, ledet av doktogradsstipendiat Anita Das. Vi vil takke Anita for et godt samarbeid, og vi ønsker henne lykke til videre med prosjektet. Vi vil takke teknisk ansvarlig ved NSEP, Terje Røsand, og alle andre ved NSEP som har hjulpet oss å gjennomføre brukbarhetstestene. En stor takk rettes også til alle som stilte opp som testpersoner ved disse anledningene.

Mest av alt vil vi takke vår veileder Dag Svanæs for god rettleiding og verdifull hjelp i arbeidet med denne oppgaven. Det har vært et artig og lærerikt siste år.

Sist, men ikke minst, vil vi takke vår gode venn og samarbeidspartner Ørjan D. Johansen for et godt arbeidsklima og for mange gode stunder gjennom denne våren på «læbben».

*Trondheim, 10. juni 2010*

---

Tarjei E. Ormestøyl      Anders K. Rognstad





# Innhold

<b>1</b>	<b>Introduksjon</b>	<b>1</b>
1.1	Motivasjon . . . . .	1
1.2	Kontekst . . . . .	2
1.3	Problemstilling . . . . .	3
1.3.1	Forskningsspørsmål . . . . .	3
1.3.2	Omfang . . . . .	3
1.4	Utforming . . . . .	4
<b>2</b>	<b>Bakgrunn</b>	<b>7</b>
2.1	Utgangspunkt . . . . .	7
2.1.1	MOBESITY-prosjektet . . . . .	7
2.1.2	Original kravspesifikasjon . . . . .	8
2.1.3	Workshop med prototype . . . . .	9
2.1.4	Brukskontekst . . . . .	10
2.2	Brukbarhet . . . . .	11
2.2.1	Hva er brukbarhet? . . . . .	11
2.2.2	Nielsens ti punkter . . . . .	12
2.2.3	Shneidermans «8 Golden Rules» . . . . .	13
2.3	Fri programvare . . . . .	14
2.3.1	Hva er fri programvare? . . . . .	15
2.3.2	Økonomiske motiver . . . . .	16
2.3.3	Historikk . . . . .	17
2.4	Brukbarhet i fri programvare . . . . .	17
2.4.1	Hvorfor fri programvare? . . . . .	17
2.4.2	Utfordringer . . . . .	18
2.4.3	Brukbarhetsstudier av fri programvare . . . . .	21
2.4.4	Muligheter og trender . . . . .	21
2.5	Integrasjon av ferdigkomponenter . . . . .	23
<b>3</b>	<b>Forskningsdesign</b>	<b>27</b>

3.1	Generelt om metoder . . . . .	27
3.1.1	Aksjonsforskning og eksperiment . . . . .	27
3.1.2	Utvikling . . . . .	28
3.1.3	Intervju . . . . .	29
3.1.4	Spørreskjema . . . . .	29
3.2	Brukbarhetsmetoder . . . . .	30
3.2.1	Heuristisk evaluering . . . . .	30
3.2.2	Brukbarhetstesting . . . . .	31
3.2.3	Verktøy ved brukbarhetstesting . . . . .	34
3.2.4	Analyse av brukbarhetstestresultater . . . . .	36
3.3	Valg av metode . . . . .	37
3.3.1	Systemutviklingscase . . . . .	37
3.3.2	Brukbarhetstesting . . . . .	38
3.3.3	Analyse av resultater . . . . .	38
<b>4</b>	<b>Utvikling</b>	<b>41</b>
4.1	Tidsforløp . . . . .	41
4.2	Oppstart . . . . .	43
4.3	Implementasjon . . . . .	44
4.4	Videreutvikling . . . . .	47
4.4.1	Visuelt tema . . . . .	47
4.4.2	Hjem-siden . . . . .	48
4.4.3	Informasjon . . . . .	49
4.4.4	Forum . . . . .	51
4.4.5	Kalender . . . . .	52
4.4.6	Elektronisk dagbok . . . . .	53
4.5	Avslutning og evaluering . . . . .	54
4.5.1	Modifiserbarhet med tanke på brukbarhet . . . . .	56
4.5.2	Videre arbeid . . . . .	57
<b>5</b>	<b>Endelig løsning</b>	<b>59</b>
5.1	Introduksjon . . . . .	59
5.2	Publiseringsverktøy: Wordpress 2.9.2 . . . . .	60
5.2.1	Innstikk og temaer . . . . .	61
5.3	Kommunikasjonsverktøy: Simple:Press 4.2.2 . . . . .	61
5.4	Kalenderverktøy: WebCalendar 1.2.0 . . . . .	62
5.5	Universal innlogging . . . . .	62
5.5.1	SMS-utsending . . . . .	63
5.6	Grafisk grensesnitt . . . . .	63
5.6.1	Logg inn-skjermene . . . . .	64
5.6.2	Hjem-siden . . . . .	64

5.6.3	Informasjon . . . . .	65
5.6.4	Forumet . . . . .	65
5.6.5	Kalenderen . . . . .	66
<b>6</b>	<b>Brukbarhetstest</b>	<b>69</b>
6.1	Formål med testene . . . . .	69
6.2	Forberedelser . . . . .	70
6.2.1	Brukerne . . . . .	70
6.2.2	Oppgavene . . . . .	71
6.2.3	Lokasjon og oppsett . . . . .	71
6.2.4	Pilottest . . . . .	72
6.3	Gjennomføring . . . . .	72
6.3.1	Innsamling av data . . . . .	72
6.3.2	Problemer og utfordringer . . . . .	73
<b>7</b>	<b>Resultater I</b>	<b>75</b>
7.1	Testpersoner . . . . .	75
7.2	Observasjoner og problemer . . . . .	76
7.2.1	Gradering av problemer . . . . .	76
7.2.2	Kategorisering av problemer . . . . .	76
7.2.3	Brukbarhetsproblemer: Brukbarhetstest I . . . . .	77
7.3	Anvendbarhet . . . . .	79
7.4	Tilfredshet . . . . .	81
<b>8</b>	<b>Resultater II</b>	<b>83</b>
8.1	Testpersoner . . . . .	83
8.2	Observasjoner og problemer . . . . .	84
8.2.1	Gradering av problemer . . . . .	84
8.2.2	Kategorisering av problemer . . . . .	84
8.2.3	Brukbarhetsproblemer: Brukbarhetstest II . . . . .	84
8.3	Anvendbarhet . . . . .	85
8.4	Tilfredshet . . . . .	89
<b>9</b>	<b>Analyse</b>	<b>91</b>
9.1	Analyse og kategorisering i heuristikker . . . . .	91
9.1.1	Konsistens og standarder . . . . .	91
9.1.2	Overensstemmelse med den virkelige verden . . . . .	92
9.1.3	Brukerkontroll og frihet . . . . .	95
9.1.4	Gjenkjenning før erindring . . . . .	95
9.1.5	Synliggjør statusen til systemet . . . . .	97
9.1.6	Estetisk og minimalistisk . . . . .	97

9.1.7	Fleksibilitet og anvendbarhet . . . . .	100
9.1.8	Systemfeil . . . . .	100
9.1.9	Metodefeil . . . . .	101
9.2	Relasjon til fri programvare . . . . .	101
9.2.1	Feil som ikke er knyttet til åpen kildekode-komponenter	102
9.2.2	Feil som er knyttet til hver enkelt åpen kildekode-- komponent . . . . .	103
9.2.3	Feil som er knyttet til integrasjonen av de ulike åpen kildekode-komponentene . . . . .	104
9.3	Tilfredsheten blant testpersonene . . . . .	105
<b>10</b>	<b>Diskusjon</b>	<b>107</b>
10.1	Problemstilling . . . . .	107
10.1.1	Forskningsspørsmål 1 . . . . .	107
10.1.2	Forskningsspørsmål 2 . . . . .	113
10.2	Resultatkvalitet . . . . .	118
10.2.1	Validitet . . . . .	118
10.2.2	Generaliserbarhet . . . . .	120
10.2.3	Reliabilitet . . . . .	121
10.2.4	Falske problemer . . . . .	122
10.3	Metodediskusjon . . . . .	123
10.3.1	Systemutviklingscase . . . . .	123
10.3.2	Brukbarhetstesting . . . . .	124
10.3.3	Intervju . . . . .	125
10.3.4	Spørreundersøkelse . . . . .	125
<b>11</b>	<b>Konklusjon</b>	<b>127</b>
11.1	Problemstilling . . . . .	127
11.1.1	Forskningsspørsmål 1 . . . . .	128
11.1.2	Forskningsspørsmål 2 . . . . .	129
11.2	Videre arbeid . . . . .	130
	<b>Bibliografi</b>	<b>136</b>
	<b>Tillegg</b>	<b>136</b>
	<b>A SUS-spørreskjema</b>	<b>137</b>
	<b>B Oppgaver: Brukertest I</b>	<b>139</b>
	<b>C Oppgaver: Brukertest II</b>	<b>141</b>

<b>D</b>	<b>Bakgrunnsskjema</b>	<b>143</b>
<b>E</b>	<b>Bakgrunnsskjema: Brukertest I</b>	<b>149</b>
<b>F</b>	<b>Bakgrunnsskjema: Brukertest II</b>	<b>153</b>
<b>G</b>	<b>Kildekode-veiledning</b>	<b>155</b>



# Tabeller

4.1	Kodekvaliteten til komponentene . . . . .	56
7.1	Brukbarhetsproblemer A1-A17 . . . . .	78
7.2	Brukbarhetsproblemer A18-A23 . . . . .	79
7.3	Sammenbrudd og fullførte oppgaver i brukbarhetstest I . . . . .	80
7.4	SUS-resultater brukertest I . . . . .	81
8.1	Brukbarhetsproblemer B1-B12 . . . . .	86
8.2	Brukbarhetsproblemer B13-B23 . . . . .	87
8.3	Brukbarhetsproblemer B24-B36 . . . . .	88
8.4	Brukbarhetsproblemer B37-B39 . . . . .	89
8.5	Sammenbrudd og fullførte oppgaver i brukbarhetstest II . . . . .	89
8.6	SUS-resultater brukertest II . . . . .	90
9.1	Problemer knyttet til eget design . . . . .	103
9.2	Andre problemer . . . . .	103
9.3	Problemer knyttet til egne tilpasninger av komponenter . . . . .	104
9.4	Problemer knyttet til standard komponentfunksjonalitet . . . . .	104
9.5	Problemer knyttet til integrasjon av komponenter . . . . .	105
10.1	Antall åpen kildekode-problemer i hver kategori . . . . .	108





# Figurer

2.1	Motivasjon for utviklere til å bidra i fri programvare . . . . .	19
2.2	Brukere av et fri programvare-prosjekt . . . . .	23
3.1	Forholdet mellom brukbarhetsproblemer avdekket og antall testpersoner . . . . .	33
3.2	Eksempel på resultat fra eye-tracking . . . . .	35
4.1	De viktigste fasene i prosjektløpet . . . . .	42
4.2	Utseendevalg i administrasjonspanelet, Simple:Press . . . . .	45
4.3	Fargevalg i administrasjonspanelet, WebCalendar . . . . .	46
4.4	Hjem-siden i versjon 1 . . . . .	48
4.5	Hjem-siden i versjon 2 . . . . .	49
4.6	Informasjons-siden i versjon 1 . . . . .	50
4.7	Informasjonssidene i versjon 2 . . . . .	51
4.8	Forumet i versjon 1 . . . . .	52
4.9	Forumet i versjon 2 . . . . .	53
4.10	Kalenderen i versjon 1 . . . . .	54
4.11	Kalenderen i versjon 2 . . . . .	55
5.1	Logg inn steg 1 . . . . .	64
5.2	Logg inn steg 2 . . . . .	64
5.3	Hjem-siden til MOBESITY-nettsiden . . . . .	65
5.4	Forumet til MOBESITY-nettsiden . . . . .	66
5.5	Kalender-siden til MOBESITY-nettsiden . . . . .	67
5.6	Legg til aktivitet-siden til MOBESITY-nettsiden . . . . .	68
7.1	Fordeling av alvorlighetsgrad brukbarhetstest I . . . . .	80
8.1	Fordeling av alvorlighetsgrad brukbarhetstest II . . . . .	85
9.1	Emneknappen kan ikke trykkes på . . . . .	92
9.2	Søkeresultater av å søke på Spirometri . . . . .	93
9.3	Sosiale spørsmål-siden . . . . .	94

9.4	Lagre nytt emne-dialog . . . . .	94
9.5	Legg til aktivitet med påminnelse . . . . .	96
9.6	Kalender med årsvisnings-valg . . . . .	97
9.7	Notifikasjonen som viser at et emne er postet. . . . .	98
9.8	Innboks-visningen i forumet . . . . .	99
9.9	SUS-skalaen . . . . .	105
10.1	Avdekkede brukbarhetsproblemer . . . . .	110
A.1	SUS-spørreskjema . . . . .	138

# Kapittel 1

## Introduksjon

Dette kapittelet starter med å forklare motivasjonen bak dette prosjektet og oppgavens kontekst. Videre vil vi definere den overordnede problemstillingen og forskningsspørsmålene som vi ønsker å besvare, samt oppgavens omfang. Til slutt kommer en oversikt over kapitlene i denne rapporten.

### 1.1 Motivasjon

Fri programvare blir stadig mer utbredt og populært blant vanligere brukere, men også blant kommersielle IT-aktører. Spesielt i Norge fokuserer blant annet flere av de kjente konsulentselskapene på fri programvare, slik som Bekk og Accenture [1, 2]. De ser en mulighet for å bruke eksisterende frie løsninger i sine prosjekter, og de ser også en mulighet for å forbedre sine produkter ved å gjøre de om til åpen kildekode-prosjekter. For å tjene penger tilbyr de for eksempel utvidet funksjonalitet, support og dokumentasjon.

Noen fri programvare-prosjekter har blitt ledende aktører på sitt område. Eksempler på dette er Mozilla-prosjektet som blant annet har utviklet en nettleser og en e-postklient, OpenSSL som er et verktøy som implementerer SSL og TLS-protokollene, og ikke minst operativsystemet Linux. Med fri programvare åpner det seg nye muligheter, men det fører også med seg en del utfordringer.

I mange åpen kildekode-prosjekter har det lenge vært vanlig å undervurdere viktigheten av brukbarhet av ulike grunner. Ofte er de fleste brukerne spesielt interesserte, og meget datakyndige. Behovet for et godt brukerdesign blir mindre viktig, da brukerne tar til takke med det meste så lenge de får

den funksjonaliteten de ønsker. Dessuten er det vanskeligere å planlegge og gjennomføre brukbarhetsutvikling og brukbarhetstesting da utviklerne ofte sitter spredt og hovedsaklig bruker tekstlig kommunikasjon [3].

Trender i tiden viser at ting er i ferd med å endres. En ser at store, verdensomspennende, fri programvare-prosjekter begynner å fokusere på brukbarhet. For eksempel har stiftelsen bak Wikipedia satt i gang et «Usability Initiative» for å forbedre brukbarheten til Wikipedia for bidragsyterne [4]. GNU, en av de største aktørene innenfor åpen kildekode som blant annet står bak GNU/Linux-operativsystemet, er et annet eksempel på dette. De har nylig ansatt meget svaksynte Chris Hofstader, en systemutvikler med 30 års erfaring. Han skal arbeide med å gjøre produktene til GNU mer tilgjengelig for svakerestilte brukergrupper, som for eksempel blinde og døve [5, 6].

## 1.2 Kontekst

Ved å bruke ulike fri programvare-komponenter er det mulig å dekke mye funksjonalitet og samtidig bruke færre ressurser enn ved tradisjonell systemutvikling. Totalprisen til et prosjekt faller på denne måten drastisk. Benyttelse av fri programvare har mange fordeler, men det følger også med en del nye utfordringer. Vi fikk en fin mulighet til å studere dette i praksis gjennom det eksisterende MOBESITY-forskningsprosjektet.

MOBESITY er et forskningsprosjekt ved NTNU i samarbeid med overvektsklinikken ved St. Olavs Hospital i Trondheim. Prosjektet ble startet av Anita Das som hennes PhD-oppgave våren 2009 og skal gjennomføres hos Norsk Senter for Elektronisk Pasientjournal (NSEP) som er et tverrfaglig forskningsmiljø ved NTNU.

Målet med prosjektet var opprinnelig å utvikle en mobil/Internett-applikasjon for egenomsorg blant overvektspasienter. Pasientene skal kunne overvåke egen helse ved for eksempel å observere/registrere inntak av mat og fysisk aktivitet. Helsepersonell skal kunne studere observasjonene pasientene gjør og dermed kunne gi råd og veiledning basert på dette. Det vil også være mulighet for å studere atferdsmønstre blant pasienter basert på denne dataen. Prosjektet har endret noe kurs underveis etter workshopper og diskusjon med pasienter og helsepersonell.

Vår rolle i dette prosjektet er å utvikle en nettside for pasienter og helsepersonell ved bruk av fri programvare, for å se nærmere på hvilke utfordringer knyttet til brukbarhet som dukker opp. Parallelt med vårt prosjekt vil Ørjan

Johansen fokusere på sikkerhet ved bruk av fri programvare i sin hovedoppgave, ved å benytte seg av samme systemutviklingscase. Utviklingsprosessen vil være lik for disse to prosjektene, men fokuset underveis vil være spesifikke for de to prosjektene.

## 1.3 Problemstilling

Hovedmålet med dette prosjektet er å se på *hvilke utfordringer knyttet til brukbarhet som kan oppstå ved bruk av fri programvare i utviklingsprosjekter, og hvordan disse kan reduseres.*

For å besvare denne problemstillingen vil vi gjennomføre et praktisk studie der vi benytter oss av fri programvare for å realisere en rekke krav til et system.

### 1.3.1 Forskningsspørsmål

Med bakgrunn i vår hovedproblemstilling har vi formulert følgende forskningsspørsmål:

**FS1** Hvilke brukbarhetsutfordringer fører utvikling basert på fri programvare med seg?

**FS2** Hvordan kan en redusere problemene knyttet til brukbarhet ved utvikling basert på fri programvare?

### 1.3.2 Omfang

Hovedfokuset til denne oppgaven er å se nærmere på brukbarhet ved å benytte fri programvare i utviklingsprosjekter. Vi vil se nærmere på hvilke type problemer som kan oppstå, og hvordan slike problemer kan reduseres.

For å kunne svare optimalt på problemstillingen vil vi gjennomføre et systemutviklingscase, der vi utvikler et system basert på åpen kildekode-komponenter. Det vi utvikler vil være et reelt system med reelle krav og reelle brukere. Ved å få denne praktiske vinklingen vil vi kunne avdekke flere problemer, og også få en dypere innsikt i problemenes natur.

Brukbarhetsproblemene vil vi avdekke ved å gjennomføre brukbarhetstesting av systemet. Vi vil fokusere på å få testet på brukere som tilhører pasientgruppen, da det senere i MOBESITY-prosjektet skal testes av en større pasientgruppe i en realistisk fullskala studie. I forbindelse med brukbarhetstesting vil vi også benytte oss av intervjuer og spørreskjema for å få et bedre datagrunnlag. Resultater fra brukbarhetstesting vil gi grunnlag for å svare på forskningsspørsmål 1.

Vi vil hovedsaklig benytte erfaring opparbeidet i dette prosjektet for å svare på forskningsspørsmål 2. Basert på denne erfaringen vil vi komme med våre anbefalinger til hvordan en bør tilnærme seg et utviklingsprosjekt der fokuset er på bruk av fri programvare, og hvordan en kan oppnå et tilfredsstillende nivå av brukbarhet.

Denne oppgaven vil fokusere på området brukbarhet, og ikke ta for seg de mange andre aspektene ved utvikling basert på åpen kildekode-komponenter. Andre kvalitetskrav til et system enn brukbarhet, og aspekter ved disse i sammenheng med fri programvare vil ikke bli diskutert. Derrest vil de strenge kravene til sikkerhet og utfordringer knyttet til dette være utenfor dette prosjektets omfang.

## 1.4 Utforming

Da dette prosjektet har vært et felles utviklingsprosjekt og en del av forskningsprosjektet MOBESITY, vil en liten del av rapporten være felles for undertegnede og Ørjan Johansen. Dette gjelder hovedsakelig deler av Kapittel 5, som presenterer den endelige løsningen.

**Kapittel 1: Introduksjon** Dette kapitlet presenterer motivasjonen for prosjektet og prosjektets kontekst. Prosjektets problemstilling og forskningsspørsmål blir presentert og utdypet.

**Kapittel 2: Bakgrunn** Dette kapitlet gir et dypere innblikk i prosjektets utgangspunkt og kontekst, og presenterer bakgrunns litteratur som er relevant for å besvare problemstillingen.

**Kapittel 3: Forskningsdesign** Dette kapitlet beskriver forskningsmetodene vi har benyttet oss av for å besvare problemstillingen, og begrunner våre valg av metoder.

**Kapittel 4: Utvikling** Dette kapitlet beskriver utviklingsprosessen, med

vekt på hva som ble gjort for å redusere brukbarhetsproblemer underveis.

**Kapittel 5: Endelig løsning** Dette kapitlet presenterer den endelige løsningen som ble utviklet i dette prosjektet.

**Kapittel 6: Brukbarhetstest** Dette kapitlet tar for seg formålet med brukbarhetstestene, samt hvordan den praktiske gjennomføringen fungerte.

**Kapittel 7: Resultater I** Dette kapitlet presenterer resultatene fra den første brukbarhetstesten.

**Kapittel 8: Resultater II** Dette kapitlet presenterer resultatene fra den andre brukbarhetstesten.

**Kapittel 9: Analyse** I dette kapitlet blir resultatene fra de to brukbarhetstestene analysert, og vi ser nærmere på hvilke type problemer som har oppstått, hvor disse oppstår og årsaker til problemene.

**Kapittel 10: Diskusjon** I dette kapitlet diskuterer vi funnene våre i lys av problemstillingen og forskningsspørsmålene. I tillegg vurderer vi kvaliteten til resultatene, og vi ser nærmere på nytteverdien og gjennomføringen av forskningsmetodene brukt.

**Kapittel 11: Konklusjon** Dette kapitlet oppsummerer svarene på problemstillingen og forskningsspørsmålene, og ser på videre arbeid med temaet.





# Kapittel 2

## Bakgrunn

Dette kapitlet presenterer bakgrunnen og utgangspunktet for dette prosjektet. Vi vil også presentere teori om brukbarhet, fri programvare og integrasjon av ferdige programvarekomponenter som er relevant for å svare på problemstillingen.

### 2.1 Utgangspunkt

Dette prosjektet er en videreføring av høstens fordypningsprosjekt. Gjennom høstprosjektet utarbeidet vi krav og ønsker til en løsning for pasienter ved Obesitasklinikken ved St. Olav. Denne seksjonen vil først presentere forskningsprosjektet MOBESITY, som vi har vært en del av. Deretter vil vi presentere de relevante resultatene fra høstens prosjekt, som fungerte som grunnlag for utviklingsprosessen. Til slutt vil vi definere brukskonteksten for systemet som skal utvikles.

#### 2.1.1 MOBESITY-prosjektet

MOBESITY er et forskningsprosjekt ved NTNU i samarbeid med overvektsklinikken ved St. Olavs Hospital i Trondheim. Prosjektet ble startet av Anita Das som hennes PhD-oppgave våren 2009, og gjennomføres hos Norsk senter for elektronisk pasientjournal (NSEP) som er et tverrfaglig forskningsmiljø ved NTNU.

Målet med prosjektet er å utvikle en mobil/Internett-applikasjon for egenomsorg blant overvektspasienter. Pasientene skal kunne overvåke egen helse ved for eksempel å observere/registrere inntak av mat og fysisk aktivitet. Helsepersonell skal kunne studere observasjonene pasientene gjør og dermed kunne gi råd og veiledning basert på dette. Det vil også være mulighet for å studere atferdsmønstre blant pasienter basert på denne dataen.

Prosjektet skal forsøke å besvare fire forskningsspørsmål [7]:

- Hva trenger overvektspasienter for å opprettholde en sunn livsstil etter en overvektoperasjon eller et livsstilskurs? Hvordan kan egenomsorg blant overvektspasienter bli forbedret ved hjelp av elektroniske hjelpemidler (PCer, mobiltelefoner)?
- Hvilke krav har helsepersonell til en mobilapplikasjon for egenomsorg blant overvektspasienter?
- Hvordan skiller pasientene sine krav til en E-helse-applikasjon seg fra helsepersonellens krav? Hvilke design-avgjørelser reflekterer disse forskjellene?
- Hva er resultatene av å bruke en mobilapplikasjon for egenomsorg blant overvektspasienter, sett fra et brukerperspektiv?

### 2.1.2 Original kravspesifikasjon

Kravspesifikasjonen [7] (versjon 1.0, sist revidert 29.09.09) er utarbeidet etter informasjon fra pasienter som har gjennomgått overvektsbehandling og fra helsepersonell som arbeider med denne pasientgruppen. Åtte personer med helsefaglig utdanning (medisin, sykepleie, ernæring) og tolv personer som har gjennomgått overvektsbehandling deltok i tre separate workshoper våren 2009. Seks personer hadde gjennomgått livsstilsbehandling og seks personer hadde gjennomgått kirurgisk behandling.

MOBESITY-kravspesifikasjonen ble utarbeidet i samarbeid med pasienter og helsepersonell med et felles mål [7]. Begge gruppene ønsket et system som kan støtte personer som har gjennomgått eller gjennomgår enten kirurgisk behandling eller livsstilskurs. Kravene er et produkt av flere workshoper der både pasienter og helsepersonell har deltatt. Begge parter hadde ønsker til et slikt system, og de ønskene som ble høyest prioritert ble tatt med i kravspesifikasjonen.

Det var ønske om ulike brukerroller i systemet; *Helsepersonell, Pasient før*

## 2.1. UTGANGSPUNKT

---

*behandling, Pasient som gjennomgår kirurgisk behandling og Pasient som gjennomgår livsstilskurs.* Felles for alle rollene er at systemet må være sikkert, og følge de lover og forskrifter som er påkrevd for et system som behandler pasientinformasjon.

Oppsummert bestod den funksjonelle kravspesifikasjonen av følgende hovedpunkter:

- Krav om tilgang til informasjonsmateriell og ofte stilte spørsmål og svar.
- Krav om kommunikasjon mellom pasienter og helsepersonell.
- Krav om fadderordning for å gi bedre støtte og oppfølging for nye pasienter.
- Krav om en egen plan der pasientene kan holde oversikt over egne aktiviteter og oppgaver.
- Krav om informasjon om kosthold og måltidsrytme, og mulighet for å få påminnelser om for eksempel å spise.
- Krav om kvalitetssikring av informasjon (av helsepersonell).

### 2.1.3 Workshop med prototype

Basert på kravene i MOBESITY-kravspesifikasjonen ble det sett på ulike måter å realisere kravene på. Vi holdt derfor en workshop med pasienter og helsepersonell 3. November 2009. Som input til workshopen ble det utarbeidet en tidlig-fase prototype for å kvalitetssikre kravene og få i gang en diskusjon med brukergruppen hvor ønskene til funksjonalitet kunne bli tilstrekkelig kartlagt.

Under workshopen ble deltagerne delt opp i forskjellige grupper og bedt om å prioritere krav til systemet. De ulike gruppene hadde noe ulike prioriteringer, men det var noen punkter som gikk igjen som spesielt viktige. Nettbasert kommunikasjon mellom pasienter og helsepersonell ble sett på som essensielt for et ønsket system. Dette var ønsket av begge brukergrupper; Helsepersonellet følte de ville spare tid på å svare elektronisk, samt at pasient til pasientkommunikasjon (gjennom for eksempel forum) ville ta av litt av lasten på helsepersonellet. Pasientene så også muligheten til å oppnå flere effekter. De var interessert i et sosialt nettverk der de kunne søke svar på spørsmål og diskutere utfordringer og muligheter knyttet til behandlingsopplegget.

*Min plan* ble også sett på som en viktig del av løsningen. Det var mange som hadde behov for å bli minnet på å spise måltider, trene, ta vitaminer og medisiner og lignende. Vi så en mulighet for at dette kunne realiseres på lik måte som Google har gjort det i Google Calendar som vi brukte som eksempel på workshopen. *Min Plan* kunne erstatte eksisterende individuelle rutiner å gjøre dette på, slik som mobilpåminnelser, egne kalendere og alarmklokker. I tillegg ville en forhåpentligvis få en bedre helhet ved å inkludere denne funksjonaliteten i systemet.

Høstens funn gjorde at vi hadde et godt bilde av hva som var ønsket av systemet, og tett samarbeid med prosjektlederen for MOBESITY-prosjektet gjorde at vi fikk kontinuerlig tilbakemelding. Gjennom dialog og workshop fikk vi også et godt bilde av den framtidige brukergruppen, og deres krav og ønsker til systemet. Vi kunne dermed sette i gang utvikling uten å konkretisere kravene ytterligere.

#### 2.1.4 Brukskontekst

Vi vil her ta utgangspunkt i ISO-definisjon av brukbarhet [8] for å definere brukskonteksten til systemet som skal utvikles. Dette inkluderer brukerne, oppgavene, utstyret og de fysiske og sosiale omgivelsene.

**Brukere** Brukerne av dette systemet er pasienter og helsepersonell tilknyttet overvektsklinikken ved St. Olavs Hospital i Trondheim. Pasientene vil ha gjennomgått kirurgisk behandling og/eller livsstilsbehandling for overvekt. Kunnskapene om domenet anses å være gode for alle brukerne. Tekniske ferdigheter, erfaring i bruk av lignende systemer vil variere over brukergruppen. Det samme gjelder kjønn, alder og utdanning, men samtlige brukere er i voksenalder.

**Oppgaver** De typiske oppgavene vil være å søke etter informasjon, kommunisere med andre pasienter og helsepersonell, benytte seg av en kalender for å sette opp aktiviteter og loggføre en personlig plan om trening, kosthold og annet. Oppgavene kan gjøres daglig, men det er i stor grad opp til hver enkelt hvor ofte de vil bruke systemet. Oppgavens mål er å støtte pasientene under og i etterkant av overvektbehandlingen. Ingen av oppgavene som skal utføres er tidskritiske.

**Utstyr** Systemet vil være nettbasert og vil normalt brukes på brukernes personlige datamaskin. Utover en PC, mobiltelefon og tilkobling til Internett er det ingen spesielle krav til utstyr for å bruke systemet.

**Omgivelser** Systemet vil hovedsakelig brukes av den enkelte bruker hjemme hos seg selv, og ellers der det er behov.

## 2.2 Brukbarhet

I denne seksjonen vil vi presentere relevant teori rundt begrepet brukbarhet og ulike heuristikker for å evaluere brukbarhet.

### 2.2.1 Hva er brukbarhet?

Kort sagt kan brukbarhet sies å være et kvalitativt mål på hvor enkelt et grensesnitt er å bruke. Det finnes flere gode og mer presise definisjoner av brukbarhet som vi vil ta for oss her.

ISO 9241-11 [8] har standardisert begrepet brukbarhet og definerer det som i hvilken grad et produkt kan brukes av en spesifikk bruker til å oppnå spesifikke mål innenfor anvendbarhet, effektivitet og tilfredshet i en spesifikk brukskontekst.

Her omhandler anvendbarhet i hvilken grad systemet er nyttig og om brukeren oppnår sine mål ved å bruke systemet. Effektivitet dreier seg først og fremst om hvor mye tid en bruker for å oppnå disse målene, mens tilfredshet sier noe om brukerens oppfatninger og holdninger til systemet. Brukskonteksten består av følgende punkter:

- Bruker(e).
- Oppgaver.
- Utstyr (Maskinvare, programvare, materialer).
- Fysiske og sosiale omgivelser.

Brukbarhets-eksperten Jacob Nielsen definerer brukbarhet ut i fra fem viktige kvalitetsattributter [9] som måler:

1. **Lærbarhet:** Hvor lett det er for brukere å utføre enkle oppgaver ved første gangs bruk av et system.
2. **Effektivitet:** Når brukerne kjenner grensesnittet, hvor raskt kan de utføre oppgaver i systemet?

3. **Huskbarhet:** Hvor enkelt kan brukerne gjenopprette sine ferdigheter i systemet etter en periode uten bruk?
4. **Feil:** Hvor mange feil gjør brukerne under bruk, hvor alvorlige er feilene og hvor lett kan de hente seg inn igjen?
5. **Tilfredshet:** Hvor tilfredse er brukerne under bruk av systemet?

Et godt designet grensesnitt er alfa omega for brukbarheten til et system. Av de mest kjente heuristikkene for å måle brukbarhet er Jakob Nielsens ti punkter og Ben Shneidermans åtte gylne regler. Disse heuristikkene har et relativt likt budskap, og underbygger hverandre svært godt. Vi ser det derfor som nyttig å presentere disse som en introduksjon til fagfeltet, og de retningslinjene en trenger å forholde seg til for å oppnå god brukbarhet på produkter som skal brukes av en sluttbruker.

### 2.2.2 Nielsens ti punkter

Jakob Nielsens ti heuristikker sees av mange på som selve grunnsteinene innenfor fagfeltet brukbarhet- og interaksjonsdesign [10]. Punktene kan benyttes som en huskeliste ved utvikling av interaksjonssystemer, eller som en heuristisk brukbarhetsevaluering.

1. **Synliggjør statusen til systemet**  
Hold brukeren informert gjennom fornuftige tilbakemeldinger til passende tid.
2. **Overensstemmelse med den virkelige verden**  
Systemet bør snakke brukerens språk med ord, fraser og konsepter som er kjent for brukeren og presentere innholdet på en naturlig og logisk måte.
3. **Brukerkontroll og frihet**  
Brukeren trenger mulighet til å avbryte og forlate det han<sup>1</sup> holder på med uten å gå gjennom en unødvendig lang dialog, samt tilgang til angre- og gjør-om-funksjoner.
4. **Konsistens og standarder**  
Følg kjente designkonvensjoner slik at ikke brukeren trenger å lure på om ulike ord, situasjoner og handlinger betyr det samme.

---

<sup>1</sup>For enkelhets skyld skal «han» leses som «han eller hun» gjennom denne rapporten

### 5. Feilforebygging

Enda bedre enn gode feilmeldinger er et system som hindrer feil fra å opptre (for brukeren). Eliminer feilutsatte tilstander eller presenter brukeren med en bekreftelsesdialog før brukeren kan utføre en slik handling.

### 6. Gjenkjenning før erindring

Minimer brukerens last på minne ved å gjøre objekter, handlinger og valg synlige. Brukeren bør ikke måtte huske informasjon fra en dialog til en annen. Bruksinstruksjoner bør være synlige eller lett tilgjengelige når det er hensiktsmessig.

### 7. Fleksibilitet og anvendbarhet

Akseleratorer, eller snarveier, vil la ekspertbrukeren utføre oppgavene raskere og samtidig ikke forstyrre den nye brukeren, slik at begge kan bruke systemet slik de ønsker.

### 8. Estetisk og minimalistisk design

Dialoger bør ikke inneholde informasjon som er irrelevant eller sjelden brukt. All ekstra informasjon konkurrerer med den viktige informasjonen.

### 9. Hjelp brukeren å gjenkjenne, diagnostisere og gjenvinne kontrollen etter feil

Feilmeldinger bør uttrykkes i naturlig språk, presist indikere problemet og komme med et konstruktivt forslag til løsning.

### 10. Hjelp og dokumentasjon

Av og til er det nødvendig å tilby hjelp og dokumentasjon, selv om det beste er om systemet kan brukes uten dette. Slik dokumentasjon bør være lett å søke i, fokusere på brukerens oppgave og liste opp konkrete steg som kan utføres.

## 2.2.3 Shneidermans «8 Golden Rules»

En av de mest kjente retningslinjene til godt grensesnittdesign er Ben Shneidermans åtte gyldne regler [11]. Disse prinsippene er utarbeidet ut i fra erfaring og er anvendelige for de aller fleste interaktive systemer. Vi vil her presentere og kort forklare disse åtte reglene.

### 1. Streb etter konsistens

Terminologi, menyer og skjermbilder bør være konsistente gjennom hele systemet.

**2. La erfarne brukere få snarveier**

Erfarne brukere ønsker å bruke færrest mulig steg og kortest mulig tid på å utføre oppgaver i et system. Derfor er det viktig å tilby snarveier, hurtigtaster og lignende til slike brukere.

**3. Gi informativ tilbakemelding**

For enhver handling som brukeren utfører bør systemet gi en eller annen form for tilbakemelding. Dette er blant annet viktig for at brukeren skal skjønne at handlingen som er utført er registrert av systemet.

**4. Gi avsluttende tilbakemeldinger på brukers handlinger**

Oppgaver som utføres i systemet bør ha en logisk start og slutt. Ved bruk av tilbakemeldinger kan systemet informere om at en handling er fullført og brukeren vil få en følelse av tilfredshet.

**5. Tilby enkel feilhåndtering**

Systemet bør designes slik at brukeren ikke kan gjøre alvorlige feil. Dersom feil oppstår bør dette håndteres på en enkel og fattbar måte i forhold til brukeren.

**6. Tillatt enkel angring**

Alle operasjoner som brukeren utfører i systemet bør kunne angres. Dette oppmuntrer brukere til å prøve seg fram og ikke være redde for å gjøre feil under bruk.

**7. Gi brukeren følelsen av kontroll**

Systemet bør designes slik at erfarne brukere føler de har kontroll og at systemet reagerer på deres handlinger.

**8. Begrens bruk av korttidsminne**

Bruk enkle skjermbilder og begrens mengde informasjon som må huskes av brukerne.

## 2.3 Fri programvare

Da prosjektet vårt i stor grad omhandler temaene åpen kildekode og fri programvare vil vi i denne seksjonen gi en kort innføring i disse begrepene. I tillegg vil vi se nærmere på historikken og økonomiske motiv knyttet til fri programvare.



### 2.3.1 Hva er fri programvare?

Forskjellene på de to begrepene *fri programvare* og *åpen kildekode* er i hovedsak mer ideologiske enn praktiske og har liten praktisk relevans for dette prosjektet. Likevel føler vi at det kan være nyttig å oppklare noen av motsetningene mellom disse begrepene.

Dersom vi oversetter åpen kildekode direkte til det engelske ordet «Open Source» er dette definert ut fra ti krav satt til distribusjon av åpen kildekode-programvare formulert av The Open Source Initiative [12]. Kravene kan oppsummeres slik:

- Fri redistribusjon av programvaren.
- Fritt tilgjengelig kildekode.
- Programvaren kan modifiseres og avledes til nye programmer.
- Modifiserte versjoner av programvaren kan måtte distribueres som patcher.
- Distribusjon av programvaren skal ikke diskriminere personer eller grupper av personer.
- Distribusjon av programvaren skal ikke diskriminere bruksområder.
- Programvarens lisens skal videreføres i alle redistribuerte utgaver.
- Programvarens lisens skal ikke være avhengig av et bestemt produkt.
- Programvarens lisens skal ikke sette restriksjoner på annen programvare.
- Programvarens lisens skal være teknologi-nøytral.

Denne definisjonen var i utgangspunktet en reaksjon på forvirringen rundt begrepet «free software» i meningen «gratis programvare», men åpen kildekodebegrepet utviklet seg etter hvert i en litt annen retning enn fri programvare. Åpen kildekode handler mer om utviklingsmetodikken og tilhengerne er ofte mer opptatt av praktiske hensyn og økonomiske og tekniske fordeler, i kontrast til fri programvare som har mer fokus på etiske og moralske hensyn [13].

Fri programvare-begrepet baserer seg på at brukeren av programvaren gis fire friheter [14]:

**Frihet 0:** Friheten til å kjøre programmet uansett hensikt.

**Frihet 1:** Friheten til å studere hvordan programmet virker og tilpasse det til sine behov.

**Frihet 2:** Friheten til å distribuere kopier for å hjelpe andre.

**Frihet 3:** Friheten til å utgi sine forbedringer til offentlig eie, slik at hele samfunnet kan få utbytte.

Det er tydelig at en forutsetning for friheten til å studere hvordan et program virker, tilpasse det til sitt behov og utgi dette er at kildekoden til programvaren er gjort fritt tilgjengelig for alle. Det er nettopp også denne forutsetningen som for folk flest kjennetegner åpen kildekode eller fri programvare.

I bunn og grunn kan vi si at begrepene fri programvare og åpen kildekode omhandler selve programvaren med kildekoden og utviklingsmetodikken basert på ideen om at offentligheten kan bidra til å utvikle og forbedre samt redistribuere programvaren. På bakgrunn av dette vil vi i denne rapporten benytte oss av begge begrepene, åpen kildekode og fri programvare, uten å betrakte de ideologiske og «juridiske» forskjellene mellom disse.

### 2.3.2 Økonomiske motiver

Å tjene penger på fri programvare høres i utgangspunktet kanskje ut som en selvmotsigelse. Likevel finnes det mange programvare-selskaper som lever av dette gjennom å ta i bruk metoder som baserer seg på verdiøkende tjenester. Vi vil her kort presentere noen av de vanligste strategiene som brukes for å tjene penger på fri programvare.

- Programvaren i seg selv er gratis, men selskapet tilbyr support og andre ettersalgs-tjenester mot betaling.
- Programvaren distribueres fritt med en intensjon om å øke salget av ikke-frie, kommersielle produkter i selskapets portefølje.
- Tilby fri programvare som støtteprogrammer eller drivere til maskinvare.
- Selge tilbehør relatert til fri programvare, som for eksempel bøker og cd-er.
- Distribuere fri programvare under hybridlisens. Det vil si at programvaren er gratis til privat bruk, men at bruk i en kommersiell setting eller i et kommersielt produkt krever betaling.

### 2.3.3 Historikk

I starten av dataalderen ble all programvareutvikling i stor grad utført av ingeniører og forskere ved de tyngste universitetene og forskningsmiljøene i USA og Europa. Miljøet var lite og for å kunne skape gode nok løsninger var man avhengig av å dele informasjonen og kunnskapen om teknologi og utvikling med hverandre.

Med skapelsen og innføringen av Internett på 70-tallet ble mange av de store forskningsmiljøene koblet sammen. Dette førte til større grad av åpenhet mellom miljøene og utviklingen av åpne løsninger økte. Utover 90-tallet vokste åpen kildekode-miljøet fram også utenfor de akademiske miljøene og vi fikk en rekke kjente åpne programvare-prosjekter som Linux, Apache og Mozilla.

I dagens IT-marked er vi omgitt av fri programvare på alle kanter og det er først og fremst Internett-revolusjonen som skal ha æren for dette. Mozilla Firefox er åpen kildekode og er i dag en av de mest populære nettleserne i bruk verden over [15], mange av de mest besøkte nettstedene i verden er basert på åpne programmeringsspråk som PHP, Python og Perl mens Apache web-tjeneren har en markedsandel på over 50% [16].

Det er ingen tvil om at fri programvare har stor påvirkningskraft og innflytelse i dagens IT-marked og det er blant annet estimert i en studie fra 2008 at fri programvare årlig koster den kommersielle programvareindustrien nærmere 60 milliarder dollar [17].

## 2.4 Brukbarhet i fri programvare

I denne seksjonen vil vi se på den økende populariteten og utviklingen til fri programvare og hvilke utfordringer dette bringer med seg i forhold til brukbarhet. I tillegg vil vi presentere et utvalg brukbarhetsstudier som er gjort av åpen kildekode-prosjekter, hvilke problemer som er funnet relatert til brukbarhet og diskutere muligheter for å takle disse utfordringene.

### 2.4.1 Hvorfor fri programvare?

Som vi var inne på i Seksjon 2.3.3 har fri programvare en sterk posisjon i IT-verdenen og programvare basert på åpen kildekode er i dag i bruk både i infrastruktur, hos bedrifter og av den «vanlige» hjemmebrukeren. Fri programvare er ikke lenger kun utviklet av «hackere» for «hackere», men også

for å tilfredsstille behovene til den normale bruker. Denne transformasjonen av fri programvare til en mer kommersiell og mainstream form er blant annet blitt døpt OSS<sup>2</sup> 2.0 [18].

Det er mange grunner til at fri programvare har fått den statusen og populariteten den har i dag, både hos private brukere og i industrien. Høy kvalitet og pålitelighet, raske utviklingscykler og ofte utgivelser, samt lavere kostnader ved utvikling og eierskap er noen av grunnene til at fri programvare representerer et godt alternativ til kommersiell programvare. Hos utviklere og i det akademiske miljøet er de etiske og sosiale årsakene også viktige. Mange er opptatt av å styrke fri programvare-miljøet og samtidig bygge et godt rykte og utvikle sine ferdigheter gjennom å bidra i åpen kildekode-prosjekter [19].

Men denne økende populariteten bringer også med seg utfordringer. Flere og flere brukere av fri programvare vil være vanlige brukere uten dyp teknisk innsikt. Derfor vil andre kvalitetskrav enn før være gyldige, da fri programvare først og fremst var utviklet for ekspertbrukere og andre utviklere [20]. Det kanskje viktigste nye kvalitetskravet til fri programvare er brukbarhet.

### 2.4.2 Utfordringer

Tidligere forskning gjort på området åpen kildekode og brukbarhet hevder at fri programvare-miljøet går gjennom en lignende brukbarhetsrevolusjon som den kommersielle programvareindustrien gjorde på 1980-tallet [21]. Men selv om mye har skjedd siden åpen kildekode-miljøets spede begynnelse på 70-tallet er det fortsatt mye som skiller denne leiren fra de store kommersielle programvareselskapene. Og det er dette som gjør brukbarhetsutfordringen så vanskelig og interessant.

#### Motivasjon

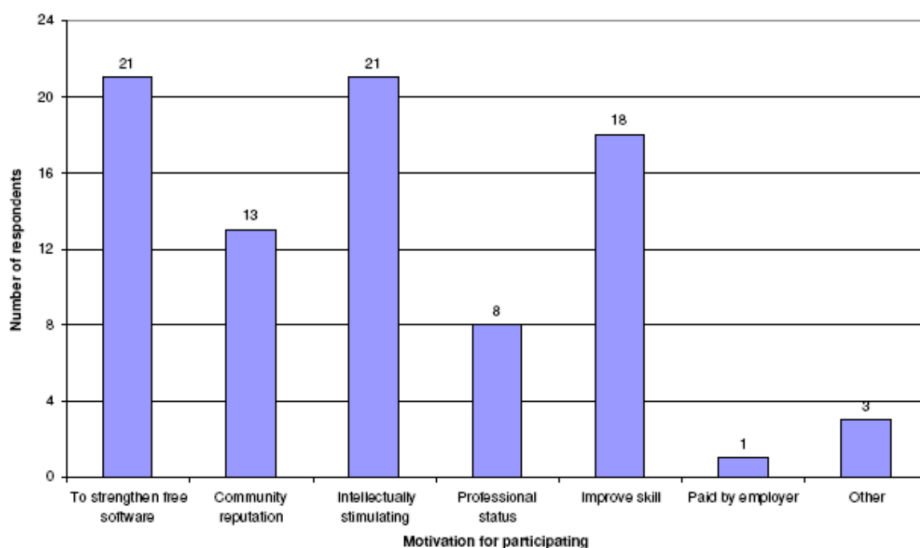
Drivkreftene og motivasjonen bak utviklingen av fri programvare skiller seg fra den tradisjonelle programvareutviklingen på mange måter. Kommersiell programvare er først og fremst laget for å tilfredsstille behov hos en gruppe brukere og få solgt programvaren til flest mulig av disse. Tradisjonelt har motivasjonen bak åpen kildekode vært av mer «egoistisk» art, hvor utviklingen startet for å oppfylle et behov hos en utvikler eller løse et teknisk problem.

---

<sup>2</sup>Open Source Software

## 2.4. BRUKBARHET I FRI PROGRAMVARE

---



**Figur 2.1:** Motivasjon for utviklere til å bidra i fri programvare

Denne motivasjonen gjør også at fri programvare ofte vokser i kompleksitet over tid, ettersom ny funksjonalitet etterspørres og eksisterende funksjoner består. De erfarne brukerne som hadde behov for programvaren i utgangspunktet setter pris på denne friheten til å nærmest velge og vrake i funksjoner og «features». Samtidig vil denne økte kompleksiteten redusere brukervennligheten for normale, mindre erfarne brukere [21].

En studie fra 2006 [22] har gjort en undersøkelse blant utviklere i åpen kildekode-prosjekter og presenterer noen av de vanligste motivene for å bidra i slike prosjekter, gjengitt i Figur 2.1. Blant de som ble intervjuet og svarte på spørreskjemaer er rundt 80% med for å utvikle sine ferdigheter og bli utfordret. Enda flere peker på ideologiske årsaker som sin motivasjon og har et ønske om å styrke fri programvare. Likevel viser det seg at mange av brukerne er opptatt av brukbarhet, selv om oppfatningen av hva som ligger i begrepet varierer mye over de spurte.

### Prosjektorganisering

Det er også vesentlige forskjeller på fri programvare og kommersiell programvare når det gjelder organisering av utviklingen og tilgangen til ressurser. Stereotypen av åpen-kildekode-miljøet er typisk hackere som utvikler programvare uten å forholde seg til formelle prosesser, utviklingsmetodikk og med et anstrengt forhold til autoriteter [23].

Det ligger nok også mye sannhet i dette. Fri programvare er fortsatt i stor grad desentraliserte prosjekter med flat struktur, hvor bidragsyterne er distribuert utover et stort geografisk område og gjerne over flere tidssoner. De som deltar i prosjektene gjør det av egen interesse og fri vilje og i sin egen fritid. Dette gjør det selvsagt vanskelig å forholde seg til tradisjonelle utviklingsmetoder og beslutningsprosesser.

De har heller ikke ressursene til å bedrive lang-distanse videokonferanser eller møter ansikt-til-ansikt, som er en viktig forutsetning for tradisjonelle brukbarhetsmetoder. I åpen kildekode-prosjekter foregår det aller meste av kommunikasjonen over e-post-lister eller IRC<sup>3</sup>. Alle disse faktorene bidrar til at konvensjonelle brukbarhetsmetodologier må tilpasses før de kan integreres problemfritt i åpen kildekode-miljøer, og de gjør det vanskelig for brukbarhetsekspertene å delta direkte i prosjektene.

### Oppfatning av brukbarhet

Som vi har vært inne på tidligere har ikke fri-programvare-miljøet tradisjonelt vært spesielt opptatt av brukbarhet. Programvaren har vært utviklet for en teknisk kyndig brukergruppe med fokus på å løse tekniske problemer snarere enn fokus på brukbarhet og brukervennlighet. Derfor har grensenittdesignet vært lite kreativt og innovativt og stort sett fulgt i fotsporene til kommersiell programvare hva utseende angår [21]. Det er kanskje ikke tilfeldig at mange av de mest suksessrike åpen kildekode-prosjektene er operativsystemkjerner (eks. Linux) og kompilatorer (eks. GCC) som krever liten grad av menneske-maskin interaksjon [24].

Likevel ser vi at brukbarhet har fått et sterkere fokus i åpen kildekode-prosjekter den siste tiden. I en fersk undersøkelse [25] fra fri-programvare-miljøet ser vi at mange er opptatt av brukbarhet og motivene som ligger bak for å bedre denne. Sosiale faktorer som positive tilbakemeldinger og fornøyde brukere blir sett på som en viktig drivkraft for å utvikle høykvalitets programvare med god brukbarhet.

Selv om interessen for brukbarhet er økende er fortsatt oppfatningen og forståelsen av begrepet som tatt ut av de kjente lærebøkene om temaet. Blant deltagerne i den overnevnte undersøkelsen går begreper som lærbarhet, effektivitet, huskbarhet og tilfredshet igjen.

---

<sup>3</sup>Internet Relay Chat: Synkron, tekstbasert sanntids-chat over Internett.

### 2.4.3 Brukbarhetsstudier av fri programvare

Det er tydelig at det er mange områder som skiller fri programvare-utvikling fra tradisjonell kommersiell programvare og som potensielt kan føre til brukbarhetsproblemer. Men hvordan kan man sammenligne brukbarheten i åpen kildekode med lukket kildekode? Det er en rekke faktorer som bidrar med usikkerhet rundt en slik direkte sammenligning; utviklingstid og ressursbruk er én ting, programvarens modenhet og tidligere utgaver av lignende programmer er en annen. Åpen kildekode-prosjektenes kontinuerlige og inkrementelle utvikling kan gjøre det vanskelig å utføre en «rettferdig» sammenligning [21].

Noen brukbarhetsstudier av fri programvare er likevel gjort. Studiene av en åpen læringsplattform [19] og biblioteksprogramvaren Greenstone [26] avdekker mange av de «klassiske» brukbarhetsproblemene man også finner igjen i proprietær programvare. Disse bunner hovedsakelig ut i forskjeller mellom utviklerne og brukerne i forhold til kunnskap og forventninger til programvaren. I tillegg nevner begge studiene mangelfull og/eller uklar dokumentasjon som et vesentlig problem, samt at én av de ikke innfrir retningslinjer for tilgjengelighet for marginale brukergrupper.

En case-studie av programvare for modellering av Gantt-diagrammer [24] finner utfordringer relatert til rapportering og håndtering av brukbarhetsproblemer i fri programvare. De utførte en såkalt protokollanalyse, hvor en bruker tenkte høyt mens han utførte en stegvis prosedyre i programmet. Selve analysen avdekte ti brukbarhetsproblemer, og av disse ti har kun fire blitt løst og implementert (over en periode på ett år etter testen). Det er også verdt å nevne at kun to av problemene ble identifisert og rapportert av andre brukere av programvaren og samlet sett ble bare tre brukbarhetsproblemer funnet i prosjektets bugrapporteringssystem.

Bugrapporteringssystemer, som for eksempel Bugzilla<sup>4</sup> blir brukt i mange åpen kildekode-prosjekter, men har vist seg å være for komplekse og hovedsakelig tilpasset funksjonelle feil og følgelig lite egnet for brukbarhetsproblemer [27].

### 2.4.4 Muligheter og trender

Utfordringene relatert til brukbarhet i fri programvare vi nå har vært inne på åpner for mange muligheter til forbedring. I skrivende stund ser vi flere trender innenfor miljøet som tar tak i brukbarhetsproblemer. Til tross for

---

<sup>4</sup>Web-basert bugtracker utviklet av Mozilla (<http://www.bugzilla.org/>)

forskjeller i organisasjonsstruktur og flere sosiale barrierer er det i dag mange store kommersielle selskaper som bidrar aktivt med brukbarhetseksptise i fri programvare-miljøet [28]. Selskaper som Sun, HP og IBM driver også egne åpen kildekode-prosjekter og sitter inne med mye brukbarhetskunnskap som kan berike miljøet [23].

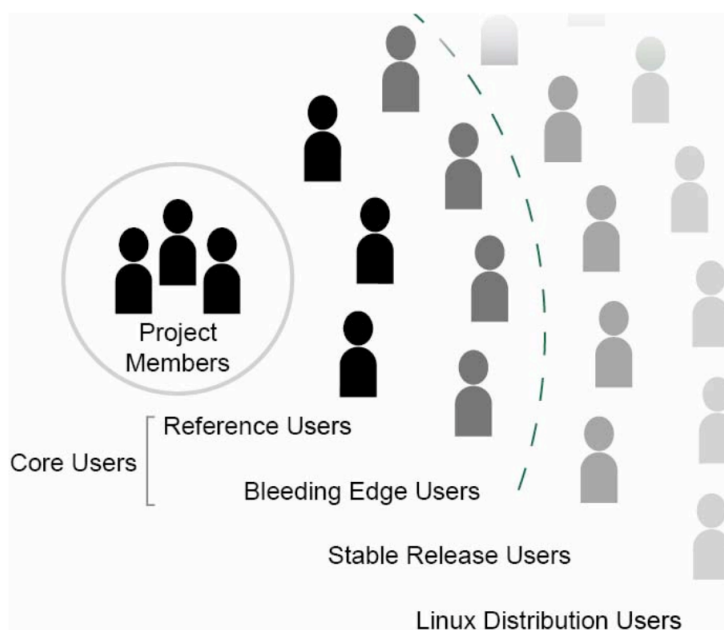
Det er også satt i gang flere prosjekter innenfor fri programvare-miljøet med mål om økt fokus på brukbarhet og brukervennlige løsninger. Wikimedia, stiftelsen bak blant annet Wikipedia, har satt i gang Wikipedia Usability Initiative [4] som har som mål å målbart øke brukbarheten til Wikipedia for bidragsyterne. Dette initiativet inkluderer flere faser med brukerundersøkelser, evalueringer og brukbarhetstester. Et annet eksempel er OpenUsability [29]; et initiativ som jobber med å fokusere på brukbarhet, dele metoder og «best practices» innenfor fri programvare-miljøet og er involvert i en rekke store prosjekter.

Innenfor utviklingsmetodikk ligger det også forbedringspotensiale og muligheter. En vesentlig faktor er å utnytte interaksjonen som foregår mellom utviklere og de engasjerte brukerne, såkalte kjernebrukerne, av fri programvare [25]. Dette er en av de viktigste brukbarhetsmetodene innenfor fri programvare i dag. I Figur 2.2 er kjernebrukerne vist som såkalte «Reference Users», det vil si brukere med høy domenekunnskap og erfaring i å bruke programvaren. «Bleeding Edge Users» tilhører gruppen av brukere som tar i bruk de ferskeste utgavene av programvaren. Dersom man involverer disse brukerne i å teste ut og evaluere brukbarheten til programvaren før endelige utgivelser vil man lettere kunne finne brukbaretsproblemer og samtidig få verdifulle tilbakemeldinger fra en kunnskapsrik brukergruppe.

En mulig fallgruve ved denne metoden er at grensenettet potensielt blir skreddersydd disse «ekspertbrukerne» og ikke møter behovet til vanlige brukere. Et mindre ressurskrevende alternativ til standard, formell brukbarhetstesting med reelle brukere kan være å involvere en typisk bruker i fjernstyrt brukbarhetstesting [24].

Det finnes også et behov for å utvikle bedre systemer for rapportering av brukbarhetsproblemer i fri programvare, som vi var inne på i slutten av Seksjon 2.4.3. Slike problemer får ofte lavere status blant utviklerne i åpen kildekode-prosjekter da de blir sett på som subjektive og upålitelige og lar seg ofte vanskelig reproduseres. I tillegg kan det være vanskelig å beskrive brukbarhetsproblemer tekstlig og isolere de til en bestemt kontekst i programmet [27].





**Figur 2.2:** Brukere av et fri programvare-prosjekt

En artikkel fra CHI 2010<sup>5</sup> har sett på hvordan man kan integrere eksisterende brukbarhetsprinsipper og metoder i Bugzilla-verktøyet [30]. I skrivende stund jobbes det med å integrere denne heuristiske evalueringen i Bugzilla-instansene som brukes av Mozilla Firefox og Thunderbird. Alle rapporterte brukbarhetsproblemer blir koblet til relevante brukbarhetsprinsipper som er nøye definert og forklart med eksempler i verktøyet. Artikkelen hevder at om man bygger slike verktøy på veletablerte brukbarhetsheuristikker, som Jakob Nielsens ti punkter [10], vil man kunne redusere antallet diskusjoner rundt subjektiviteten og upåliteligheten til rapporterte brukbarhetsproblemer og oppnå en felles forståelse av disse prinsippene i miljøet.

## 2.5 Integrasjon av ferdigkomponenter

COTS<sup>6</sup> er betegnelsen på ferdigprogramvare-komponenter som anskaffes fra kommersielle aktører og settes sammen i en produktlinje [31]. Dette kan også inkludere kommersialisert fri programvare, men i de fleste tilfeller er ikke kildekoden til COTS-komponenter tilgjengelig. Dette er den store begrens-

---

<sup>5</sup>ACM Conference on Human Factors in Computing Systems

<sup>6</sup>Commercial Off The Shelf

ningen til COTS-komponenter i forhold til åpen kildekode-komponenter, der muligheten for å endre på arkitektur og kodebase er mye større. Åpen kildekode-komponenter har derimot større utfordringer enn COTS hva gjelder for eksempel mangelfull support og dokumentasjon, eller at de ikke følger standarder.

Di Giacomo [32] har diskutert mange av forskjellene på COTS-komponenter og åpen kildekode-komponenter. Han finner blant annet ut at ved valg av åpen kildekode-komponenter er det andre faktorer som blir viktige enn ved COTS-valg. Lisensbetingelser, community<sup>7</sup> og aktivitet på prosjektet kan være avgjørende. Samtidig har mange fri programvare-prosjekter blitt mer profesjonelle og tatt i bruk forretningsmodeller fra den kommersielle programvareindustrien. Velger man slike komponenter forsvinner mange av risikoene ved fri programvare, som mangelfull support og dokumentasjon.

Overnevnte argumenter gir derfor grunn til å betrakte mange åpen kildekode-komponenter omtrent som COTS og vi vil på grunnlag av dette benytte oss av eksisterende litteratur rundt COTS-komponenter og integrasjon i denne seksjonen. Vi observerer også at det finnes mer tilgjengelig litteratur som omhandler COTS og integrasjon, enn åpen kildekode-komponenter og integrasjon.

Boehm [33] argumenterer for at det ved utvikling basert på COTS-komponenter vil være feil å benytte seg av en typisk vannfall utviklingsmodell. En slik utviklingsmodell tar utgangspunkt i ganske spesifikke krav, og disse kravene bestemmer hva som blir levert i det endelige produktet. Når en benytter seg av COTS-komponenter må en se på hvilke muligheter de ulike komponentene tilbyr, fordi disse vil i stor grad være med på å forme den leverte funksjonaliteten. Det er derfor viktig å holde kravene diskuterbare til de overordnede komponentene er på plass.

Li et al. [34] utførte i 2006 en utforskende studie av hvordan utvikling med COTS-komponenter ble gjennomført i praksis i norske IT-bedrifter. Denne studien underbygger påstandene om at en iterativ utviklingsmodell er å foretrekke. Majoriteten av de 16 prosjektene som ble undersøkt brukte en inkrementell utviklingsmetode med noe prototyping. Studien viser også at risikoen er lavere med en slik modell når man velger ukjente komponenter. Et viktig kriterium for et vellykket COTS-prosjekt er å velge komponenter basert på hvor lette de er å integrere. En må derfor fokusere på arkitektur, og ikke bare på funksjonaliteten som komponentene tilbyr.

---

<sup>7</sup>Community er brukt som betegnelsen på et fellesskap av likesinnede innenfor et domene, for eksempel et «hacker-community».

## 2.5. INTEGRASJON AV FERDIGKOMPONENTER

---

Det medfølger stor risiko ved integrasjon av COTS-komponenter. De fleste COTS-komponenter er ikke designet for å interoperere med hverandre [33, 35]. Det er derfor lurt å integrere komponentene tidlig i utviklingsløpet for å få avdekket de mest avgjørende og risikofylte momentene på et tidlig stadie. En av de mest vektlagte case-studiene av COTS-integrering er gjort av Garlan [35]. Garlans to-manns, seks måneders prosjekt endte opp som et fem-personers, to-årig prosjekt.

Et typisk trekk ved ferdigvare-komponenter, både COTS og åpen kildekode, er at de er utviklet for én brukergruppe, og tilpasset denne gruppens behov og ønsker. Ved integrasjon av flere slike komponenter kan det være problematisk å tilpasse de integrerte komponentene til sin egen brukergruppe. Et av problemene som kan oppstå er at det ikke er noen felles mal for brukbarhet. En kan ende opp med å få et system som fungerer funksjonelt, men som mangler et helhetlig grensesnitt utad. Dette kan gjøre det endelige produktet vanskeligere å bruke, da de ulike komponentene kan ha forskjellige måter å presentere informasjon på. I tillegg kan komponentene følge ulike standarder for design og grafiske elementer som kan føre til brukbarhetsutfordringer [36].

Hoffman og Cook [36] forsøkte å redusere problemer knyttet til brukbarhet under utviklingen av et kommandosupportsystem basert på COTSkomponenter for det britiske forsvaret. De tok i bruk en «style guide» som de evaluerte komponentene opp mot, og tilpasset ny programvare til, med det overordnede målet å oppnå et konsistent og helhetlig brukergrensesnitt for hele systemet. Hoffman og Cook mener det er et stort potensiale for å evaluere komponenter opp mot en slik stilveiledning før en komponent blir anskaffet, for å få et bedre bilde av i hvilken grad komponenten kan passe inn med resten av systemet.



# Kapittel 3

## Forskningsdesign

Dette kapitlet beskriver metodene vi har brukt for å best mulig besvare problemstillingen og forskningsspørsmålene presentert i Seksjon 1.3. Etter å ha beskrevet metodene generelt, vil vi gjøre rede for det endelige valget av metoder og forskningsdesign.

### 3.1 Generelt om metoder

I denne seksjonen vil vi beskrive generelle forskningsmetoder som er aktuelle i denne oppgaven.

#### 3.1.1 Aksjonsforskning og eksperiment

Vil vil her kort presentere de to forskningsmetodene aksjonsforskning og eksperiment. Det kan argumenteres for at vi vil benytte oss av begge metodene i dette prosjekt, noe vi kommer nærmere inn på i Seksjon 3.3.1.

Aksjonsforskning<sup>1</sup> handler om å gå inn i en ekte instans av en organisasjon, utviklingsprosjekt eller lignende og utføre en endring eller gjøre et arbeid som er av praktisk nytte for denne instansen. Målet med dette er å generere ny kunnskap i form av lærdommer og erfaringer som resultat av det som ble gjort og reflektere over disse [37]. Fokuset i aksjonsforskning er på å kombinere teori og praksis gjennom at forskere og de som er involvert i selve caset jobber sammen for å oppnå verdi for begge parter [38].

---

<sup>1</sup>Kanskje bedre kjent under navnet «Action Research» på engelsk.

Et eksperiment innenfor systemutvikling vil ofte ha mindre grad av realisme, men mer kontroll over variabler enn hva aksjonsforskning har i denne sammenheng. Generelt utføres eksperimenter for å utforske årsak-virkning-forhold og bevise eller motbevise hypoteser gjennom å undersøke et fenomen i kontrollerte forhold. Eksperimentet observeres og målinger gjøres for å kunne dra noen slutninger om hypotesene som testes [37].

Innenfor systemutvikling gjøres eksperimenter hovedsakelig for å utvikle nye eller forbedre eksisterende prosesser, metoder og verktøy. Sjøberg et al. [39] påpeker at et viktig mål for eksperimenter i systemutvikling er å overføre kunnskapen som genereres til industrien. For å kunne gjøre dette må validiteten og nytteverdien til resultatene fra slike eksperimenter økes, gjennom å øke eksperimentenes realisme. Systemutviklingseksperimenter må utføres med ekte oppgaver, på ekte systemer og med ekte utviklere og tilhørende verktøy og omgivelser.

### 3.1.2 Utvikling

Mange forskningsprosjekter innenfor IT inkluderer design og utvikling av et system for å kunne besvare forskningsspørsmål og undersøke hypoteser. Derfor er ofte utviklingen av et IT-system en del av forskningsmetoden som brukes i et prosjekt, gjerne i kombinasjon med et eksperiment eller case-studie. På denne måten kan man undersøke hvordan et IT-system brukes i en naturlig kontekst av ekte brukere.

Utviklingen av et IT-system i forskningssammenheng bør ikke bare resultere i et teknisk vellykket system, men også generere ny kunnskap til å analysere og eller forklare forskningsspørsmål. Det som kanskje skiller vanlig programvareutvikling i industrien fra utvikling i forbindelse med forskning er at et industriprosjekt er mest vellykket dersom alt går etter planen. I forskning derimot er det gjerne en fordel dersom problemer oppstår eller endringer må gjøres underveis. På denne måten lærer man mer og man får generert ny kunnskap [37].

Da man har utviklet et IT-system er det vanlig å foreta en evaluering av systemet i en naturlig kontekst basert på ulike kriterier. Dette kan for eksempel være pålitelighet, funksjonalitet, tilgjengelighet eller brukbarhet. Man kan også benytte seg av datainnsamlingsmetoder som intervju og observasjon av brukere av systemet.

### 3.1.3 Intervju

Et intervju kan karakteriseres som en samtale mellom to eller flere personer med et bestemt mål eller agenda. Som datainnsamlingsmetode i forskningsarbeid er intervju vanlig å bruke i forbindelse med blant annet «case-studier» og etnografisk arbeid. Intervju egner seg spesielt godt når man ønsker å skaffe detaljert informasjon eller få svar på spørsmål som er for komplekse og åpne til at de enkelt kan observeres eller svares på i skriftlig form gjennom f.eks. spørreskjemaer. Ofte kan det også kombineres med de nevnte metodene ved at man utfører et intervju som oppfølging til observasjon eller for å få nærmere innsikt i svarene som er gitt i et spørreskjema.

Det finnes i hovedsak tre typer intervjuer [37]; strukturert, semi-strukturert og ustrukturert intervju.

**Strukturert intervju** I et strukturert intervju er alle spørsmålene som skal stilles definert på forhånd og de samme spørsmålene stilles til hver respondent. Det er heller ikke meningen at intervjueren skal komme med egne kommentarer eller følge opp svarene til respondenten på noen måte, og et strukturert intervju kan derfor betraktes som en form for spørreundersøkelse.

**Semistrukturert intervju** Et semistrukturert har noen temaer eller spørsmål listet opp og forberedt på forhånd, men er mindre låst til disse enn et strukturert intervju. Man kan tilpasse seg og endre spørsmålsrekkefølgen etter hvordan samtalen flyter under intervjuet og man kan komme med oppfølginsspørsmål underveis.

**Ustrukturert intervju** I denne formen vil intervjueren presentere et overordnet tema og la respondenten fritt komme med sine egne tanker, idéer og meninger rundt det. Intervjueren bør fokusere på å la respondenten snakke og forstyrre så lite som mulig.

For å samle inn data fra et intervju benytter men seg gjerne av lydopptak, videopptak eller notater.

### 3.1.4 Spørreskjema

Spørreskjemaer er en datainnsamlingsmetode velegnet for å sende ut til en større gruppe mennesker. De inneholder et fast antall spørsmål i en fast rekkefølge og gir en større mengde data som kan analyseres og tolkes [37]. Det

er vanlig å bruke data fra spørreskjema til å se etter mønstre og generalisere funnene til en større populasjon enn utvalget som har svart på spørreundersøkelsen.

Når man designer et spørreskjema er det viktig å gjøre det på en slik måte at det gir pålitelige og gyldige data og at spørsmålene blir forstått likt av alle respondenter. Blant de vanligste måtene å formulere spørsmålene i en spørreundersøkelse har vi:

- Ja/Nei-spørsmål
- Kvantitative spørsmål
- Enig/uenig-påstander
- Påstander med grad av enighet/uenighet (Likert skala)

## 3.2 Brukbarhetsmetoder

Det finnes mange metoder for å måle brukbarhet og for å avdekke brukbarhetsproblem, og et utall varianter av disse. To av de største metodene er spesielt anvendbare i denne sammenheng, og vi vil derfor gå nærmere inn på disse. Vi kan se på disse som to skoler som kjemper for hver sin sak. Den første skolen tar i bruk heuristisk evaluering for å finne brukbarhetsproblemer underveis i utviklingsprosessen, mens den andre gjennomfører tester med brukerne for å oppdage brukbarhetsfeil ved systemet.

### 3.2.1 Heuristisk evaluering

Hensikten med heuristisk evaluering er å finne brukbarhetsproblemer tidlig i utviklingen så de kan rettes opp som en del av den iterative designprosessen. Evalueringen gjennomføres av en til tre evaluatorene. Evaluatorene undersøker brukbarhetsgrensesnittet individuelt og måler det opp mot et sett av brukbarhetsprinsipp. Jakob Nielsens ti punkter og Ben Shneidermans 8 gylne regler, som introdusert i Seksjon 2.2.2 og 2.2.3, er eksempler på ofte brukte heuristikker.

Resultatet fra en heuristisk evaluering er en liste med potensielle brukbarhetsproblemer, og ideelt sett er disse problemene koblet til et eller flere brukbarhetsprinsipp for å bedre forstå problemet. Fordeler med en heuristisk evaluering inkluderer at metoden gir et godt utbytte med tanke på den relativt lave kost-



## 3.2. BRUKBARHETSMETODER

---

naden. Metoden gir også et resultat tidlig i designprosessen, og kan dermed gjøre endringer mindre omfattende.

Blant ulemper kan det nevnes at det kreves en del kunnskap for å gjennomføre en god heuristisk evaluering, og personer med slike ressurser er ikke alltid for hånden. Et annet problem er at en kan ende opp med flere små problemer og færre alvorlige problemer [40]. Dette kalles gjerne «lokale problemer». Det vil si mindre problemer som er en del av én dialog eller meny, for eksempel et ord eller inputfelt, i systemet. Lokale problemer er noe som ikke nødvendigvis berører alle brukere [41]. Det er ikke sikkert at å utbedre slike problemer vil bedre den opplevde brukbarheten nok til at det er verdt det.

### 3.2.2 Brukbarhetstesting

Brukbarhetstesting er en teknikk brukt for å evaluere et produkt ved å teste det på representative brukere. Brukerne får oppgaver de skal utføre ved å bruke produktet samtidig som de blir overvåket av observatører. En slik test kan avdekke brukbarhetsproblemene ved systemet, måle brukertilfredshet, måle effektiviteten ved å bruke systemet og gi innspill til forbedringer [42]. Det er derfor lurt å teste tidlig i utviklingsfasen slik at problemer og muligheter kan avdekkes på et tidlig tidspunkt og dermed integreres før utviklingen har gått for langt (selv om teorien sier det bør være mulig å endre design og funksjonalitet til en hver tid, blir det ofte slik at noen ting blir «hugget i stein»).

Brukbarhetstesting avdekker ofte flere unike problemer enn andre evalueringsmetoder, og problemene er ofte større og alvorligere enn de som blir avdekket ved for eksempel en heuristisk evaluering. Vi kan kalle dette globale problemer; problemer som omhandler store deler av systemet, som at valg og info er dårlig strukturert over alle menyene, og som berører mange brukere [41].

Det finnes ulike typer av brukbarhetstesting, og ofte utføres disse på ulike stadium i utviklingsprosessen [42]. På et tidlig stadium kan det være nyttig å bruke utforskende tester. Her er fokuset på å oppnå kvalitative resultat ut i fra et høyere nivå design av produktet. Designet kan gjerne være på skisseplan ennå, slik at brukertesten utføres for eksempel med lapper som endre og flyttes på. I den andre enden av skalaen finner en verifiseringstester som fokuserer på å oppnå kvantitative data. Dette er for eksempel noe som foregår når et produkt beta-testes.

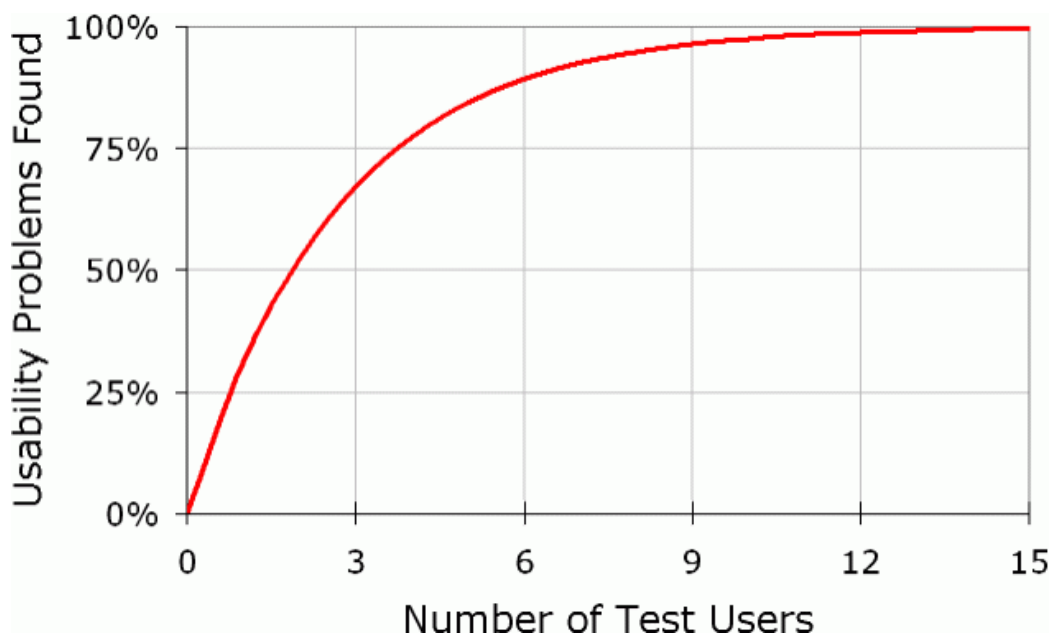
Den vanligste formen for brukbarhetstester er vurderingstester, der både kvalitative og kvantitative data blir innhentet. En slik test utføres gjerne tidlig eller midtveis i designprosessen, når et produkt begynner å finne sin endelige form, men ikke er «ferdigspikret» ennå. Dette er den type brukbarhetstest som vil være aktuell for dette prosjektet.

Under gjennomføring av en brukbarhetstest er det viktig å informere testobjektet før testen starter. Testobjektet må gjøres klar over følgende punkter [43]:

- Formålet er ikke å teste personen, men heller avdekke feil og mangler ved systemet. Brukeren bør ikke være redd for å gjøre feil eller dumme seg ut.
- Det blir som regel brukt verktøy under testen, slik som kamera og lydopptak. Avhengig av stadiet av brukertest kan det være annet spesielt utstyr også.
- Testobjektet bør «tenke høyt», og det kan være lurt å lære bort hvordan en gjør dette. Slik får observatørene bedre innblikk i hvorfor personen handler som han gjør.
- En bør introdusere «settingen», hvilken situasjon personen skal innbille seg han er i når oppgavene utføres og hva som er hensikten med produktet.

### Antall testpersoner

Opp igjennom brukbarhetsforskningen har det vært noe diskusjon rundt antallet brukere som er optimalt for en brukbarhetstest. Det har blitt en viss konsensus om at fem til åtte brukere er ideelt. Jakob Nielsen og Tom Landauer kom fram til at antallet brukbarhetsproblemer funnet i en brukbarhetstest med  $n$  brukere kan beregnes ved  $N(1-(1-L)^n)$ , der  $N$  er det totale antallet brukbarhetsproblemer og  $L$  er andelen brukbarhetsproblemer avdekket under testingen av en enkelt bruker [44]. En typisk verdi av  $L$  er 31%. Grafen i Figur 3.1 viser dette forholdet [45], og en ser tydelig at en vil oppdage en stor del av problemene ved rundt 5 testpersoner, og at det sjelden er noe poeng å ha flere enn rundt 7-8 personer. En ser riktignok av grafen at dersom en har 15 brukere vil en som regel avdekke alle brukbarhetsproblemene. Nielsen argumenterer derimot for at en heller bør holde seg rundt 5 brukere siden dette er mye mer kost-effektivt, og heller da ha flere iterasjoner med brukertester istedet for en stor gjennomføring.



**Figur 3.1:** Forholdet mellom brukbarhetsproblemer avdekket og antall testpersoner

Nielsen nevner også at dersom produktet har flere ulike brukergrupper med ulik funksjonalitet trenger en flere testpersoner siden det vil være distinkte forskjeller på det som testes. Han argumenterer dog for at det vil være en del overlapp mellom brukergruppens funksjonalitet og at det fundamentale samspillet mellom produkt/nettsiden og brukeren vil være likt. Derfor vil tre til fire testpersoner for hver gruppe være tilstrekkelig.

Andre studier [46, 47] har vist at fem testpersoner ikke er tilstrekkelig for å avdekke 85% av brukbarhetsproblemene, som Nielsen hevder. Spool og Schroeder [46] testet flere nettsider med mange brukere, og i flere av testene viste det seg at fem testpersoner bare avdekker 35% av problemene, et ganske radikalt avvik.

Det er viktig å huske på at denne diskusjonen hovedsakelig passer for brukbarhetstesting i forbindelse med utviklingsprosjekter, hvor man er mest interessert i å finne det optimale antallet testpersoner for å avdekke brukbarhetsproblemer fra et kost-nytte-perspektiv. I forskningssammenheng vil det i mange tilfeller være nødvendig med et større antall testpersoner for å kunne dra noen gyldige konklusjoner med større sikkerhet ut i fra dataene. Et antall på 10-11 testpersoner, som i følge grafen avdekker ca. 95% av problemene, ville vært mer passende i den sammenheng.

### 3.2.3 Verktøy ved brukbarhetstesting

Hvilke resultater en får fra brukbarhetstesten vil variere etter hva en er på utkikk etter og hvilke type brukbarhetstest en gjennomfører. Det finnes flere verktøy som kan brukes til å gi nyttige resultater fra testen. ISO-standarden beskrevet nedenfor foreslår en måte å rapportere resultatene fra en brukbarhetstest.

#### ISO/IEC 25062:2006-standarden

ISO/IEC 25062:2006 er en internasjonal standard for rapportering av resultat fra brukbarhetstester [48]. Denne standarden forteller ikke hvordan en brukbarhetstest skal gjennomføres, men gir retningslinjer for hvordan resultatene fra slike tester kan rapporteres. Standarden fokuserer hovedsaklig på kvantitative resultater med innsamlede data, og ikke på kvalitative resultater.

#### Bruk av eye-tracking

Eye-tracking er å følge stien til en persons blick. Ved å bruke tilgjengelige teknologier er det mulig å følge hvor en person ser på en skjerm. Eyetracking-programvare kan også holde oversikt over hva som vises på skjermen brukeren ser på. Det er vanlig å kombinere bruken av eye-tracking med vanlig brukertesting, slik at en fortsatt får nyttegjøre seg av utbyttet fra en vanlig brukertest, og i tillegg ha muligheten til å studere utbyttet fra eye-trackingen.

Det er vanlig å generere et såkalt varmekart som viser hvor på skjermen personen som blir testet ser mest. Et varmekart er et skjermdump med et lag av farger med ulike nivå, slik som et typisk varmesøkende kamera lager. For eksempel er de røde områdene der brukeren ser oftest. Når en får et større antall brukere kan dette gi pekepinner på hvordan brukeren oppfatter nettsiden, og hvordan de skanner skjermen. Figur 3.2 viser et eksempel på et varmekart [49].

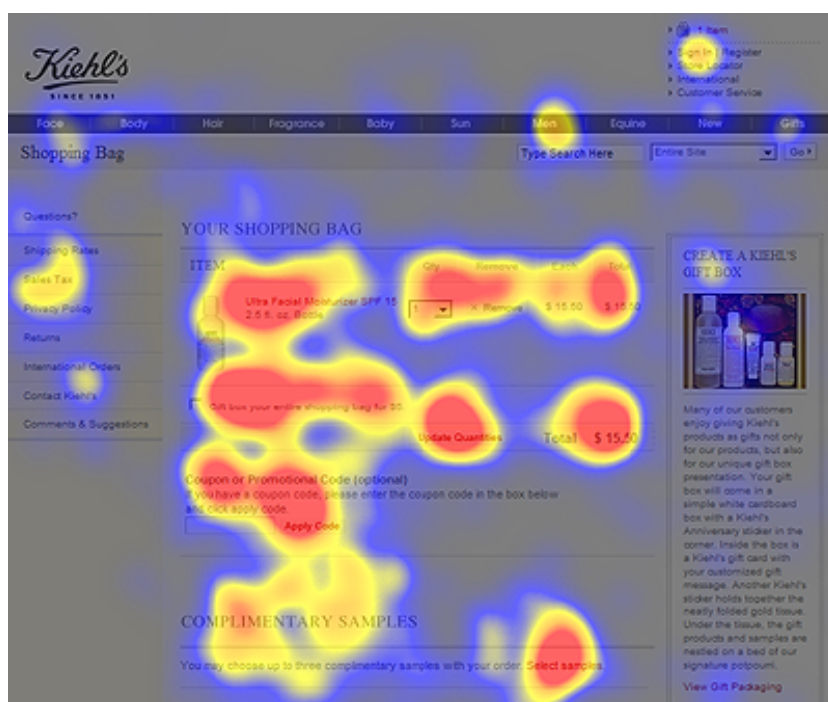
#### SUS-evaluering

Det er vanskelig å gi et absolutt mål for brukbarhet; det kan kun defineres i en spesifikk kontekst. En har likevel ofte et behov for å måle brukbarhet på tvers av brukskontekster, og det er her SUS<sup>2</sup> gjør sin entré. SUS er en

---

<sup>2</sup>System Usability Scale

## 3.2. BRUKBARHETSMETODER



Figur 3.2: Eksempel på resultat fra eye-tracking

pålitelig og kost-effektiv skala for å måle brukbarhet på et generelt nivå [50]. SUS blir ofte brukt som et mål på brukerens tilfredshet ved å bruke systemet.

Evalueringen går ut på at testpersonen svarer på i hvilken grad han er enig eller uenig i ti generelle påstander. Graden av enighet går fra *svært uenig* til *svært enig*, og det er vanlig å ha enten 5 eller 7 grader av enighet, en såkalt Likert-skala. Påstandene omhandler testpersonens oppfattelse av systemet i bruk, og vinkler seg både positivt og negativt til dette. Eksempel på påstander er «Jeg syntes systemet var enkelt å bruke» og «Jeg syntes systemet var unødvendig komplekst».

Basert på svarene gir hver påstand et siffer mellom 0 og 4 (dersom det er 5 grader av enighet). Den totale SUS-skåren regnes da ut ved å ta verdien for påstand 1, 3, 5, 7 og 9 og trekke fra 1 addert med verdien for påstand 2, 4, 6, 8, 10 trukket fra 5. Summen av dette multipliseres med 2.5 og du ender da opp med den totale SUS-skåren for det målte systemet. SUS-skåren vil ligge mellom 0 og 100.

Testpersonen skal svare på SUS-skjemaet rett etter at han har benyttet seg av systemet som testes, før eventuelle oppfølgingsspørsmål blir stilt. SUS-skjemaet er vedlagt i Tillegg A.1.

### **Tilfredshet, anvendbarhet og effektivitet**

Tilfredshet sier noe om hvor tilfreds brukeren er med å bruke systemer og blir gjerne målt ved å bruke for eksempel et SUS-skjema, som forklart i forrige seksjon.

Anvendbarheten er et mål på hvor mange oppgaver testbrukeren klarer å fullføre sammenlignet med hvor mange han ikke klarer, eller hvor mange han trenger hjelp til å fullføre. Antall brukbarhetsproblemer som oppstår blir også brukt som et mål på anvendbarhet. For å evaluere brukbarhetsproblemene bruker en gjerne Molich sin gradering (kritisk, alvorlig og kosmetisk) for å si noe om alvorligheten til problemene.

Effektivitet blir målt ved å ta tiden hver person bruker på å fullføre hver oppgave. Ved å sammenligne to systemer eller metoder kan en se hvilken som er mest effektiv i bruk. Dette kan være svært viktig når det er snakk om livskritiske systemer eller systemer som behandler økonomi.

### **Oppfølgingsintervju**

Det er vanlig å gjennomføre et oppfølgingsintervju av testpersonen som en del av en brukbarhetstest. Det er gunstig å gjøre dette relativt raskt etter oppgavene i testen er gjennomført, gjerne rett etter at SUS-skjema er utfylt, slik at testpersonen har et ferskt inntrykk av systemet som ble testet.

Det er greit å starte et oppfølgingsintervju med generelle aspekter rundt systemet og brukerens helhetsinntrykk før man beveger seg inn på konkrete oppgaver eller deler av systemet [43]. Dersom testpersonen hadde problemer underveis er det lurt å gå dypere inn på hva som var vanskelig og hvorfor for å forstå problemene bedre. Man kan også spørre brukeren om innspill og forslag til endringer ved systemet.

#### **3.2.4 Analyse av brukbarhetstestresultater**

En brukbarhetstest kan gi store mengder data, både kvalitative og kvantitative, som må analyseres og tolkes. Eksempler på kvantitative data kan være tiden en testdeltager bruker på å gjennomføre en oppgave eller antall oppgaver som ikke fullføres. Slike data er egnet for fremstilling i tabeller og grafer og kan deretter analyseres for å se etter mønstre og generelle trender [37]. Det er også vanlig å gjøre dypere statistiske analyser av kvantitative data.

Eksempel på kvalitative data kan være testpersonens tanker og meninger som kommer fram under brukbarhetstesten og i oppfølgingsintervjuet etterpå. Et typisk funn i brukbarhetstester er problemene som oppstod underveis i testen. Det er viktig å analysere disse problemene; forsøke å forstå hvorfor de oppstår, hva de skyldes, samt kategorisere og prioritere de. Når man har gjort dette vil man se etter mønstre i problemene og eventuelt knytte de opp mot design teori og brukbarhetsheuristikker [43]. En slik analyse kan gi alternative designløsninger og input til redesign av systemet som testes.

Det krever ferdigheter å analysere kvalitative data fra brukbarhetstester. Spesielt evnen til å se de bakenforliggende årsakene, globale problemer og mønstre vil ofte kreve flere års erfaring med brukbarhetstesting [51].

## 3.3 Valg av metode

Som beskrevet i Seksjon 1.3 er det overordnede målet med denne oppgaven å undersøke hvilke utfordringer og muligheter knyttet til brukbarhet som kan oppstå ved bruk av fri programvare i utviklingsprosjekter. Vi vil i denne seksjonen se på hvilke forskningsmetoder vi benytter for å nå dette målet og for å kunne svare på de underliggende forskningspørsmålene presentert i Seksjon 1.3.1.

### 3.3.1 Systemutviklingscase

For å besvare dette prosjektets problemstilling velger vi å gjennomføre et systemutviklingsprosjekt hvor vi går inn i forskningsprosjektet MOBESITY og bruker dette som case. I dette caset utvikler vi et system for egenomsorg blant overvektspasienter basert på åpen kildekode-komponenter. Utgangspunktet for utviklingen er høstens fordypningsprosjekt hvor krav og behov fra brukergruppen ble identifisert gjennom en workshop, nærmere forklart i Seksjon 2.1. Vi vil ikke benytte oss av en spesiell utviklingsmetodikk, men utvikle i iterasjoner med to satte brukbarhetstester som frister.

Som en del av systemutviklingscaset vil vi gjennomføre brukbarhetstesting for å avdekke brukbarhetsproblemer og få tilbakemeldinger til videreutvikling av systemet. Denne testingen vil også være en del av vår forskningsmetode og gi oss data til analyse og diskusjon.

Erfaringene vi høster fra deltagelsen i MOBESITY-prosjektet, selve utviklingsprosessen og metodene brukt underveis danner grunnlaget for å kunne

svare på forskningsspørsmålene. I tillegg vil vi basere oss på relevant litteratur som er lagt fram i Kapittel 2. Det er mulig å se på dette systemutviklingscaset både som en form for aksjonsforskning og et slags systemutviklings-eksperiment. Vi ønsker å undersøke et årsak-virking-forhold i form av hvilke brukbarhetsutfordringer som oppstår ved utvikling basert på fri programvare. Likevel er ikke graden av kontroll i undersøkelsen tilstede på den måten at det kan gjennomføres direkte og presise målinger av det som gjøres, slik som er vanlig i eksperimenter.

Sjøberg et al. [38] sier følgende om aksjonsforskning: «It attempts to provide practical value to the client organization while simultaneously contributing to the acquisition of new theoretical knowledge». Det er tydelig at vårt systemutviklingscase er en form for aksjonsforskning da vi utvikler et system som gir verdi for MOBESITY-prosjektet, samtidig som vi får innsikt og kunnskap om brukbarhetsutfordringer i forbindelse med et slikt prosjekt.

### 3.3.2 Brukbarhetstesting

Systemutviklingscaset har som mål å resultere i et ferdig system bestående i stor grad av åpen kildekode-komponenter og som realiserer flest mulig krav. Det ferdige systemet skal så brukbarhetstestes med representative brukere i brukbarhetslaboratoriet på NSEP. Målet med brukbarhetstestene er todelt. I forhold til systemutviklingscaset er det viktig å avdekke brukbarhetsproblemer og få input til redesign og videreutvikling av systemet. Samtidig er brukbarhetstesting en del av vår forskningsmetode, da dataene fra denne er en del av grunnlaget for å besvare oppgavens problemstilling.

Under brukbarhetstestene vil vi benytte oss av intervju og spørreskjemaer som datainnsamlingsmetoder. Det vil bli utført oppfølgingsintervjuer (semi-strukturert intervju) av testdeltagerne etter hver test og hver deltager må fylle ut SUS-skjema og generelt bakgrunnsskjema. I tillegg vil vi observere testene og samle inn relevante data på denne måten. Selve gjennomføringen av brukbarhetstesting inkludert datainnsamlingsmetodene er nærmere beskrevet i Kapittel 6.

### 3.3.3 Analyse av resultater

Etter å ha gjennomført to iterasjoner brukbarhetstester har vi avdekket brukbarhetsproblemer som kan analyseres og diskuteres med hensyn på forskningsspørsmål 1. Samtlige brukbarhetsproblemer vil bli kategorisert i forhold til



### 3.3. VALG AV METODE

---

relevante brukbarhetsheuristikker og gradert etter alvorlighet, mens enkelte problemer blir nærmere illustrert med skjermbilder fra systemet.

Når brukbarhetsproblemene er kategorisert vil vi se nærmere på hvilke problemer som kan skyldes bruk av åpen kildekode-komponenter og om problemene kan isoleres til enkeltkomponenter eller om de er et resultat av integrasjonen av flere komponenter. Vi vil ta i bruk etablerte brukbarhetsprinsipper og se på hvilke prinsipper som brytes mest i vårt system.



# Kapittel 4

## Utvikling

Dette kapitlet vil ta for seg utviklingsprosessen, med vektlegging på hva som er relatert til brukbarhet. Vi vil gjennomgå de viktigste milepælene i prosjektløpet, og dokumentere hva som ble gjort med tanke på å øke brukbarhet underveis.

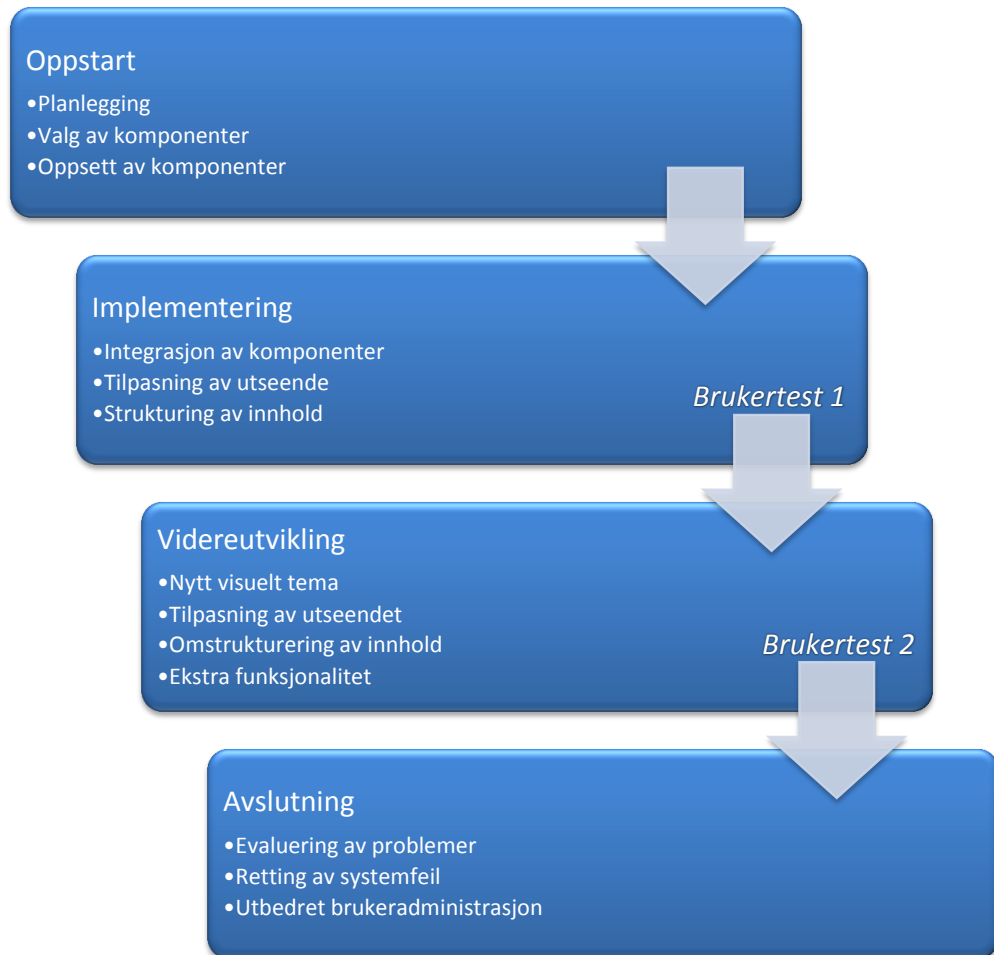
### 4.1 Tidsforløp

Utviklingen tok utgangspunkt i kravene fra Anita Das og MOBESITYprosjektet. I tillegg baserte vi oss på resultatene fra høstens fordypningsprosjekt, der vi diskuterte oss fram til hva pasientene og helsepersonellet ønsket seg fra et slikt system, samt hva som var realistisk oppnåelig å utvikle.

Vi utviklet i iterasjoner, men uten å tilpasse utviklingen til en konkret utviklingsmodell (som for eksempel Scrum eller vannfall). Gjennom møter med veileder og leder for MOBESITY-prosjektet nesten ukentlig fikk vi tilbakemeldinger på ferdigstilt funksjonalitet, og kunne deretter endre eksisterende eller legge til ny funksjonalitet. Figur 4.1 viser de viktigste utviklingsfasene for dette prosjektet.

Vi fikk raskt opp en tidlig versjon av en Wordpress-nettside med et forum, et to-fase-innloggingsskall og en kalender. Selv om alle disse komponentene var satt opp, hadde vi problemer med å integrere disse, spesielt med tanke på autentisering.

Etter at en tidlig versjon av de ulike komponentene var satt opp, så vi mulighet for å gjennomføre en brukbarhetstest i tillegg til den planlagte



**Figur 4.1:** De viktigste fasene i prosjektløpet

brukbarhetstesten med reelle brukere. På denne måten kunne vi avdekke de største problemene knyttet til grensesnittet på et tidlig stadium. Vi var klar over at det vi hadde utviklet på dette tidspunktet hverken var pent eller veldig brukervennlig, og vi hadde mange tanker om hva som kunne forbedres. Likevel ville vi gjennomføre en brukbarhetstest for å få ytterligere innspill.

Vi var også klar over at vi ikke kunne få to iterasjoner brukbarhetstesting med representative brukere, på grunn av problemer med å rekruttere såpass mange unike brukere. Derfor planla vi å ta kontakt med studievenner og andre som kunne være interessert. Alle som deltok hadde et høyt teknisk nivå, noe vi antok var ganske ulikt de faktiske framtidige brukerne. Dette var dog ikke nødvendigvis noe negativt. Vi håpte at slike kyndige brukere av data og

Internett kunne gjøre sitt for at vi fikk avdekket mange problemer som vi ikke ville avdekket selv. En kan si at det minnet litt om en brukbarhetstest med profesjonelle som evaluerte produktet, ikke så ulikt en heuristisk evaluering. Dette passet ypperlig inn i en tidlig fase av prosjektet.

Etter den første brukbarhetstesten satt vi igjen med en god del data. Vi hadde meninger om hvordan vi kunne forbedre nettsiden, basert på resultatene fra brukbarhetstest I, men også basert på våre egne idéer og sunn fornuft. Endringene og videreutviklingen som ble gjort i etterkant av brukbarhetstest I er presentert i Seksjon 4.4. De viktigste funnene fra testen nevnes også i denne seksjonen, mens en mer detaljert beskrivelse av problemer og observasjoner er presentert i Kapittel 7.

Endringene etter brukbarhetstest I resulterte i det endelige systemet som er presentert i Kapittel 5. Siden vi satte punktum for utvikling ved brukbarhetstest II ble det ikke gjort store endringer etter denne, og vi vil derfor ikke utdype disse i dette kapittelet. Resultatene fra brukbarhetstest II, blant annet problemer og observasjoner, er presentert i Kapittel 8.

Hva som ble gjort etter brukbarhetstest II er formidlet i Seksjon 4.5. Vi vil også presentere hvordan vi oppfattet komponentenes modifiserbarhet med tanke på brukbarhet. Til slutt presenterer vi hva som kan gjøres videre på nettsiden for å tilfredsstille flere behov fra brukergruppen(e).

## 4.2 Oppstart

For å få en oversikt over hva vi trengte av komponenter gjennomførte vi en idémyldring i starten av prosjektet. Resultatet av denne ble en liste over komponenter som var nødvendige for å tilfredsstille de viktigste kravene til systemet. Vi kom fram til at de følgende punktene ville dekke funksjonaliteten som var ønsket.

- Et publiseringsverktøy for å formidle informasjon til brukere.
- Et forum for kommunikasjon, deling av informasjon og erfaringer.
- En kalender for påminnelser om spising, trening, vitaminer o.l.
- Et meldingssystem for private meldinger mellom brukere.
- En elektronisk dagbok.

Det var ikke utarbeidet konkrete krav til systemets brukbarhet, men det var implisitt at det skulle ha et tilstrekkelig nivå av brukbarhet tilpasset den aktuelle brukergruppen. Hvordan systemet ville se ut grafisk var derfor opp til oss, og vi måtte bruke våre erfaringer fra liknende prosjekter og kunnskap om brukbarhet for å utforme løsningen.

Den viktigste modulen i systemet er publiseringsverktøyet<sup>1</sup>. Etter råd fra veileder og etter en samtale med Professor Maria Letizia Jaccheri valgte vi å ta i bruk Wordpress som vår CMS-løsning. Vi så nærmere på Wordpress, og syntes dette virket som et godt verktøy å ta i bruk.

Wordpress er kjent for god brukervennlighet og modifiserbarhet, samtidig som vi visste at det eksisterte en rekke temaer til Wordpress, som gjorde at vi kunne endre brukergrensesnittet relativt enkelt til det som passet best til nettsidens bruk. På et tidlig tidspunkt virket derfor Wordpress som et godt valg.

Vi lette etter kalendere som var passende for det vi trengte. Et av kriteriene var at kalenderen burde være skrevet i samme språk som Wordpress, nemlig PHP. Dette ville sannsynligvis gjøre integreringen med Wordpress noe enklere enn om den hadde vært skrevet i et annet språk. Samtidig ville det være lettere å modifisere utseende og funksjonalitet til å tilpasses brukerne og resten av systemet. Et annet kriterium var at kalenderen skulle være enkel, da det ikke var noen krav til avansert funksjonalitet fra brukernes side. Likevel hadde kalenderen mer funksjonalitet enn det som krevdes, og vi måtte gå inn i kildekoden og skjule funksjonalitet og valg på denne måten.

Vi var også på utkikk etter et kommunikasjonsverktøy som kunne ta seg av de ulike kommunikasjonsveiene mellom pasienter og helsepersonell. Opprinnelig falt valget på phpBB, som er en fullstendig internettforum-pakke med en mengde funksjonalitet tilgjengelig gjennom moduler som kan installeres etter behov. Vi fikk phpBB anbefalt av veileder, og i tillegg hadde et av gruppemedlemmene erfaring med å sette opp og administrere denne forum-pakken.

### 4.3 Implementasjon

Vi startet utarbeidningen av nettsiden med å installere Wordpress og legge til et veldig enkelt og minimalistisk tema. Deretter satte vi opp de andre

---

<sup>1</sup>Kanskje bedre uttrykt på engelsk ved Content Management System (CMS).

## 4.3. IMPLEMENTASJON



Figur 4.2: Utseendevalg i administrasjonspanelet, Simple:Press

komponentene og tilpasset deres utseende til Wordpress-temaet. Da vi skulle integrere phpBB og Wordpress viste det seg vanskelig å få til en sømløs integrasjon med single sign-on<sup>2</sup>. Før vi investerte for mye tid i å integrere phpBB undersøkte vi andre alternativer for kommunikasjon. Vi fant Simple:Press<sup>3</sup>, som er et foruminnstikk til Wordpress. Dette forumet hadde all funksjonalitet vi trengte og var enkel å integrerte i Wordpress. I tillegg tillot den oss å bruke den samme brukerdata-basen på tvers av komponentene.

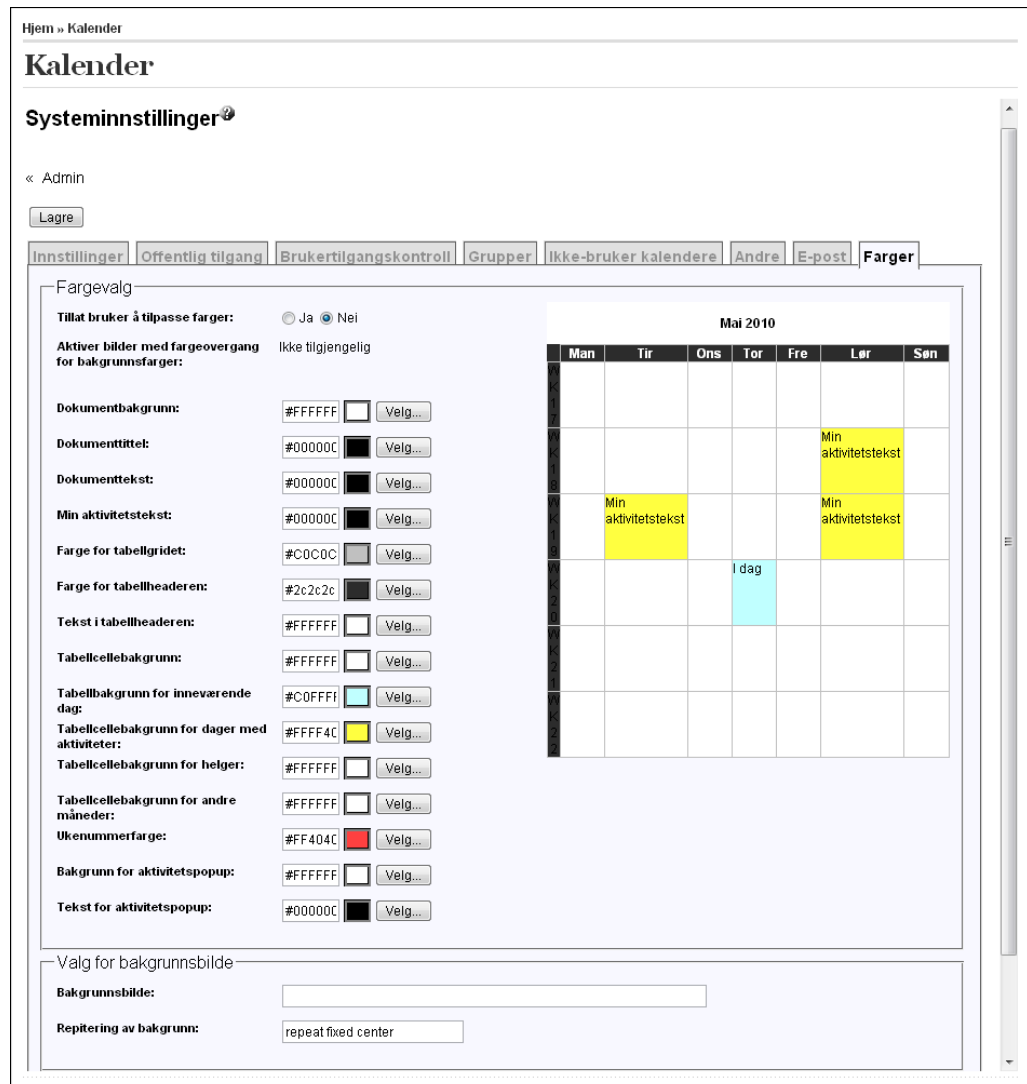
Under utseendetilpasningen av komponentene viste det seg at forumet var vanskeligst å endre layout på. Likevel var det visuelt sett ganske minimalistisk og nøytralt fra før, så det stred ikke med Wordpress-temaet på noen markant måte. Dessuten var det mulig å endre stil ved å modifisere eksisterende stil- og knappesett. Figur 4.2 viser administrasjonspanelet, med mulighet for å endre tema, knappeoppsett og hvilke ikoner som skal vise tilhørende titteltekst. Som en ser av figuren har vi laget et tema og et knappesett, begge navngitt «mobese».

Kalenderen hadde en egen side der administratorene kunne endre farger og utseende, vist i Figur 4.3. I tillegg var det mulig å endre tema på kalenderen,

<sup>2</sup>En universal innlogging som autentiserer deg i flere uavhengige systemer [52]

<sup>3</sup><http://simple-press.com/>

noe som egentlig var en enklere måte å endre alle farger og stiler på. Dog fantes det ytterst få tema å velge mellom.



Figur 4.3: Fargevalg i administrasjonspanelet, WebCalendar

Å integrere WebCalendar som en Wordpress-side viste seg utfordrende. Etter å ha undersøkt ulike løsninger ganske grundig, kom vi fram til at vi dessverre måtte ta i bruk HTML iframe<sup>4</sup>-elementet for å inkludere kalenderen på en side. Dette er generelt anbefalt å unngå, da det kan føre med seg en del strukturelle og utseendemessige utfordringer da en i prinsippet inkluderer en

<sup>4</sup>Definerer en ramme for visning av en nettside i en annen nettside.



nettside i en annen nettside.

Med de ulike komponentene og det visuelle temaet på plass gikk mye av arbeidet ut på å integrere komponentenes funksjonalitet og utseende sammen for å oppnå et helhetlig system. Parallelt med dette utviklet vi innloggingsfunksjonalitet. Dette ble lagt som et «skall» utenpå systemet, i tillegg til en side der brukeren kunne logge inn. Denne siden ble tilpasset slik at den matchet resten av nettsiden visuelt. Dette ble gjort for å gi brukeren et mer helhetlig inntrykk av systemet, slik at han hele tiden er sikker på at han befinner seg på samme nettside.

Da hoveddelen av de ulike komponentene var på plass, la vi inn den relevante informasjonen. Vi hadde ikke fått klare instruksjoner om hvordan informasjonen skulle struktureres, og gjorde dette noenlunde likt som det var gjort på overvektsklinikkens egne nettsider. Vi registrerte at denne strukturen var suboptimal, men vi valgte å ikke gjøre noe mer med dette da brukbarhetstest I var rett rundt hjørnet, og vi antok at eventuelle problemer med informasjonstrukturen ville bli avdekket under denne testen.

## 4.4 Videreutvikling

Denne seksjonen oppsummerer videreutviklingen og endringene som ble gjort i systemets ulike deler i etterkant av den første brukbarhetstesten. Disse endringene omfatter både utseende, funksjonalitet og informasjonsstrukturen på nettsiden og ble gjort både på bakgrunn av tilbakemeldinger vi fikk under testen og vår egen oppfatning av siden. De viktigste funnene fra brukbarhetstesten som dannet grunnlaget for endringene er nevnt her, mens en detaljert beskrivelse av resultatene fra testen kan sees i Kapittel 7.

Resultatet av endringene og utviklingen vi gjorde etter brukbarhetstest I er presentert som den endelige løsningen i Kapittel 5.

### 4.4.1 Visuelt tema

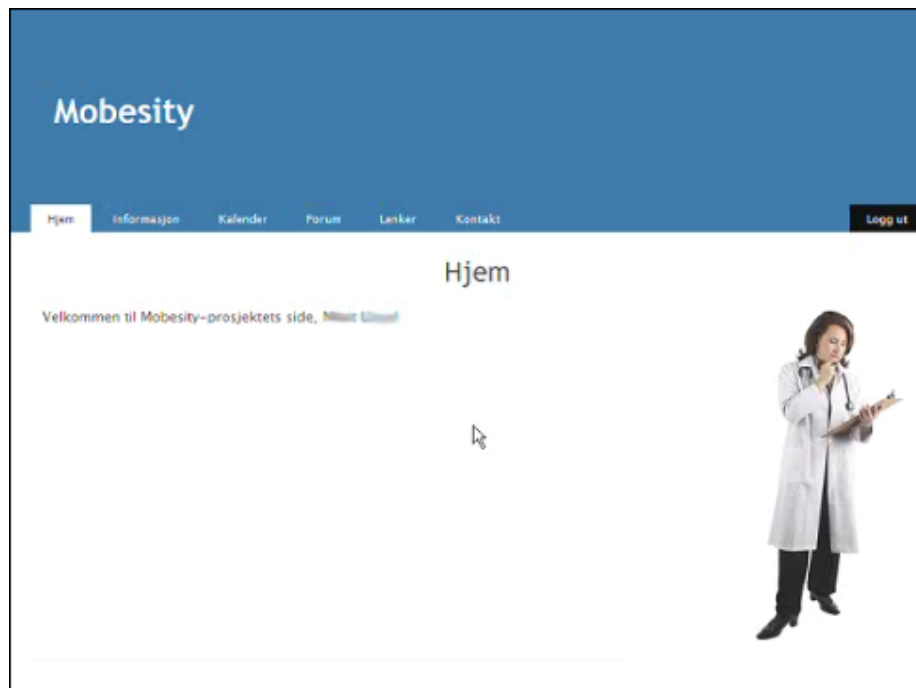
Flere av testpersonene under brukertesten påpekte at utseendet på nettsiden var «kjedelig». Vi var heller ikke fornøyd med utseendet og sa oss enige i at brukergrensesnittet var i overkant minimalistisk og innholdsløst.

Vi bestemte oss derfor relativt raskt etter brukbarhetstest I for å installere et nytt visuelt tema. Valget falt på Arthemia-temaet [53] til Wordpress. Vi tok

utgangspunkt i standard fargeoppsettet for temaet og tilpasset blant annet kalenderens utseende til dette. I tillegg til å endre temaet for selve nettsiden endret vi også utseende på innloggings-siden slik at det «matchet» det nye temaet.

#### 4.4.2 Hjem-siden

Under den første brukertesten bestod hjem-siden kun av et enkelt bilde og en velkomsttekst, som vist i Figur 4.4<sup>5</sup>. Dette gjorde at denne siden ble oppfattet som kjedelig og nærmest unødvendig av enkelte deltagere i brukertesten. En av testdeltagerne foreslo å gjøre denne siden mer personlig, for eksempel ved å vise brukerens nye meldinger og kommende aktiviteter som en del av innholdet. Vi bestemte oss for å følge opp dette forslaget og gjøre hjem-siden mer informativ og personlig.

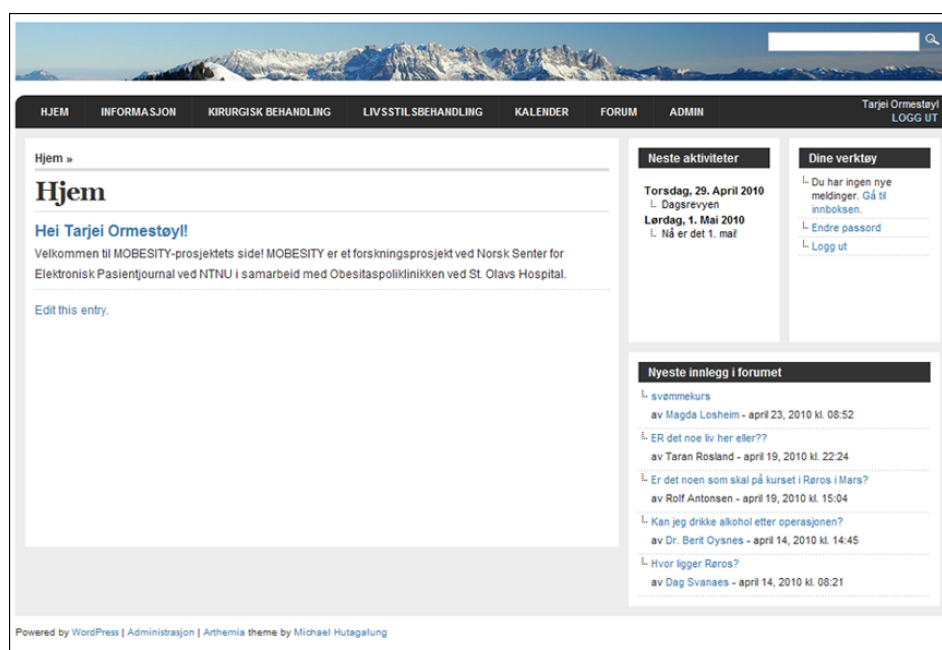


**Figur 4.4:** Hjem-siden i versjon 1

Det ble lagt til tre bokser på framsiden med informasjon fra kalender og forum. En av boksene viser en liste med de neste kommende aktivitetene

<sup>5</sup>Forfatterne tar selvkritikk på bildekvaliteten til skjermdumpene fra versjon 1 av systemet.

## 4.4. VIDEREUTVIKLING



Figur 4.5: Hjem-siden i versjon 2

i den innloggede brukerens kalender. En annen boks signaliserer om den innloggede brukeren har mottatt nye personlige meldinger via forumsystemet og inneholder valg for endre passord og logge ut av nettsiden. Den siste boksen viser en liste over de nyeste innleggene i forumet som brukeren har tilgang til. Figur 4.5 viser en skjermdump av hjem-siden etter at endringene ble gjort.

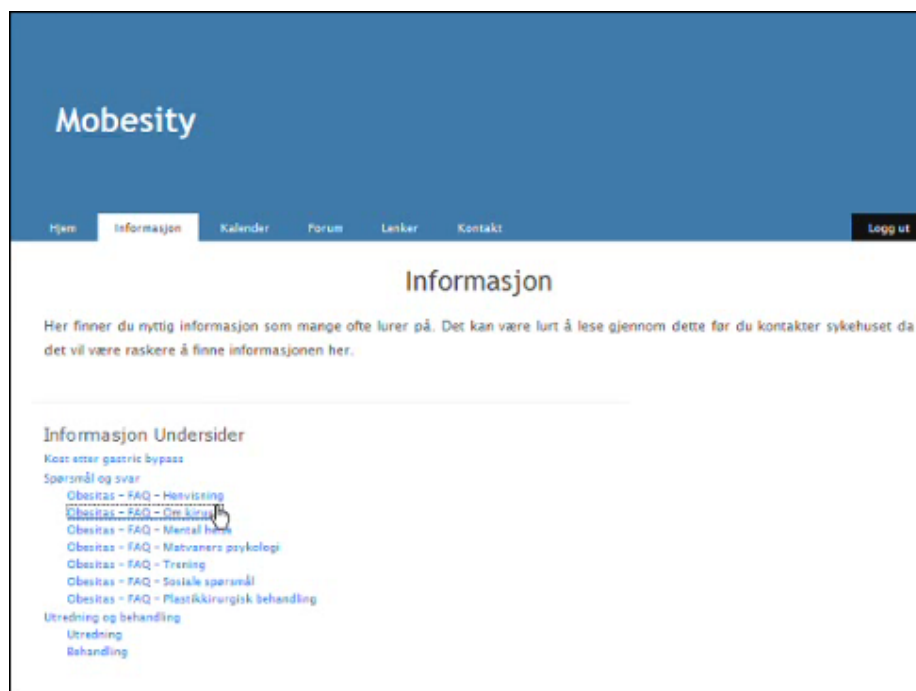
### 4.4.3 Informasjon

Informasjonsdelen av nettsiden var der testdeltagerne hadde flest problemer under den første brukbarhetstesten. Innholdet her var fordelt på lange dokument med mye tekst, og titlene på dokumentene var ikke veldig gode, noe flere deltagere kommenterte. Vi hadde heller ikke lagt til rette for noen form for søkefunksjonalitet, så testpersonene måtte lete gjennom lange dokumenter for å finne det de lette etter. Informasjons-hovedsiden er vist i Figur 4.6.

Vi bestemte oss for å gå bort i fra én enkel side med all informasjon og heller samle dette i hovedkategorier, som for eksempel «Operasjon», «Kosthold» og lignende. Videre ønsket vi å unngå lange sider med tekst og heller dele opp innholdet over flere undersider. I tillegg måtte det komme tydeligere fram at innholdet på informasjonssidene kom fra klinikken og ikke var lagt ut av

pasientbrukere på nettsiden, noe enkelte testdeltagere misforstod.

En av fordelene ved å ta i bruk det nye Wordpress-temaet var muligheten for nedtrekksmenyer i navigasjonsfeltet på siden. Dette førte til at inndelingen av informasjon på sider og undersider ble mer integrert i designet på siden og brukerne kunne lettere få oversikt over hva slags informasjon som fantes på sidene ved å holde musa over navigasjonsmenyen. Et eksempel på hvordan informasjonssidene så ut etter endringene er vist i Figur 4.7.

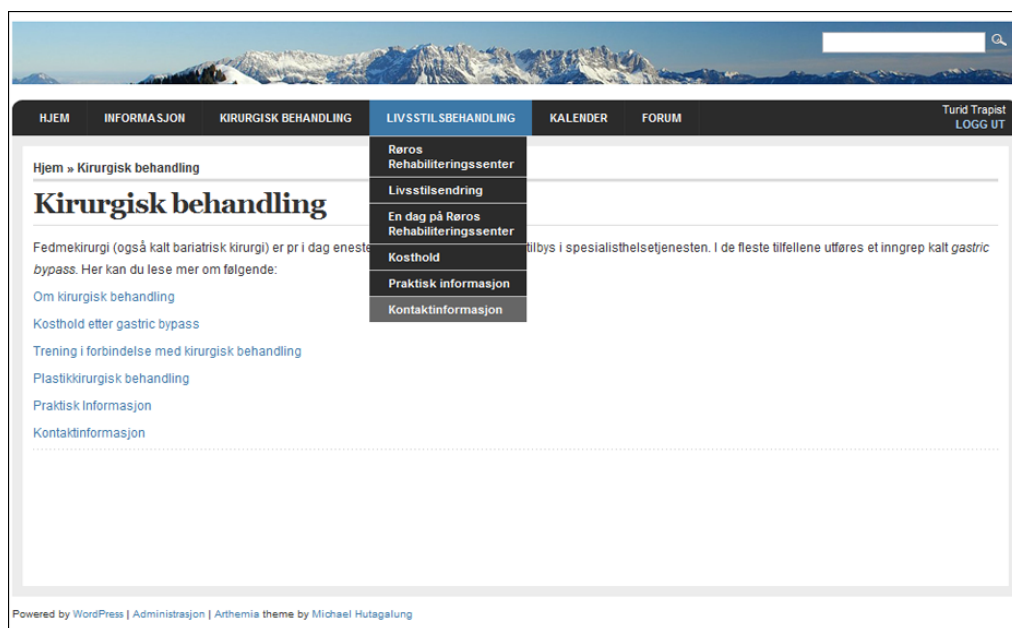


**Figur 4.6:** Informasjons-siden i versjon 1

Det er i hovedsak tre av nedtrekksmenyene i navigasjonsfeltet som inneholder informasjon fra overvektsklinikken. Den første, kalt «informasjon», inneholder generell informasjon om utredningsløp og henvisning i forbindelse med de to behandlingsformene (kirurgi og livsstilskurs). I tillegg finnes artikler om matvaner og mental helse samt ofte stilte spørsmål om sosiale hensyn og en liste med begreps- og ordforklaringer.

De to andre nedtrekksmenyene inneholder informasjon om de to behandlingsformene for sykelig overvekt; kirurgisk behandling og livsstilsbehandling. Når det gjelder kirurgisk behandling kan pasientene blant annet finne informasjon om selve inngrepet og tips om trening og kosthold i tiden etter operasjon. I

## 4.4. VIDEREUTVIKLING



Figur 4.7: Informasjonssidene i versjon 2

tillegg finnes annen praktisk informasjon og kontaktinfo til overvektsklinikken.

Livsstilsbehandlingen finner sted på Røros Rehabiliteringssenter og informasjon om senteret og opplegget der finnes i den siste nedtrekksmenyen.

Det ble også lagt til et søkefelt i «headeren» på siden hvor brukerne kan søke i innholdet på nettsiden (forum og kalender har egen separat søkefunksjonalitet).

### 4.4.4 Forum

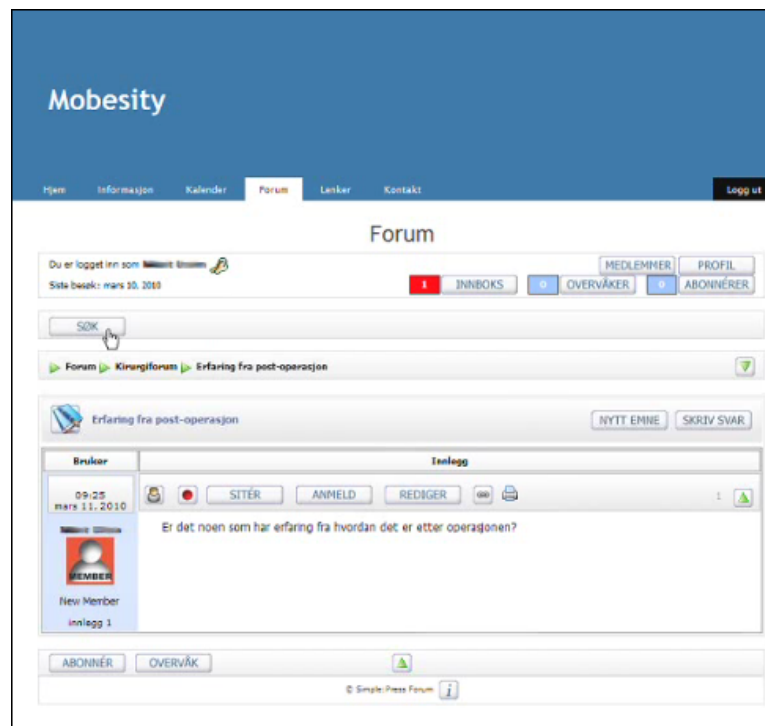
Forumet virket forståelig for deltagerne under den første brukbarhetstesten og det var få problemer knyttet til dette. Likevel var det enkelte deler av forumet som hadde forbedringspotensiale.

Vi fikk enkelte kommentarer på at forumet inneholdt for mange forvirrende knapper og valg, som for eksempel «overvåke» og «abonnér» som brukerne flest ikke følte behov for. Et eksempel på dette er vist i Figur 4.8. Det var også noe inkonsistens når det gjaldt den norske oversettelsen av forumet vi baserte oss på.

På bakgrunn av dette fjernet vi enkelte knapper og valg som ikke de aktuelle

brukerne hadde behov for eller som vi følte skapte unødig forvirring under bruk. Blant annet gjaldt dette mulighetene for å overvåke og abonnere på tråder og innlegg. Vi ryddet også opp i den norske oversettelsen som vi tok i bruk. En skjermdump av det nye forumutseendet kan sees i Figur 4.9.

Det ble også foretatt en oppgradering av forum-kodebasen hvor funksjonalitet for å «tagge» tråder og rangere innlegg ved å gi tommel opp eller ned ble lagt til. Vi følte tagging av tråder kunne bidra til å kategorisere all informasjonen som fantes i forumet bedre, mens rangering av innlegg var etterspurt funksjonalitet fra brukerne for å «kvalitetssikre» brukernes foruminnlegg.



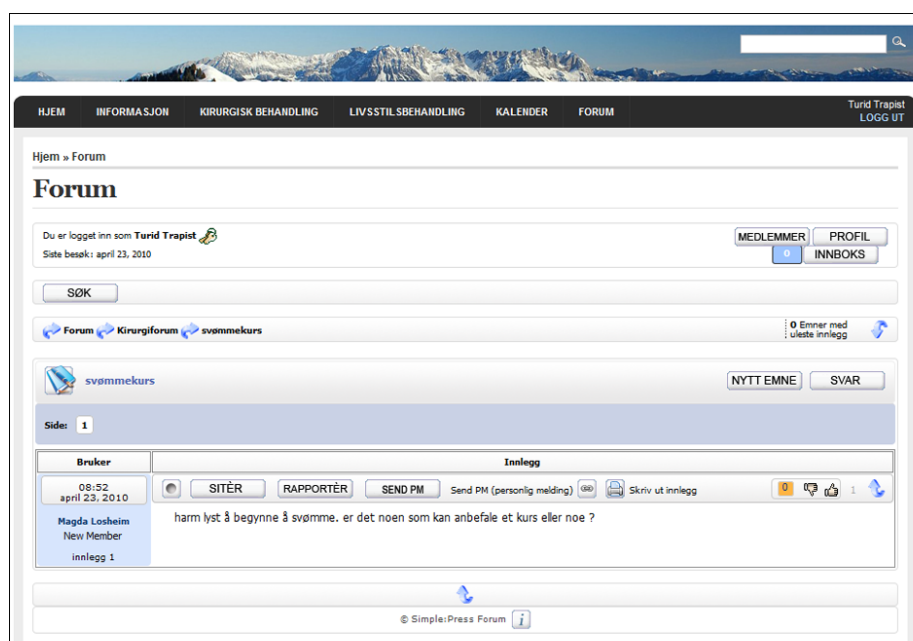
Figur 4.8: Forumet i versjon 1

#### 4.4.5 Kalender

Utseendet til kalenderen under den første brukbarhetstesen kan sees i Figur 4.10. Kalenderen var relativt uferdig rent utseendemessig da den ble testet og vi identifiserte flere områder som kunne forbedres.

Flere av testpersonene hadde problemer med å finne årsvisningen og dette valget måtte åpenbart gjøres tydeligere. Vi fikk også kommentarer på at

## 4.4. VIDEREUTVIKLING



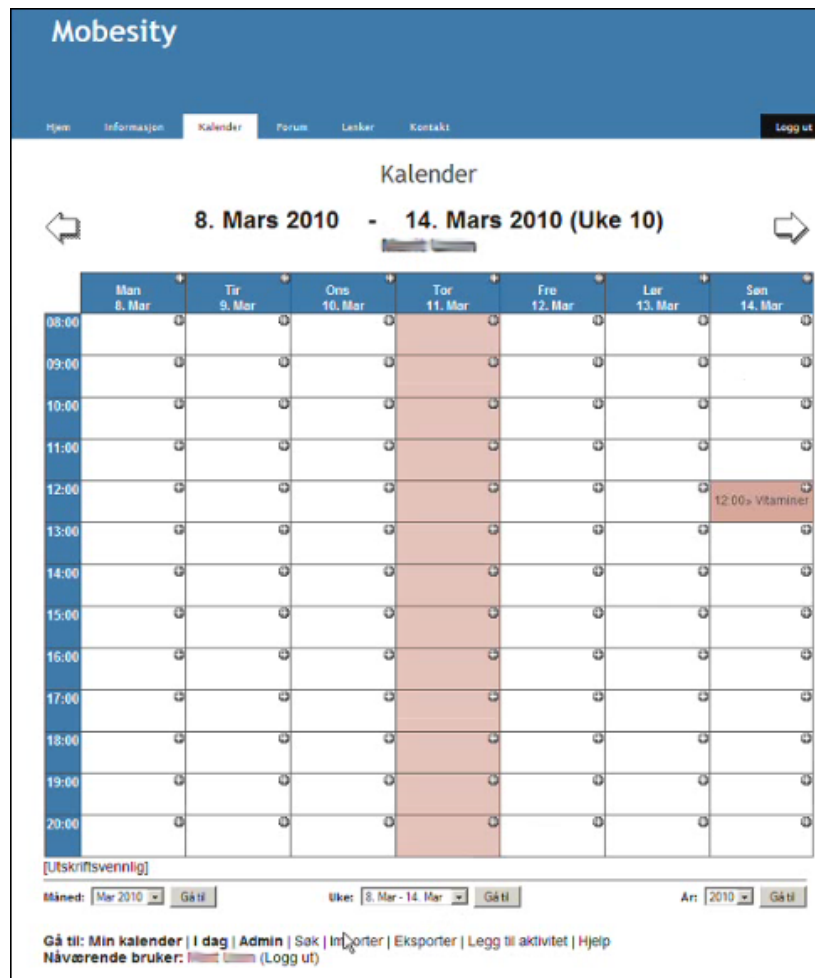
Figur 4.9: Forumet i versjon 2

fargebruken på dagens dato og aktuelle aktiviteter var for lik og at dager med aktiviteter var lite synlige i årsvisningen. Vi så i tillegg et behov for å fjerne en del av valgene i menyen under kalendervisningen som føltes lite relevante for den aktuelle brukerguppen.

Som nevnt i Seksjon 4.4.1 startet vi med å tilpasse kalenderens utseende til det nye visuelle temaet. Videre ble teksten på knappene for å vise uke, måned og år forandret fra «Gå til» til «Vis». I tillegg ble det fjernet en rekke unødvendige valg fra menyen nederst i kalenderen. Dette var valg som ikke var godt nok integrert i nettsiden i sin nåværende form eller funksjoner som ikke var nødvendige for de aktuelle brukerne. Figur 4.11 viser kalenderforsiden etter endringene ble gjennomført.

### 4.4.6 Elektronisk dagbok

Det var ønskelig for brukerne å kunne føre diverse dagbøker i systemet, for eksempel treningsdagbok, måltidsdagbok og vanlig dagbok, som igjen kunne deles med helsepersonell. Vi fant ingen åpen kildekode-komponenter som tilfredsstilte disse kravene og denne funksjonaliteten måtte derfor implementeres fra grunnen av.



Figur 4.10: Kalenderen i versjon 1

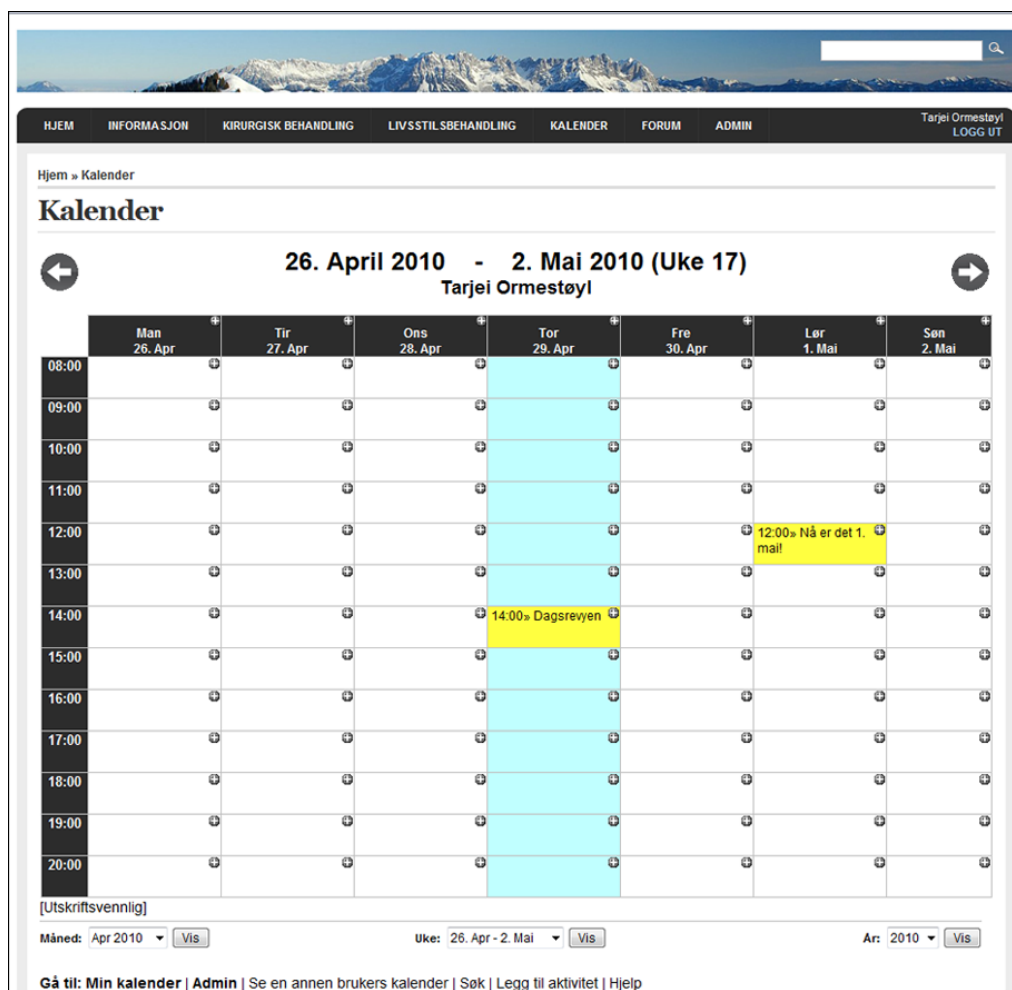
Vi skisserte en løsning på dette, men ressurskapasiteten førte til at vi måtte prioritere andre moduler, og heller forsøke å heve kvaliteten på disse. Dagbok-funksjonaliteten kom derfor ikke med i det endelige systemet til brukbarhets-test II.

## 4.5 Avslutning og evaluering

Etter at den andre brukbarhetstesten var avsluttet gikk vi gjennom problemene som ble identifisert og rettet opp systemfeil samt de største brukbarhets-problemene som oppsto på grunn av systemfeil. Vi implementerte også et



## 4.5. AVSLUTNING OG EVALUERING



Figur 4.11: Kalenderen i versjon 2

enkelt grensesnitt for å administrere brukere i systemet. Her kunne administrator legge til og slette brukere, og man har mulighet til å endre sitt eget passord. Vi ryddet også opp i databasen og fjernet testdata som ble brukt under testingen. Til slutt ble det utført noe vedlikehold på autorisering og autentisering av brukere, da den eksisterende løsningen inneholdt noen sikkerhetshull.

### 4.5.1 Modifiserbarhet med tanke på brukbarhet

I denne seksjonen har vi vurdert kodekvaliteten til de ulike komponentene i forhold til hvor enkelt det var å utbedre brukbarheten. Tabell 4.1 viser vår tolkning av de ulike komponentenes kodekvalitet og tilhørende kvalitetsattributter. Det er viktig å understreke at vi ikke foretok noen grundig og formell undersøkelse av dette, men kun baserer oss på vår egen erfaring fra de delene av kildekoden vi studerte og modifiserte.

En del av arbeidet med å redusere brukbarhetsproblemer bestod i å gå inn i komponentenes kodebase for å endre eller legge til hva som skulle vises. For eksempel gikk vi inn i kildekoden til Wordpress for å legge til en logg utknapp øverst til høyre i navigasjonsmenyen. Også i kalenderen modifiserte vi koden for skjule noen menyvalg som ikke var av nytte, og annen unødvendig funksjonalitet. Vi hadde også behov for å endre koden til forumkomponenten for å rette opp i noen problemer med den eksisterende koden. Vi så tydelig at de ulike komponentene hadde helt ulik kodestruktur, og det var tydelig forskjell på hvor enkelt det var å modifisere disse.

En ser av tabellen at vi var mest fornøyd med kvaliteten på Wordpress og tilhørende kodebase. Vi opplevde at den var relativt enkel å foreta endringer på, og hadde også få problemer relatert til dette. Wordpress har et veldig stort community med flere tusen aktive brukere, som medfører fyldig dokumentasjon om kjente problemer, modifikasjoner og annen informasjon.

<b>Kvalitetsattributt</b>	<i>Wordpress</i>	<i>Simple:Press</i>	<i>WebCalendar</i>
Forståelighet	God	Nokså god	Mangelfull
Grad av dokumentasjon	God	OK	Mangelfull
Community og support	Bra	Bra	OK
Kodestruktur	God	Nokså God	Dårlig
Modifiserbarhet	Enkel	Vanskelig	Vanskelig
Problemer/feil	Ingen	Noen	Få
Gjennomført	Ja	Ja	Noe uferdig

**Tabell 4.1:** Kodekvaliteten til komponentene

Simple:Press var også tilfredstillende å arbeide med. Det som trakk ned var at det tok litt lengre tid å forstå hvordan alt fungerte på grunn av noe mangelfull dokumentasjon, og litt komplisert kodestruktur. Vi hadde også noen problemer med forumet og forumets kobling til databasen. Vi måtte her gå inn i koden for å legge til ekstra databasetilkoblinger som ble lukket andre steder i koden. Forumets kodestruktur gjorde det vanskelig å gjøre endringer

i forhold til utseende. Vi var i stor grad bundet til å bruke administrasjons-panelet for å modifisere forumets utseende.

WebCalendar var den svakeste komponenten vi benyttet oss av når det gjaldt kodekvalitet. Komponentene var problematiske å integrere og vi oppfattet kildekoden som rotete. Enkelte filer og metoder var på mange hundre linjer og det var utfordrende å få oversikt nok til å modifisere noe uten at det fikk uforutsette ringvirkninger. Det fantes heller ikke dokumentasjon relatert til å gjøre direkte endringer i kildekoden. Vi fant også en del gjenstående arbeid i de ulike filene og metoder som ikke var fullstendig implementert. Det meste av utseendeforandringer ble derfor gjort gjennom administrasjonsmenyen som tilbød funksjonalitet for å gjøre dette.

### 4.5.2 Videre arbeid

Et naturlig neste steg i utviklingsprosessen vil være å forbedre nettsiden basert på problemene som ble funnet under brukbarhetstest I og II. Basert på resultatene fra brukbarhetstestene, fremstilt i Kapittel 7 og 8 har vi følgende forslag til forbedringer og endringer av nettsiden:

- Vurdere om informasjon og innhold kan struktureres bedre.
- Innføre et tydeligere skille mellom hva som er privat og hva som er åpent, spesielt bør meldingsfunksjonaliteten skilles ut.
- Det bør gjøres klarere hvordan en bruker påminnes i kalenderen.
- Kalenderen bør integreres bedre med Wordpress, og vi anbefaler å undersøke alternativer til iframe.
- I tillegg er det ønskelig at mindre, kosmetiske problemer med brukergrensesnittet utbedres.

Det eksisterer et ønske om at pasientbrukerne av systemet skal ha en egen dagbok der de kan notere ned relevant informasjon om trening og kosthold, samt oppføringer av mer personlig karakter. Enkelte av disse oppføringene må også helsepersonellet ha tilgang til. Selv om vi skisserte en mulig løsning for dette ville det vært bedre å finne en ferdigutviklet komponent som enkelt kan integreres i Wordpress. Dette har høyest prioritet av funksjonaliteten som ikke ble implementert i løpet av utviklingsløpet.

Det vil også være hensiktsmessig å se på en mulig sammenslåing av kalender og dagbok, da en del av dette hører naturlig sammen. For eksempel kunne det vært mulig for brukeren å lagre ulike typer oppføringer i kalenderen,

av typen personlig, trening eller kosthold. I tillegg bør kalenderen fortsatt beholde den nåværende støtten for å legge inn aktiviteter med mulighet for påminnelse via SMS. På denne måten kan kalenderen fungere som en personlig blogg/logg/dagbok i tillegg til å oppfylle sin rolle som kalender.

Til slutt bør de ulike sikkerhetsmessige aspektene utbedres og evalueres videre, da dette er utenfor denne oppgavens omfang.

# Kapittel 5

## Endelig løsning

I dette kapitlet vil vi presentere hvordan den endelige løsningen er satt sammen, og hvilke komponenter den består av. Det er viktig å merke seg at løsningen presentert i dette kapitlet er den siste versjonen, altså etter to iterasjoner med brukertesting. Vi har valgt å ikke presentere tidligere versjoner utover det som er gjort i Kapittel 4. Hva som ble endret underveis kommer også fram i nevnte kapittel.

### 5.1 Introduksjon

Som beskrevet i Seksjon 2.1 ble det gjennomført et forprosjekt hvor det ble laget en veldig enkel prototype på systemet. Den reviderte kravspesifikasjonen, tilbakemeldinger fra brukere, og våre erfaringer fra dette prosjektet var grunnlaget for å starte utviklingen av dette systemet.

Vi ønsket å se på utfordringene knyttet til brukbarhet ved utvikling av et system basert på åpen kildekode-komponenter. MOBESITY-prosjektet var passende som et systemutviklingscase for å se nærmere på disse utfordringene. Det var selvfølgelig også ønskelig å tilfredsstillte så mange krav som mulig i systemet på en tilstrekkelig måte. For Anita Das, doktorgradstudenten bak MOBESITY-prosjektet, var det vesentlig at implementasjonen av systemet nådde et nivå som gjorde det mulig å teste på et større antall pasienter i realistisk bruk. Dersom vi skulle ha utviklet systemet fra grunnen av, ville vi sannsynligvis ikke kunnet oppfylle nok krav til at det ville vært brukbart til dette formålet.

Prosjektmedlemmene hadde problemstillinger knyttet til bruk av åpen kildekode i utviklingsprosjekter. Det var derfor ønskelig å ta i bruk hovedsaklig åpen kildekode-komponenter i utviklingen av systemet, og selv kun utvikle det som var helt nødvendig. Alle komponenter måtte selvfølgelig integreres på tilfredsstillende måte.

Mastergradsstudent Ørjan Johansen er en del av samme utviklingsprosjekt, men med et annet fokus. På grunn av dette vil en del av det som blir presentert i dette kapitlet være tilsvarende i hans hovedoppgave.

## 5.2 Publiseringverktøy: Wordpress 2.9.2

Denne modulen fungerer som selve plattformen for nettsiden, og de andre komponentene er integrert inn i denne. Modulen håndterer selve innholdsstrukturen på nettsiden og all informasjonsbehandling. Wordpress benytter seg av brukere med ulike roller samtidig som det eksisterer et utall innstikk («plugins») som vi har tatt i bruk og i noen tilfeller modifisert til å passe systemets behov.

Vi valgte å bruke Wordpress<sup>1</sup> versjon 2.9.2. Neste generasjon Wordpress (3.0) er like om hjørnet, og aller helst ville vi nok ta i bruk denne. Men siden denne versjonen var kun tilgjengelig i beta på dette tidspunktet, valgte vi å bruke den stabile 2.9 versjonen. Wordpress gis ut under GNU GPL-lisensen.

I ettertid kan det virke som om Wordpress ikke var det beste valget av publiseringverktøy, og selv om vi fikk det anbefalt så burde en grundigere undersøkelse vært gjort. Det viste seg vanskelig å tilpasse visse aspekter ved Wordpress for å tilfredsstille kravene stilt til systemet. På et tidspunkt var det allerede lagt ned veldig mye tid i Wordpress og det var ikke lenger hensiktsmessig å bytte publiseringverktøy.

Blant andre CMS-verktøy som er utbredt finnes Drupal og Joomla. Litt undersøkelser gjort i ettertid viser at av de tre nevnte systeme er Drupal kjent for å være enklest å tilpasse til egne behov. Drupal tilbyr også en bedre løsning ved for å støtte flere brukere der det er behov for brukerspesifikt innhold. På en annen side går Drupal for å være minst brukervennlig å administrere for brukere som ikke er teknisk kyndige. Det som taler for Wordpress er at det i utgangspunktet er veldig enkelt å sette opp samtidig som det er enkelt å forholde seg til for nye brukere. For oss som utviklere hadde det vært en fordel

---

<sup>1</sup><http://wordpress.org/>

### 5.3. KOMMUNIKASJONSVERKTØY: SIMPLE:PRESS 4.2.2

---

med et publiseringsystem som var enklest mulig å tilpasse, men samtidig er det personer som ikke er spesielt teknisk kyndige av seg som skal administrere systemet og da er det en fordel at det er intuitivt og lett å forstå.

#### 5.2.1 Innstikk og temaer

Wordpress tilbyr en stor mengde innstikk og temaer, hovedsaklig gjennom tredjeparter, og de fleste av disse er gratis å installere og ta i bruk. For å integrere de ulike komponentene bedre og for å få et bedre grafisk brukergrensesnitt tok vi i bruk noen av disse.

**auto login** Et selvlaget innstikk som er en metode for å autentisere brukeren i wordpress gjennom vår logg inn-funksjonalitet.

**Exec-PHP** <sup>2</sup> Et innstikk som gjør det mulig å kjøre PHP-kode i sider, widgets og poster.

**Force SSL** <sup>3</sup> Et innstikk som krever at all kommunikasjon går over https-protokollen for å øke sikkerheten.

**Simple:Press** Foruminnstikket, presentert i Seksjon 5.3.

**WP Security Scan** <sup>4</sup> Et innstikk for å søke etter sikkerhetshull på nettsiden.

I tillegg til innstikkene benytter vi oss av temaet Arthemia<sup>5</sup> for Wordpress. Temaet tilbyr et pent utseende kombinert med hendig widget-behandling og god navigering, og var et av de mest passende temaene vi kom over.

## 5.3 Kommunikasjonsverktøy: Simple:Press 4.2.2

Det var ønske om et kommunikasjonsverktøy som hadde støtte for pasient til pasient og pasient til helsepersonell-kommunikasjon, slik som et standard forum med ulike roller støtter. Etter noe undersøkelser, samt prøving og feiling, besluttet vi å benytte oss av Simple:Press-forumet.

---

<sup>2</sup><http://bluesome.net/post/2005/08/18/50/>

<sup>3</sup><http://wordpress.org/extend/plugins/force-ssl/>

<sup>4</sup><http://semperfiwebdesign.com/plugins/wp-security-scan/>

<sup>5</sup><http://michaelhutagalung.com/2008/05/arthemia-magazine-blog-wordpress-theme-released/>

Simple:Press har standard forumfunksjonalitet, med mulighet for flere forum og brukergrupper. Simple:Press støtter også private meldinger mellom forum-medlemmene. På denne måten tilfredsstilte vi flere krav gjennom å ta i bruk modulen, og vi slapp å involvere flere eksterne komponenter for å støtte kommunikasjonsveiene.

Simple:Press har gode muligheter for å endre hva slags funksjonalitet som skal være tilgjengelig for brukeren, selv om det er få muligheter til store utseendemessige endringer. Forumet har dog et administratorgrensesnitt som lar oss skreddersy mye til vårt behov. Det er også trivielt å integrere Wordpress-brukerne med forumet, siden forumet er et innstikk til Wordpress og benytter seg av samme brukerdatabase.

## 5.4 Kalenderverktøy: WebCalendar 1.2.0

Når det kom til valg av kalenderfunksjonalitet evaluerte vi de kalenderne vi kom over med tanke på støttet funksjonalitet, sikkerhet og brukervennlighet. Vi kom over et relativt godt alternativ, nemlig WebCalendar, som gis ut under GNU GPL-lisensen.

WebCalendar<sup>6</sup> er et åpen kildekode-prosjekt skrevet i PHP som tilbyr standard kalenderfunksjonalitet, samt en del andre funksjoner som vi fant nyttige. Det finnes blant annet støtte for flere brukere, norsk språk, påminnelser, ulik kalendervisning og egen autentisering [54]. Det er også mulighet for å sende ut kalenderpåminnelser på e-post, og vi så her en mulighet for å modifisere dette til å sende ut SMS-påminnelser, noe vi også implementerte.

WebCalendar er fortsatt under utvikling, og benytter seg av sourceforge.net som vert for utvikling av åpen kildekode-prosjektet. Siden til WebCalendar finnes på <http://sourceforge.net/projects/webcalendar/>, og her kan en blant annet laste ned kalenderen og all kildekode, bli med å utvikle og få hjelp av utviklerne.

## 5.5 Universal innlogging

Vi trengte en komponent som på en sømløs måte integrerte brukerhåndtering og innlogging på tvers av de ulike komponentene. I utgangspunktet tilbød

---

<sup>6</sup><http://www.k5n.us/webcalendar.php>



Wordpress en god måte å gjøre dette på, men på grunn av kravene til sikkerhet var ikke dette tilstrekkelig.

Vi stod ovenfor et par problemer som ikke lot seg løse ved hjelp av Wordpress. For det første måtte vi implementere en to-fase innlogging som benyttet SMS, noe vi ikke fant støtte for blant Wordpress-innstikk. For det andre var det krav om sterkere sikkerhet enn det Wordpress kunne tilby, og vi trengte derfor en egen database til å lagre ulike data i forbindelse med dette. For det tredje hadde vi behov for å spesifisere brukergrupper utover det som tilbys i Wordpress.

Vi valgte følgelig å utvikle egen funksjonalitet for å håndtere disse kravene. Dette var tidkrevende, da vi måtte også integrere denne funksjonaliteten på en god måte i de ulike åpen kildekode-komponentene. Dette inkluderte å opprette brukere i alle moduler, oppdatere brukere i alle moduler, autentisere og autorisere brukere i alle moduler og til slutt å slette brukere i alle moduler dersom ønskelig.

Vi endte til slutt opp med en tilfredsstillende løsning som autentiserte brukeren i alle komponenter gjennom en to-fase-innlogging ved bruk av personlig passord og SMS-kode. Ved innlogging blir brukeren gitt de riktige rettighetene i systemet i forhold til rolle.

### 5.5.1 SMS-utsending

SMS-utsendingen vi bruker benytter seg av Telenor sitt PATS-rammeverk<sup>7</sup>. IDI har en avtale med Telenor om fri bruk av denne løsningen i forskningsøyemed og det egnet seg derfor godt til vårt formål. Rammeverket tilbyr et SOAP<sup>8</sup>-grensesnitt for utsending av SMS som vi kunne bruke både i vår universale innlogging og til kalenderpåminnelser.

## 5.6 Grafisk grensesnitt

I denne seksjonen vil vi presentere utseendet på nettsiden. Vi bruker noen skjermdumper for å forklare hvordan nettsiden er strukturert og for å forklare det fundamentale designet. Vi vil derimot ikke diskutere kvaliteten på designet her, men heller komme tilbake til dette i Kapittel 9 og 10.

---

<sup>7</sup><http://www.pats.no/?q=node/64>

<sup>8</sup>Protokoll for utveksling av XML-baserte meldinger i datanettverk [55]

### 5.6.1 Logg inn-skjermene

Figur 5.1 og Figur 5.2 viser logg inn-skjermene. Disse er ganske enkle og bør være intuitive for de fleste.



The screenshot shows a login form titled "Logg inn steg 1 av 2". It contains two input fields: "Brukernavn:" and "Passord:". Below the fields is a button labeled "Neste steg".

Figur 5.1: Logg inn steg 1



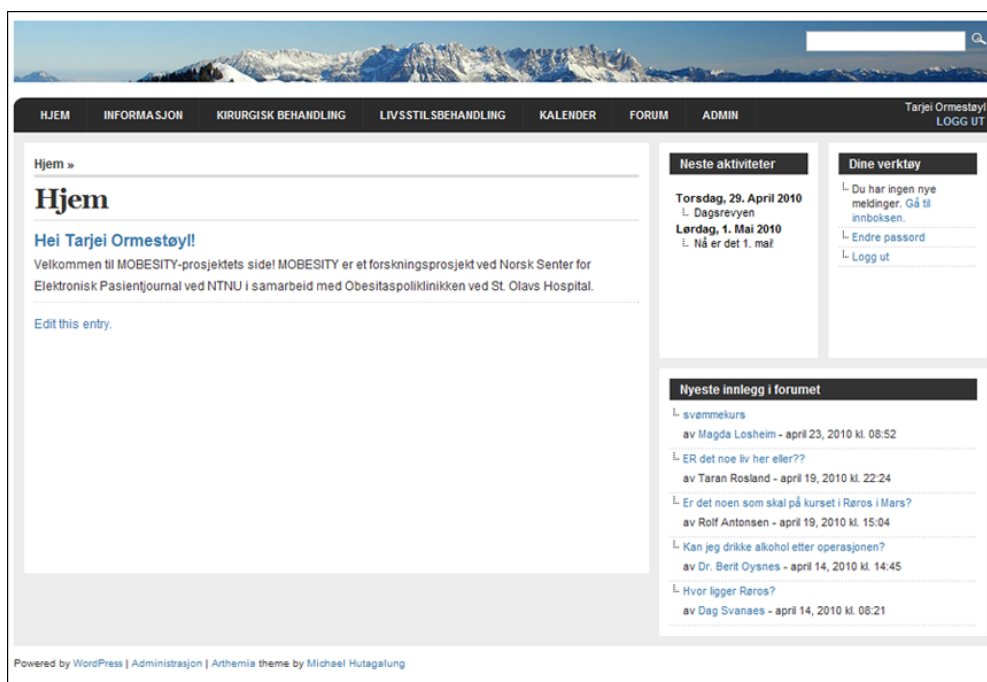
The screenshot shows a login form titled "Logg inn steg 2 av 2". It contains a message: "Vi har sendt en SMS-kode til nummeret du er registrert med. Skriv inn koden for å fortsette." Below the message is an input field labeled "Mottatt SMS-kode:". At the bottom is a button labeled "Logg inn".

Figur 5.2: Logg inn steg 2

### 5.6.2 Hjem-siden

Etter at brukeren har logget inn kommer han til Hjem-siden, vist ved Figur 5.3. Dette er en startportal for å bruke nettsiden, og brukeren har snarveier til de viktigste funksjonene via denne siden. Til høyre ser en de nyeste foru-minnleggene, oppkommende aktiviteter i kalenderen, eventuelle nye private meldinger, og mulighet for å endre passord.

## 5.6. GRAFISK GRENSESNIITT



Figur 5.3: Hjem-siden til MOBESITY-nettsiden

### 5.6.3 Informasjon

Informasjonsdelen av nettsiden er ganske omfattende, siden den tar for seg mesteparten av all informasjon som blir formidlet til pasientene. Opprinnelig ble all denne informasjonen nokså ukritisk kopiert inn i nettsiden fra en ekstern kilde. Det viste seg at dette ikke fungerte veldig bra å navigere i, og under den første brukertesten slet mange med å finne riktig informasjon. Informasjonsstrukturen ble revurdert, og endret slik at det skulle være lettere å finne det en er på utkikk etter. Menyfanen «Informasjon» inneholder generell informasjon, «Kirurgisk behandling» inneholder ganske naturlig informasjon om kirurgisk behandling, og fanen «Livsstilsbehandling» inneholder informasjon som gjelder livsstilsbehandling.

### 5.6.4 Forumet

Et skjermdump av forumet er vist i Figur 5.4. Forumet her et ganske standard forumutseende, og bør være kjent for brukere som har brukt lignende tjenester før. Øverst på forumsiden kan brukeren få opp en liste over foru-

mets medlemmer, endre sin profil og besøke innboksen, der han kan motta og sende private meldinger.

Emner	Startet	Siste innlegg	Innlegg	Visninger
[VIKTIG] Beskjed fra administrator til brukerne	april 13, 2010 19:42 av admin	april 13, 2010 19:49 av Tarjei Ormestøy	2	18
svømmekurs	april 23, 2010 08:52 av Magda Losheim	april 23, 2010 08:52 av Magda Losheim	1	1
Er det noen som skal på kurset i Røros i Mars?	april 19, 2010 12:29 av Nils Olsen	april 19, 2010 15:04 av Rolf Antonsen	2	8
Kan jeg drikke alkohol etter operasjonen?	april 13, 2010 20:13 av Tarjei Ormestøy	april 14, 2010 14:45 av Dr. Berit Oysnes	3	19
Hvor ligger Røros?	april 13, 2010 19:13 av Bruker Olsen	april 14, 2010 08:21 av Dag Svanaes	3	12
Kan noen fortelle meg hvilken Sushi-restaurant som er best i Trondheim	april 13, 2010 19:13 av Bruker Olsen	april 13, 2010 19:13 av Bruker Olsen	1	3

Figur 5.4: Forumet til MOBESITY-nettsiden

### 5.6.5 Kalenderen

Et skjermdump av hovedsiden til kalenderen er vist i Figur 5.5. Her er ukevisning valgt, men det går også an å se måneds- og årsvisning av kalenderen. Brukeren kan legge til aktiviteter ved enten å trykke på plusstegnet på hver time, eller velge «Legg til aktivitet»-valget nederst. Figur 5.6 viser siden man får opp når man legger til en aktivitet. Valgene her er relativt standard kalenderfunksjonalitet som beskrivelser, tidsangivelse og eventuelle gjentakelser.

## 5.6. GRAFISK GRENSESNIITT

Den tredje fanen gir mulighet for å spesifisere tidspunkt for å få påminnelse om aktiviteten per SMS. Lignende valg som disse er også aktuelle om man velger å redigere en eksisterende aktivitet.

Hjem » Kalender

### Kalender

26. April 2010 - 2. Mai 2010 (Uke 17)  
Tarjei Ormestøyl

	Man 26. Apr	Tir 27. Apr	Ons 28. Apr	Tor 29. Apr	Fre 30. Apr	Lør 1. Mai	Søn 2. Mai
08:00							
09:00							
10:00							
11:00							
12:00						12:00» Nå er det 1. mai!	
13:00							
14:00				14:00» Dagsrevyen			
15:00							
16:00							
17:00							
18:00							
19:00							
20:00							

[Utskriftsvennlig]

Måned: Apr 2010  Uke: 26. Apr - 2. Mai  Ar: 2010

Gå til: [Min kalender](#) | [Admin](#) | [Se en annen brukers kalender](#) | [Søk](#) | [Legg til aktivitet](#) | [Hjelp](#)

Figur 5.5: Kalender-siden til MOBESITY-nettsiden

Hjem » Kalender

## Kalender

Robbe Bobbe  
LOGG UT

HJEM | INFORMASJON | KIRURGISK BEHANDLING | LIVSSTILSBEHANDLING | KALENDER | FORUM

Hjem » Kalender

### Legg til aktivitet ( aktivitet )

Detaljer | Gjentagelse | Påminnelser

Kort beskrivelse: Legelime

Full beskrivelse: Legetime hos Dr. Oysnes

Sted: Byåsen Legesenter

Dato: 4 Mai 2010 Velg...

Aktivitet uten tidsangivelse

Lagre

Måned: Mai 2010 Vis Uke: 3. Mai - 9. Mai Vis Ar: 2010 Vis

Gå til: [Min kalender](#) | [Admin](#) | [Søk](#) | [Legg til aktivitet](#) | [Hjelp](#)

Figur 5.6: Legg til aktivitet-siden til MOBESITY-nettsiden

# Kapittel 6

## Brukbarhetstest

Dette kapittelet vil ta for seg formålet med brukbarhetstestene og hvilke forberedelser som ble gjort. Vi vil se nærmere på hvordan den praktiske gjennomføringen av brukertestene foregikk og hvilke problemer og utfordringer som oppstod.

Opprinnelig hadde vi planlagt å gjennomføre kun én brukbarhetstest. Denne skulle gjennomføres med representative brukere, det vil si pasienter eller tidligere pasienter ved Obesitasklinikken. Tidlig i prosjektfasen så vi derimot at utviklingen av vår løsning gikk fortere enn antatt, og vi fant det derfor fornuftig å gjennomføre en ekstra iterasjon med brukertesting.

### 6.1 Formål med testene

Formålet med å gjennomføre brukertester er generelt å få avdekket potensielle brukbarhetsproblemer ved det planlagte brukergrensesnittet. En søker å finne ut hvor og på hvilke måter grensesnittet gir brukeren problemer ved interaksjonen med systemet. Det er lurt å gjennomføre disse på et så tidlig tidspunkt som mulig for å gjøre det minst mulig problematisk å endre noe [42].

Den første brukertesten ble utført på et relativt tidlig tidspunkt i utviklingsprosessen og formålet med denne var derfor, i tillegg til å avdekke feil, å få innspill til redesign av nettsiden og videre utvikling. Nettsiden var ikke ferdig på dette tidspunktet, noe vi var fullstendig klar over, og videre utvikling var planlagt uansett resultat av brukertesten.

Formålet med den andre brukertesten var å teste den viktigste funksjonaliteten ved nettsiden vi utviklet. Størrelsen på nettsiden i denne fasen gjorde at det var realistisk å teste hovedfunksjonaliteten til alle komponentene nettsiden bestod av. Det inkluderte logg inn-funksjon, navigere i den tilgjengelige informasjonen, bruke forumet med tilhørende meldingsfunksjonalitet og å bruke kalenderen.

Samlet sett skulle begge brukertesten bidra til å avdekke brukbarhetsproblemer som vi kunne drøfte videre og eventuelt relatere til bruk av åpen kildekode-komponenter.

## 6.2 Forberedelser

Vi vil i denne seksjonen se på hvordan forberedelsene av brukbarhetstestene ble gjort. Vi utførte to iterasjoner med brukertesting, men fremgangsmåten for de to iterasjonene var relativt lik. Den store forskjellen på de to iterasjonene var personene som ble rekruttert og deltok på testen. Doktorgradstudent Anita Das, som står bak MOBESITY-prosjektet, ledet brukbarhetstestene assistert av teknisk ansvarlig Terje Røsand, mastergradsstudent Ørjan Johansen samt oss selv. Brukertesting foregikk i brukbarhetslaben ved Norsk Senter for Elektronisk Pasientjournal (NSEP).

### 6.2.1 Brukerne

Det var dessverre begrensninger på antall representative brukere vi kunne benytte oss av, og siden vi ikke ville bruke de samme testpersonene flere ganger, ble vi nødt til å finne deltagere i andre omgangskretser. Deltagerne til den første brukbarhetstesten besto derfor av et knippe med kjærester, venninner, klassekamerater og kollegaer. De fleste i denne gruppen holdt derfor et teknisk høyt nivå, og dette kan ha påvirket resultatene vi fikk. Likevel var dette også en fordel, da vi fikk verdifulle tilbakemeldinger rundt designet av nettsiden fra denne gruppen, noe som også var mye av formålet med den første brukbarhetstesten.

Anita Das hadde ansvaret for å rekruttere personer til den andre iterasjonen med brukertesting. Hun rekrutterte 6 pasienter ved Obesitasklinikken, St. Olav. Disse brukerne hadde variert erfaring med å bruke Internett. En person hadde en jobb som gikk ut på å behandle, strukturere og formidle



informasjon, mens en annen brukte sjelden data og Internett. Vi ønsket oss litt flere personer til testingen, men dessverre meldte ikke flere sin interesse.

### 6.2.2 Oppgavene

Brukeren ble bedt om å spille en rolle som pasient ved overvektsklinikken, tilhørende kirurgigruppen. Ut ifra dette utgangspunktet skulle brukerne utføre en del oppgaver ved å bruke den utviklede nettsiden. Oppgavene i begge brukbarhetstestene forsøkte å dekke over all hovedfunksjonalitet, slik at vi kunne avdekke de mest kritiske problemene. Det er enkelte små forskjeller på oppgavene i den første og andre brukbarhetstesten, men disse anser vi ikke for å være av betydning for resultatet.

Overordnet gikk oppgavene ut på å først logge seg inn i systemet ved å bruke vår 2-fase logg inn-funksjonalitet. Deretter skulle brukeren navigere og finne spesiell informasjon i informasjonshierarkiet. Videre skulle brukeren manøvrere i forumet, først finne en spesiell forumoppføring, for så å lage et nytt emne. Brukeren ble så bedt om å lese en melding han hadde fått i innboksen, før han skulle sende en melding til en helsepersonell-bruker. Deretter skulle brukeren finne ut når han hadde legetime neste gang ved å benytte seg av kalenderen samt lage en ny kalenderoppføring med påminnelse. Til slutt skulle brukeren finne kontaktinformasjon til klinikken.

Oppgavene for brukertest I og II er vist i sin helhet i Tillegg B og C.

### 6.2.3 Lokasjon og oppsett

Begge iterasjonene med brukertesting ble utført i brukbarhetslaboratoriet ved NSEP. Testpersonen og fasilitator satt i et adskilt rom som inneholdt en PC med tilkoblet eye-tracking verktøy, samt mikrofon og flere videokamera. Det ble også tilbudt kaffe, te og frukt. Ellers var rommet ganske tomt, og det var et poeng at det var PC-skjermen som skulle fokuseres på. To til tre observatører satt i et overvåkningsrom i nærheten. Herfra kunne de ved hjelp av eye-tracking utstyret følge hvor brukeren så på skjermen og hva han gjorde. De så også flere kameravinkler som bedre viste dialogen mellom fasilitator og testperson. Observatørene var også klare til å bistå dersom fasilitator og testpersonen trengte teknisk assistanse under gjennomføringen.

### 6.2.4 Pilottest

Før hver iterasjon med brukertesting ble det gjennomført en pilottest for å kunne avdekke eventuelle problemer ved det tekniske utstyret, oppgavene eller ved den planlagte gjennomføringen. Pilottesten ble gjennomført så realistisk som mulig for å være sikker på at eventuelle problemer ble avdekket. Pilottesten ble ikke ta i betraktning som et gyldig resultat for videre vurdering.

## 6.3 Gjennomføring

Testpersonene testet løsningen etter tur ved å utføre et sett med oppgaver. Under utførelsen var en fasilitator til stede hele tiden for å introdusere brukerne til testen i tråd med retningslinjene presentert i Seksjon 3.2.2. Vi byttet på rollen som fasilitator, men i den andre iterasjonen tok Anita Das denne rollen under alle testene. Hun hadde intensjoner utover det som er relevant for dette prosjektet, og det var derfor mest hensiktsmessig å gjøre det på denne måten, spesielt med tanke på intervjuet i etterkant av testen.

### 6.3.1 Innsamling av data

Under hver brukbarhetstest ble det samlet inn en stor mengde data. Først ble testpersonene bedt om å fylle ut et spørreskjema, vedlagt i Tillegg D, for å kartlegge deres bakgrunn i bruk av data og Internett samt innhente personlige data som alder og utdanning. Skjemaet hentet også inn opplysninger om bruk av mobiltelefon, men dette er mer relevant for Anita Das sin PhD og vi vil ikke ta med resultatene fra denne delen her.

Under selve testen benyttet vi oss også av eye-tracking-verktøy for å registrere hvor testpersonen hadde fokus på skjermen, samtidig som det ble tatt lydopptak og video fra flere vinkler. 2-3 observatører fulgte hver brukbarhetstest via video fra et annet rom og relevante oppdagelser ble notert i observasjonsskjemaer underveis.

### Tilfredshet, anvendbarhet og effektivitet

Umiddelbart etter testen fylte deltagerne ut et SUS-skjema. Hovedpoenget med et slikt skjema, som forklart i Seksjon 3.2.3, er å måle den generelle

brukbarhetene til systemet som testes ved å innhente data om testpersonens tilfredshet.

Vi registrerte antall sammenbrudd og antall fullførte oppgaver for å måle graden av anvendbarhet. På denne måten kunne vi sammenligne de to brukertestene og se om det ble en nedgang eller oppgang i antall sammenbrudd.

Vi utførte ingen eksakte målinger av effektiviteten til testpersonene under brukertesten. Selv om vi hadde lett tilgang til disse dataene ved for eksempel videoopptak i ettertid, tror vi ikke vi ville fått noe stort utbytte av dette. Vi påstår dette blant annet fordi vi ikke ser effektivitet som en viktig faktor i et slikt system. Andre faktorer, som anvendbarhet og tilfredshet, er mye viktigere tror vi. Ved bruk av et nettsted som dette vil typisk brukeren sitte hjemme, uten noen form for tidspress og med ingen tidskritiske oppgaver å fullføre. Dersom en utvikler et system som er kritisk for å redde liv eller for å spare penger, er selvfølgelig dette en faktor av største viktighet. Dessuten tror vi det er viktigere å fokusere på de kvalitative dataene en får fra en brukbarhetstest, spesielt når antall deltagere er relativt få. Da vil uansett de tallene en får være veldig upålitelige på grunn av den store variansen, og en vil egentlig ikke kunne si noe ut ifra disse resultatene alene.

### **Intervju etter endt brukbarhetstest**

Umiddelbart etter at testerpersonen hadde gjort seg ferdig med oppgavene og fylt ut skjemaene fulgte et oppfølgingsintervju. Her stilte fasilitator en del spørsmål om hvordan brukeren opplevde nettsiden, om hvilke problemer brukeren hadde underveis og om brukeren hadde forslag til endringer. På denne måten oppstod det en diskusjon rundt nettsiden, og mens brukeren hadde nettsiden fremme kunne han navigere rundt og komme med synspunkter.

Vi fikk også gjennom dette intervjuet et inntrykk av tilfredsheten til brukeren. Brukeren kom med positiv og negativ kritikk som var nyttig og ble blant annet spurt om han kunne tenke seg å benytte et slikt system i praksis.

### **6.3.2 Problemer og utfordringer**

Vi fikk store problemer med eye-trackingen underveis. Hovedutfordringen var at flere av testpersonene hadde en annen hodestilling under bruken av nettsida enn de hadde under kalibreringen av eye-trackeren. Dermed registrerte

ikke eye-trackingen øyebevegelsene til brukeren under utførelsen av oppgavene. Vi forsøkte å få brukerne til å sitte i en naturlig bruksstilling under kalibreringen, men flere lente seg nærmere skjermen uansett.

Et annet problem som oppstod var at flere klagde over at tastaturet var «treigt», og det ble ofte skrivefeil fordi noen tastetrykk ikke ble registrert. Dermed brukte flere av testpersonene mye tid under testingen til å rette opp i egne feil (selv om dette egentlig var irrelevant for våre resultater). På et ganske sent tidspunkt oppdaget vi at det faktisk var et problem ved kapasiteten på datamaskinen som førte til treg responstid når eye-trackingprogramvaren kjørte i bakgrunnen. Vi måtte derfor slå av eye-trackingen for de siste to brukertestene.

Disse to problemene gjorde at vi fikk ganske få gode eye-trackingmålinger. Nielsen [49] mener en trenger et relativt stort antall brukere (rundt 15) for å få gode målinger med eye-tracking, og de få målingene vi hadde valgte vi derfor å forkaste.

# Kapittel 7

## Resultater I

Dette kapitlet beskriver resultatene fra den første brukbarhetstesten som ble gjennomført 11. og 12. Mars 2010. Det er verdt å merke seg at systemet på dette tidspunktet hadde et noe annet utseende enn det som er beskrevet i Kapittel 5, men at komponentene er de samme.

### 7.1 Testpersoner

Som forklart i Seksjon 6.3.1 ble testdeltagerne bedt om å fylle ut et skjema i forkant av testen for å kartlegge deres bakgrunn. Vi vil i denne seksjonen oppsummere de viktigste resultatene fra bakgrunnsskjemaet i den første brukbarhetstesten. Fullstendige resultater kan sees i Tillegg E.

Deltagerne til denne testen ble hovedsakelig rekruttert i vår omgangskrets og en overvekt av deltagerne er derfor studenter i aldersspennet 18 til 25 år. Vi kan også fastslå at samtlige testdeltagere har erfaring med bruk av datamaskin. Seks av åtte sier de har mye erfaring med data, og alle testpersonene bruker datamaskin daglig og har dette hjemme.

Når det gjelder erfaring med Internett sier alle at de bruker dette daglig. Seks av deltagerne har estimert ukentlig bruk av Internett i antall timer. Her ligger snittet på 53.33 timer, så det er tydelig at denne gruppen er meget erfarne i bruk av Internett. Ingen av deltagerne benytter seg av færre enn seks av de vanlige Internett-tjenestene som er presentert i skjemaet.

Oppsummert kan vi fastslå at testgruppen i den første brukbarhetstesten har mye erfaring i bruk av både data og Internett. En stor andel av de er også

studenter (og flere av disse studerer IT og har kunnskap om brukergrensesnitt). Selv om de ikke er realistiske brukere av systemet, er de i høyeste grad brukere som kan komme med nyttige innspill i forhold til videre utvikling av nettsiden.

## 7.2 Observasjoner og problemer

I denne seksjonen vil vi presentere alle brukbarhetsproblemer som ble oppdaget under brukbarhetstest I. Vi vil også vurdere alvorligheten til hvert problem og kategorisere problemene i forhold til relevante brukbarhetsheuristikker.

Samtlige brukbarhetsproblemer som ble oppdaget under den første brukbarhetstesten er presentert i tabellene i Seksjon 7.2.3.

### 7.2.1 Gradering av problemer

Hvert brukbarhetsproblem er gradert etter alvorlighet. Vi benytter oss av alvorlighetsgradene *kritisk*, *alvorlig* og *kosmetisk*, som er en vanlig måte å vurdere alvorligheten til brukbarhetsproblemer [56]. Beskrivelsen av de tre alvorlighetsgradene er som følger [57]:

**Kritisk** Kritiske brukbarhetsproblemer er problemer som gjør at testpersonen ikke får fullført en oppgave, eller problemer som forekommer i alle brukbarhetstestene.

**Alvorlig** Problemer i denne kategorien gjør at testpersonen bruker mye ekstra tid på å fullføre en oppgave. Kategorien inkluderer også problemer som forekommer i mange av brukbarhetstestene.

**Kosmetisk** Et kosmetisk brukbarhetsproblem forekommer sjelden eller kan sies å være av kosmetisk natur. Det kan føre til at testpersonen bruker litt ekstra tid på å fullføre en oppgave.

### 7.2.2 Kategorisering av problemer

For kategorisering av brukbarhetsproblemer som oppstod benyttet vi oss av relevante deler av Nielsens 10 heuristikker [10], som er presentert i Seksjon

## 7.2. OBSERVASJONER OG PROBLEMER

---

2.2.2. I tillegg benyttet vi oss av kategoriene *systemfeil* og *metodefeil* som ikke kan knyttes opp mot spesifikke designheuristikker.

Problemene ble kategorisert i forhold mangler eller utilstrekkelighet innenfor de følgende kategorier.

**Konsistens og standarder** Designet bør følge kjente designkonvensjoner og ord og funksjoner bør bety det samme og være konsistente i hele systemet.

**Overenstemmelse med den virkelige verden** Systemet bør snakke brukerens språk. Informasjonen bør være strukturert og formidlet på en måte som virker naturlig og logisk for brukeren.

**Brukerkontroll og frihet** Systemet bør gi brukeren mulighet til å angre og å gå tilbake til tidligere tilstander i systemet. Brukeren bør kunne få navigere fritt fra alle systemets deler.

**Gjenkjenning før erindring** Valg og funksjoner må gjøres godt synlige i systemet, og de må være forståelige uten at brukeren må lære eller huske hva de gjør.

**Synliggjør statusen til systemet** Brukeren bør få tilstrekkelig tilbakemelding etter hver operasjon som gjøres i systemet.

**Estetisk og minimalistisk** Designet bør være fokusert på det som er viktig, og ikke forstyrre brukeren med unødvendig informasjon eller valg.

**Fleksibilitet og anvendbarhet** Systemet bør legge til rette for å kunne bruke snarveier for å raskere utføre operasjoner. Det bør legges til rette for å kunne gjøre oppgaver på alternative måter.

**Systemfeil** Denne kategorien omhandler typisk mangler eller feil ved systemet som dukker opp under testing.

**Metodefeil** Dette er feil som opptrådte under testingen grunnet konteksten, mangler ved metoden eller utførelsen av denne, og ikke feil ved selve systemet.

### 7.2.3 Brukbarhetsproblemer: Brukbarhetstest I

Brukbarhetsproblemene som ble avdekket under brukbarhetstest I er fremstilt i Tabell 7.1 og Tabell 7.2. Hvert problem er gitt en unik ID (Ax for brukbarhetstest I) som vi vil bruke for å referere til problemene senere i oppgaven. Den andre kolonnen refererer til i hvilken oppgave i brukbarhetstesten

Nr	Oppgave	Problembeskrivelse	Gradering	Kategori	Testperson
A1	3b Forum	Kvier seg for å trykke på "Lagre nytt emne"	Alvorlig	Overenstemmelse med den virkelige verden, Konsistens	T1
A2	4 Meldinger	Misliker at Innboksen vises i sin helhet uansett	Kosmetisk	Estetisk og minimalistisk	T1
A3	6c Kalender	Bruker lang tid på å få opp årsvisning.	Alvorlig	Gjenkjenning før erindring	T1, T7, T8
A4	2a Info	Bruker lang tid for å finne informasjon om vektnedgang	Alvorlig	Overenstemmelse med den virkelige verden	T2, T4
A5	2c Info	Bruker lang tid for å finne informasjon om # måltider	Alvorlig	Overenstemmelse med den virkelige verden	T2, T3, T5
A6	5 Meldinger	Treig autoforslagsliste	Kosmetisk	Systemfeil	T2
A7	7 Lenker	Vil første finne kontaktinfo om Røros under 'Informasjon'-tabben	Kosmetisk	Overenstemmelse med den virkelige verden	T2
A8	2 Info	Link fungerer ikke	Kosmetisk	Systemfeil	T3
A9	2c Info	Ønsker en søkefunksjon	Kosmetisk	Fleksibilitet og anvendbarhet	T3
A10	7 Lenker	Død Lenke	Alvorlig	Systemfeil	T3, T6, T7, T8
A11	Eksternt	Trykker escape, og avbryter eyetrackingen (og besøket på nettsida)	Kritisk	Metodefeil	T4
A12	1 Logg inn	Enter-knappen fungerer ikke ved logg inn.	Kosmetisk	Fleksibilitet og anvendbarhet, Konsistens og	T4
A13	2c Info	Finner ikke info om mat etter operasjon. Forstår ikke at operasjon = gastric bypass. <b>Fullførte ikke oppgaven</b>	Alvorlig	Overenstemmelse med den virkelige verden	T4, T8
A14	3b Forum	Kommenterer at under 'valg' er det ikoner som ikke gir mening	Kosmetisk	Estetisk og minimalistisk	T4
A15	5 Meldinger	Skriver feil pga treigt tastatur	Kosmetisk	Metodefeil	T4
A16	3b Forum	Prøver først å legge ut spørsmålet under FAQ på info-sidene.	Kosmetisk	Overenstemmelse med den virkelige verden	T5
A17	3 Forum	Synes det er mye knapper og valg.	Kosmetisk	Estetisk og minimalistisk	T5

Tabell 7.1: Brukbarhetsproblemer A1-A17



### 7.3. ANVENDBARHET

---

Nr	Oppgave	Problembeskrivelse	Gradering	Kategori	Testperson
A18	2a Info	Klikker seg rundt og bruker lang tid.	Alvorlig	Overenstemmelse med den virkelige verden	T6
A19	3c Forum	Usikker på om members-listen han får fram er kirurgi-gruppen.	Alvorlig	Metodefeil	T4, T6, T7, T8
A20	5 Meldinger	Usikker på forskjell mellom medlemsliste og kompisliste.	Kosmetisk	Overenstemmelse med den virkelige verden	T6
A21	6 Kalender	Legger inn aktiviteten, men ikke eksplisitt påminnelse. <b>Fullfører egentlig ikke oppgaven</b>	Kritisk	Gjenkjenning før erindring	T1, T2, T3, T4, T6
A22	2c Info	Fant ikke riktig informasjon. <b>Fullførte ikke oppgaven</b>	Alvorlig	Overenstemmelse med den virkelige verden	T7
A23	6a Kalender	Synes det er rart at dagens dato har samme farge som aktiviteten en annen dag.	Kosmetisk	Konsistens og standarder	T7

Tabell 7.2: Brukbarhetsproblemer A18-A23

problemet oppstod mens den ytterste kolonnen viser hvilke(n) testperson(er) som hadde dette problemet.

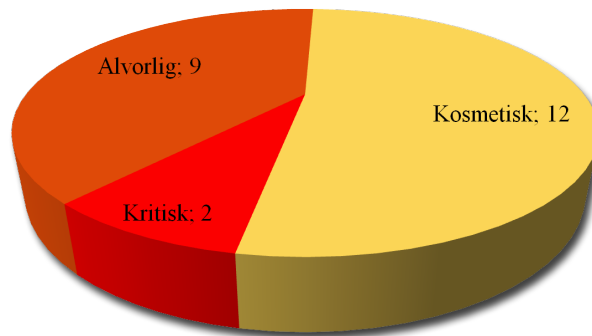
Hvert brukbarhetsproblem er også kort beskrevet, og vi vil beskrive noen av de ytterligere under videre analyse senere i oppgaven. Hvert brukbarhetsproblem er kategorisert og vurdert alvorligheten til.

## 7.3 Anvendbarhet

Denne seksjonen oppsummerer resultater fra den første brukbarhetstesten som er relevant for å måle systemets anvendbarhet. Dette inkluderer antallet brukbarhetsproblemer som oppstod totalt og av hver alvorlighetsgrad, samt antallet fullførte oppgaver og sammenbrudd.

Figur 7.1 viser andelene kosmetiske, alvorlige og kritiske brukbarhetsproblemer. Som vi ser av figuren oppstod det totalt 23 brukbarhetsproblemer i brukertest I, hvorav 2 kritiske, 9 alvorlige og 12 kosmetiske.

Som beskrevet i Seksjon 6.3.1 registrerte vi antall fullførte oppgaver og sammenbrudd under hver brukbarhetstest. Med sammenbrudd mener vi at den aktuelle testpersonen satte seg fast og mislyktes i å fullføre en oppgave. Der- som noen oppgaver ble fullført med assistanse fra fasilitator vil vi også nevne



**Figur 7.1:** Fordeling av alvorlighetsgrad brukbarhetstest I

dette her.

Tabell 7.1 og 8.1 viser at i problemene A13, A21 og A22 ble ikke oppgaven fullført. I brukbarhetsproblem A13 finner ikke T4 og T8 informasjon om måltidsrytme etter operasjon. Problem A22 ligner noe på dette. Her finner deltageren informasjon relatert til dette, men ikke det oppgaven spesifikt spurte etter. A21 handler om å legge inn en aktivitet med påminnelse i kalenderen. Alle testdeltagerne klarte å legge inn aktiviteten, men siden oppgaven ba om påminnelse og fem av brukerne ikke la inn dette tolker vi det som sammenbrudd.

Antall fullførte oppgaver og antall sammenbrudd i brukbarhetstest I er presentert i Tabell 7.3.

<b>Antall fullførte oppgaver</b>	104
<b>Antall sammenbrudd</b>	8

**Tabell 7.3:** Sammenbrudd og fullførte oppgaver i brukbarhetstest I

To av brukerne brukte lang tid og måtte ha assistanse for å finne informasjon om måltidsrytme etter operasjon i oppgave 2c, problem A13. Denne oppgaven var noe dårlig formulert da den brukte andre begreper enn det som stod på nettsiden. De fullførte likevel uten problemer da fasilitator oppklarte begrepet og vi velger derfor å ikke ta med dette som et sammenbrudd.

## 7.4 Tilfredshet

Vi ønsket å måle den generelle brukbarheten til systemet ved å måle testpersonenes tilfredshet ved hjelp av et SUS-skjema. Tabell 7.4 viser SUS-skårene for testpersonene i brukertest I. Vi ser av tabellen at seks av testdeltagerne gir en SUS-skår på 80 eller mer, noe som kan karakteriseres som veldig god tilfredshet med systemet. De to gjenværende deltagerne gir en SUS-skår mellom 60 og 80, som er god tilfredshet.

Testperson	SUS-skår
T1	87,5
T2	85
T3	70
T4	80
T5	65
T6	92,5
T7	97,5
T8	87,5
Gjennomsnitt	83,13
Maks	97,5
Min	65
Konfidensintervall 95%	73,9 - 92,3

**Tabell 7.4:** SUS-resultater brukertest I



# Kapittel 8

## Resultater II

Dette kapittelet beskriver resultatene fra den andre brukbarhetstesten som ble gjennomført i løpet av uke 15 og 16 i April 2010. Vi var på dette tidspunktet i slutfasen av utviklingen og systemets komponenter og grensesnitt var som beskrevet i Kapittel 5.

### 8.1 Testpersoner

Vi vil i denne seksjonen oppsummere de viktigste dataene fra bakgrunns-skjemaet i den andre brukbarhetstesten. Fullstendige data kan sees i Tillegg F.

Av resultatene ser vi at det er fire kvinner og to menn blant testpersonene med et spenn i alder fra 31 til 55 år, og en overvekt av deltagere på rundt 50 år. Det er god spredning i utdanningsnivå og alle unntatt én testperson er i full jobb. Tre testpersoner har litt erfaring med data, mens tre sier de har mye erfaring. Samtlige har datamaskin hjemme og de bruker datamaskin daglig. Ingen av testpersonene har ingen erfaring i bruk av data og Internett.

Fire av seks mener også at de har mye erfaring med Internett og to mener de har litt erfaring. Alle bruker Internett daglig, stort sett hjemme eller på jobb. Ukentlig bruk av Internett spenner fra fem timer til 30 timer, med et gjennomsnitt blant deltagerne på 16.83 timer. Deltagerne skårer også ganske høyt på bruk av Internett-tjenester. Blant de åtte vanlige tjenestene vi presenterte i skjemaet krysset halvparten av deltagerne av for at de brukte alle tjenestene. I tillegg brukte ingen av deltagerne færre enn fire tjenester.

Oppsummert kan vi si at samtlige testpersoner har større eller lik mengde erfaring som en gjennomsnittlig voksen Internett-bruker. De bruker Internett daglig, til å lese e-post, surfe nettaviser og banktjenester og burde følgelig ha gode forutsetninger for å bruke systemet.

## 8.2 Observasjoner og problemer

Brukbarhetstestens hovedfunksjon var å avdekke eventuelle brukbarhetsproblemer i systemet. I denne seksjonen vil vi presentere alle brukbarhetsproblemer som ble oppdaget under brukbarhetstest II. Vi vil også vurdere alvorligheten til hvert problem og kategorisere problemene i forhold til relevante brukbarhetsheuristikker.

### 8.2.1 Gradering av problemer

Hvert brukbarhetsproblem er gradert etter grad av alvorlighet. Vi benytter oss av alvorlighetsgradene *kritisk*, *alvorlig* og *kosmetisk*. Definisjonen av de ulike graderingsnivåene kan sees i Seksjon 7.2.1.

### 8.2.2 Kategorisering av problemer

For kategorisering av brukbarhetsproblemer som oppstod benyttet vi oss av relevante deler av Nielsens 10 heuristikker, som er presentert i Seksjon 2.2.2. I tillegg benyttet vi oss av kategoriene *systemfeil* og *metodefeil* som ikke kan knyttes opp mot spesifikke designheuristikker. Samtlige kategorier er presentert i Seksjon 7.2.2.

### 8.2.3 Brukbarhetsproblemer: Brukbarhetstest II

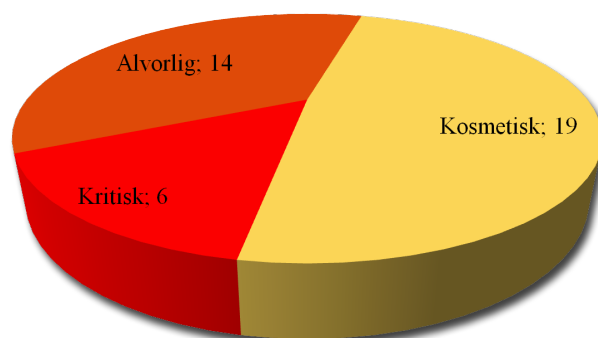
Brukbarhetsproblemer som ble avdekket under brukbarhetstest II er fremstilt i Tabellene 8.1, 8.2, 8.3 og 8.4. Hvert problem er gitt en unik ID (Bx for brukbarhetstest II) som vi vil bruke for å referere til problemene senere i oppgaven. Den andre kolonnen refererer til i hvilken oppgave i brukbarhetstesten problemet oppstod mens den ytterste kolonnen viser hvilke(n) testperson(er) som hadde dette problemet.

Hvert brukbarhetsproblem er også kort beskrevet og vi vil beskrive noen av de ytterligere under videre diskusjon senere i oppgaven. Vi har sett på hvert brukbarhetsproblem og kategorisert og vurdert alvorligheten til dette.

## 8.3 Anvendbarhet

Denne seksjonen oppsummerer resultater fra den andre brukbarhetstesten som er relevante for å måle systemets anvendbarhet. Dette inkluderer antallet brukbarhetsproblemer som oppstod totalt og av hver alvorlighetsgrad, samt antallet fullførte oppgaver og sammenbrudd.

Figur 8.1 viser andelene kosmetiske, alvorlige og kritiske brukbarhetsproblemer. Som vi ser av figuren oppstod det totalt 39 brukbarhetsproblemer i brukertest II, hvorav 6 kritiske, 14 alvorlige og 19 kosmetiske.



**Figur 8.1:** Fordeling av alvorlighetsgrad brukbarhetstest II

Vi kan se av tabellene i Seksjon 8.2.3 at i problemene B9, B20, B29 og B38 ble ikke oppgaven fullført. Brukbarhetsproblem B20 omhandler systemets meldingsfunksjon. I dette problemet klarer ikke T3 å finne valget for å sende melding. B38 er også relatert til dette; her sender T6 melding til seg selv og har derfor ikke fullført oppgaven. I problem B29 klarer ikke testpersonen å finne årsvisningen i kalenderen. I B9 ser vi igjen problemet med at testdeltagerne ikke legger inn eksplisitt påminnelse, som vi var inne på i Seksjon 7.3. Kun én av de seks deltagerne gjorde det oppgaven ba om her, så vi tolker dette som fem sammenbrudd.

Antall fullførte oppgaver og antall sammenbrudd i brukbarhetstest II er presentert i Tabell 8.5.

Nr	Oppgave	Problembeskrivelse	Gradering	Kategori	Testperson
B1	2b Info	Bruker lang tid på å finne informasjon om skattefradrag. Forbinder sosiale spørsmål med noe annet enn systemet legger opp til.	Alvorlig	Overenstemmelse med den virkelige verden.	T1, T4, T5
B2	2b, 2c, 6a Info	Bruker søkefeltet, men skriver feil pga dårlig tastatur	Kosmetisk	Metodefeil	T1
B3	3a Forum	Trykker på 'emnetagger' i forumet, og blir ført til Hjem-siden	Alvorlig	Systemfeil	T1
B4	3a Forum	Forsøker å trykke på forumpostikonet, men ingenting skjer	Kosmetisk	Konsistens og standarder	T1
B5	3b Forum	Ønsker ikke å trykke på "Lagre nytt emne". Tror dette betyr noe annet.	Alvorlig	Overenstemmelse med den virkelige verden.	T1, T5, T6
B6	5 Melding	Skal sende melding til Doktoren, men oppfatter ikke Forumet som privat. Vil ikke at alle skal se PM til doktoren. Leter etter annen mulighet til å sende melding.	Kritisk	Overenstemmelse med den virkelige verden. Konsistens og standarder.	T1
B7	5, 6b	Skriver feil flere ganger pga dårlig tastatur	Kosmetisk	Metodefeil	T1
B8	6a Hjem-siden	Går inn på legetime fra startsidene. Linken åpner som eget bilde(og bypasser Wordpress). Vanskelig å finne tilbake til "navigasjonssiden"	Kritisk	Systemfeil. Brukerkontroll og frihet.	T1
B9	6b Kalender	Legger inn aktiviteten, men ikke inn påminnelse. Forstår ikke at det er det hun skal. <b>Fullfører egentlig ikke oppgaven</b>	Alvorlig	Gjenkjenning før erindring	T1, T3, T4, T5, T6
B10	7b Info	Bruker lang tid på å finne kontaktinformasjon	Kosmetisk	Overenstemmelse med den virkelige verden.	T1
B11	2c Info	Bruker søk, men trykker ikke på noen av svarene. Finner ikke det hun leter etter i tekstsutdraget til søkerresultatene.	Alvorlig	Konsistens og standarder	T2
B12	3b Forum	Bruker lang tid på å velge "Nytt emne". Forstår ikke at dette er rett knapp. Føler rett plass er under "Profil"	Alvorlig	Konsistens og standarder. Gjenkjenning før erindring.	T2

Tabell 8.1: Brukbarhetsproblemer B1-B12



### 8.3. ANVENDBARHET

Nr	Oppgave	Problembeskrivelse	Gradering	Kategori	Testperson
B13	4 Melding	Forstår ikke at meldingen er Privat (siden den er sendt til flere testpersoner også). Tror ikke hun har funnet rett plass.	Kosmetisk	Metodefeil	T2
B14	6b Kalender	Er ikke sikker på om hun har klart å legge inn påminnelse.	Kosmetisk	Synliggjør statusen til systemet.	T2
B15	6c Kalender	Ved visning av hele året kunne det kommet tydeligere fram når det er aktiviteter på en dato	Kosmetisk	Konsistens og standarder	T2
B16	2 Info	Ønsker større skrift	Kosmetisk	Estetisk og minimalistisk design	T3
B17	2a Info	Bruker lang tid på å finne informasjon om vektnedgang. Tror all informasjon ligger under 'Informasjon'-fanen, og ser ikke under 'Kirurgisk'-fanen. Tar lang tid før han merker at det ligger informasjon andre plasser.	Alvorlig	Overenstemmelse med den virkelige verden.	T3
B18	2c Info	Ser ikke valget "Ordforklaringer"	Alvorlig	Estetisk og minimalistisk design	T3
B19	3a Info	Tar ikke skillet mellom hva andre personer har skrevet og den statiske informasjonen på sida. Leser først i informasjonsdelen av nettsiden under mat og kosthold.	Alvorlig	Overenstemmelse med den virkelige verden.	T3, T6
B20	5 Melding	Klarer ikke å sende melding til Dr. Oysnes. Finner ikke valget for det. <b>Fullfører ikke oppgaven</b>	Kritisk	Gjenkjenning før erindring	T3
B21	6b Kalender	Når en trykker på "Velg.." ved datovalget i kalendern, åpnes det et nytt vindu som går i fullskjermmodus. Mister alle muligheter for navigasjon etc.	Kritisk	Metodefeil, Brukerkontroll og frihet	T3
B22	6b Kalender	Når en trykker på "Hjelp" i kalendern, åpnes det et nytt vindu som går i fullskjermmodus. Mister alle muligheter for navigasjon etc.	Kritisk	Metodefeil, Brukerkontroll og frihet	T3
B23	2a Info	Bruker lang tid på å finne informasjon om vektnedgang før kirurgi.	Kosmetisk	Overenstemmelse med den virkelige verden.	T4

Tabell 8.2: Brukbarhetsproblemer B13-B23

Nr	Oppgave	Problembeskrivelse	Gradering	Kategori	Testperson
B24	2c Info	Føler det mangler et nivå til i nedtrekksmenyen så han kan bla i overskrifter.	Kosmetisk	Overenstemmelse med den virkelige verden, Fleksibilitet og anvendbarhet.	T4
B25	5 Melding	Skriver ikke inn meldingstittel.	Kosmetisk	Estetisk og minimalistisk design	T2, T4, T6
B26	6a Hjem-siden	Leter først etter en personlig side "min side"-lignende sak.	Kosmetisk	Fleksibilitet og anvendbarhet	T4
B27	6b Kalender	Forsøker å skrive rett inn i cellen på kalender-forsiden	Kosmetisk	Konsistens og standarder, Fleksibilitet og anvendbarhet	T4, T5, T6
B28	6b Kalender	Usikker på om varighet har noe med påminnelse å gjøre.	Alvorlig	Konsistens og standarder	T4
B29	6c Kalender	Klikker og ser på Måned og Uke-dropdown-boksene, uten å fokusere på År. <b>Fullfører ikke oppgaven</b>	Alvorlig	Gjenkjenning før erindring	T4
B30	6c Kalender	Ønsker å kunne skrive inn et datointervall å få opp liste med hendeleser i tidsrommet	Kosmetisk	Fleksibilitet og anvendbarhet	T4
B31	7a Info	Opplever forskjell i informasjon om kosthold fra det han tidligere har fått.	Kosmetisk	Overenstemmelse med den virkelige verden	T4
B32	2a, 2b Info	Sliter noe med å finne hvilken underside det logisk kan ligge på. Navigerer rundt uten mål.	Alvorlig	Overenstemmelse med den virkelige verden	T5, T2, T3
B33	3b Forum	Usikker på om innlegget er postet etter at lagre er gjort. Mener det burde vært en send/post-knapp.	Alvorlig	Synliggjør statusen til systemet.	T5
B34	4 Melding	Usikker på om hun har fått privat melding. Det står "ingen nye meldinger" fordi hun var i innboksen i forrige oppgave.	Kosmetisk	Synliggjør statusen til systemet.	T5
B35	4 Hjem-siden	Forsøker å trykke på legetime i neste aktiviteter, uten respons.	Kosmetisk	Konsistens og standarder, synliggjør statusen til systemet.	T5
B36	5 Melding	Går først til kontakt-siden.	Alvorlig	Fleksibilitet og anvendbarhet	T5

Tabell 8.3: Brukbarhetsproblemer B24-B36

## 8.4. TILFREDSHET

---

Nr	Oppgave	Problembeskrivelse	Gradering	Kategori	Testperson
B37	6c Kalender	Mener det burde vært en "velg ingen" for måned og uke.	Kosmetisk	Konsistens og standarder, Estetisk og minimalistisk design	T5
B38	5 Melding	Skriver melding til seg selv. <b>Fullfører ikke oppgaven på riktig måte.</b>	Kritisk	Konsistens og standarder	T6
B39	6b Kalender	Mangler kort beskrivelse.	Kosmetisk	Estetisk og minimalistisk design	T6

Tabell 8.4: Brukbarhetsproblemer B37-B39

Antall fullførte oppgaver	76
Antall sammenbrudd	8

Tabell 8.5: Sammenbrudd og fullførte oppgaver i brukbarhetstest II

## 8.4 Tilfredshet

Testpersonenes tilfredshet med systemet er målt ved hjelp av SUS-skjemaet. Tabell 8.6 viser SUS-skårene for testpersonene i brukertest II. Vi ser av tabellen at halvparten av testdeltagerne gir en SUS-skår på 80 eller mer, noe som kan karakteriseres som veldig god tilfredshet med systemet. To deltagere gir en SUS-skår mellom 60 og 80, som er god tilfredshet, mens deltager T3 skiller seg ut med en skår på 52.5, som er OK tilfredshet. Konfidensintervallet i denne brukbarhetstesten er på nesten 30 grunnet relativt få resultater og stor spredning i disse.

Testperson	SUS-skår
T1	77,5
T2	87,5
T3	52,5
T4	67,5
T5	82,5
T6	87,5
Gjennomsnitt	75,83
Maks	87,5
Min	52,5
Konfidensintervall 95%	61,5 - 90,2

**Tabell 8.6:** SUS-resultater brukertest II

# Kapittel 9

## Analyse

I dette kapitlet vil vi analysere funnene fra de to brukbarhetstestene. Vi vil kategorisere brukbarhetsproblemer i forhold til brukbarhetsheuristikker og se på hvilke problemer som kan skyldes bruk av åpen kildekode.

### 9.1 Analyse og kategorisering i heuristikker

I denne seksjonen vil vi gå igjennom hver kategori som er presentert i Seksjon 7.2.2 og gi eksempler på brukbarhetsproblemer som oppstod innenfor hver av disse. Vi vil forsøke å analysere mulige årsaker til at problemene oppstod og demonstrere med skjermbilder fra systemet.

#### 9.1.1 Konsistens og standarder

Vi fant et knippe problemer som kunne knyttes til at designet ikke fulgte kjente designkonvensjoner eller at ord og funksjoner ikke betød det samme over hele nettsida. Problemene A1, A12, A23, B4, B11, B12, B15, B27, B35, B37 og B38 kan knyttes til denne kategorien.

Et eksempel er problem B4, der testperson T1 forsøker å trykke på forumpostikonet ved siden av en forumpost, og oppdager at dette ikonet ikke fungerer som en knapp. I Figur 9.1 er dette ikonet markert med rød firkant. Her hadde det vært naturlig å åpne forumposten når dette ikonet trykkes på, siden ikonet er på størrelse med en knapp og fra andre systemer og programmer er dette ofte vanlig. På denne måten bryter designet med typiske

konvensjoner, noe som er uheldig.



Figur 9.1: Emneknappen kan ikke trykkes på

Et annet fint eksempel finner vi i problem B11, der T2 søker etter ordet «spirometri» ved å bruke søkefunksjonen på nettsiden. Når hun får opp resultatene, vist i Figur 9.2, skjønner hun derimot ikke at søkeresultatene kan trykkes på slik at en åpner den aktuelle siden. Hun tror svaret må befinne seg i tekstutdraget til hvert søkeresultat, men der finner hun det ikke. Hun velger heller å gi opp, og går inn i informasjonsnavigeringen for å finne det hun leter etter. Det kunne vært enda klarere hva som skal klikkes på her, for eksempel kunne hver link hatt en understrek til enhver tid.

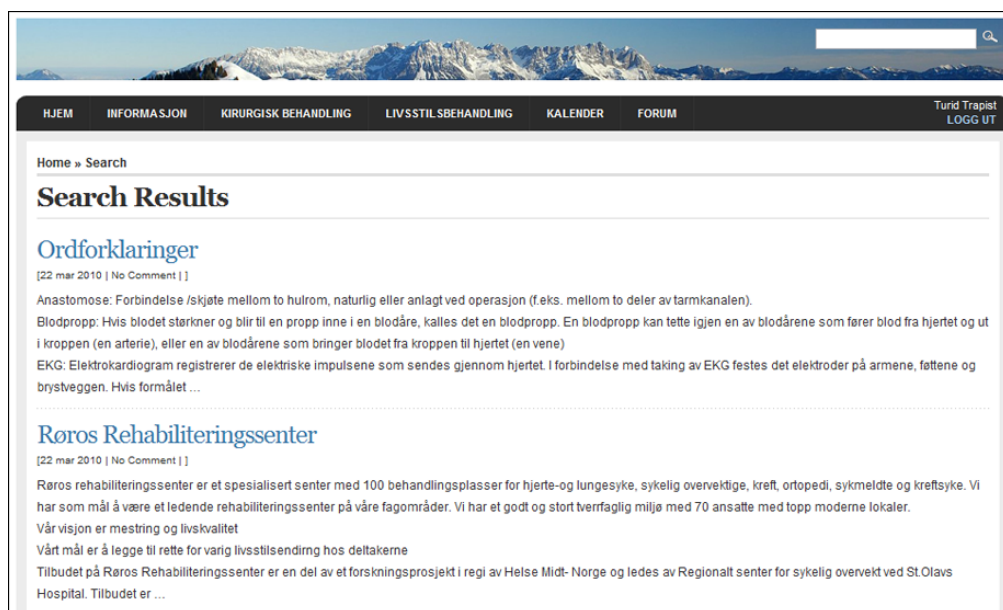
Et annet problem som ble kommentert noen ganger var hvordan aktiviteter ble vist i årsvisningen av kalenderen. I denne visningen var dagene som hadde aktiviteter markert med fet skrift, noe som nesten ikke var synlig for det blotte øyet. Her burde kalenderen fulgt sine egne standarder, og markert dagene på lik måte som de andre visningene gjorde det, det vil si med gul skrift eller gul bakgrunn på de respektive rutene. På denne måten ville de skilt seg bedre ut, samtidig som det ville vært lettere å forstå hvorfor dagene er markert.

### 9.1.2 Overensstemmelse med den virkelige verden

Denne kategorien omhandler brukbarhetsproblemer knyttet til forskjeller mellom brukeren og systemets oppfatning av ord, begreper og handlinger, samt strukturen og formidlingen av informasjon i systemet.

Et vanlig problem under brukbarhetstestene var at testpersonen brukte lang tid på å finne informasjon eller ikke fant informasjon i det hele tatt. Pro-

## 9.1. ANALYSE OG KATEGORISERING I HEURISTIKKER



**Figur 9.2:** Søkeresultater av å søke på Spirometri

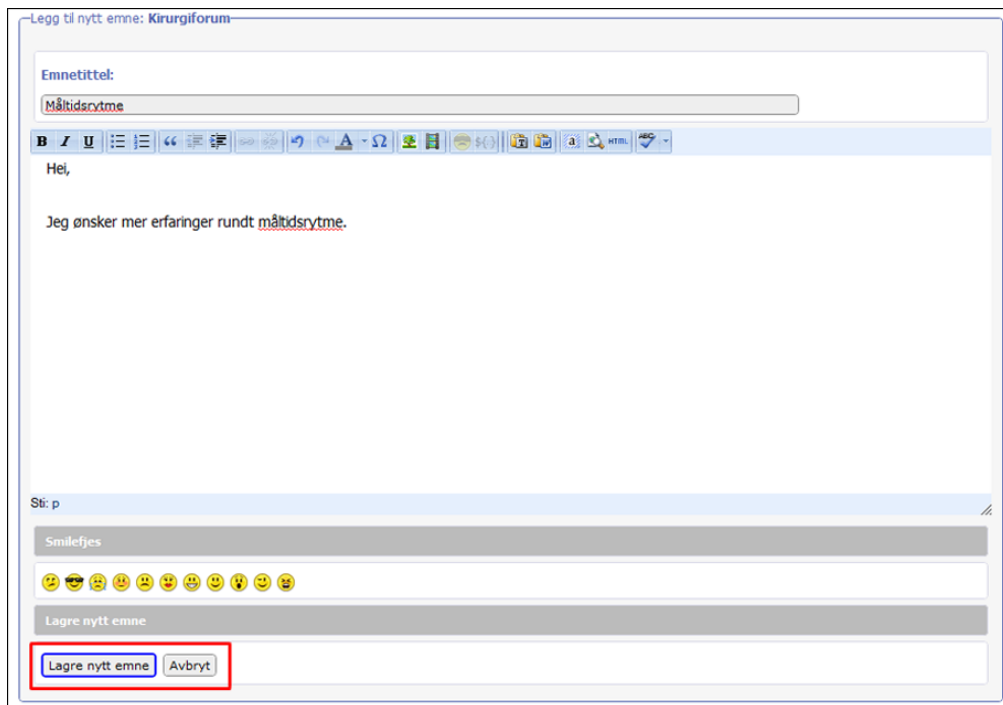
blemene A13, A22, B1, B10, B23, B24 og B32 er direkte knyttet til dette. Disse problemene skyldes hovedsakelig at informasjonen brukerne lette etter lå andre steder på nettsiden enn der de forventet å finne det.

Brukbarhetsproblem B1 er et godt eksempel på dette. Her skulle testpersonene finne informasjon om skattefradrag grunnet overvekt. I systemet ligger denne informasjonen under «Sosiale spørsmål», men tre av testdeltagerne (T1, T4, og T5) hadde problemer med å finne fram til dette. Her er det ikke samsvar mellom brukernes og systemets språk. De nevnte testpersonene kommenterte at de forbinder begrepet «sosiale» med andre ting enn økonomiske spørsmål, som for eksempel seg selv, sine pårørende, erfaringer og denslags. Et utklipp fra denne fanen er vist i Figur 9.3. Vi ser at de aller fleste spørsmålene her omhandler økonomi og kunne med fordel vært skilt ut i egen fane som omhandlet økonomiske spørsmål.

Problemene A16, A20, B5, og B6 omhandler i stor grad forskjeller på systemet og brukerens oppfatning av ord og handlinger. B5 er et godt eksempel denne type feil. Den norske oversettelsen av forumet benytter seg av knappen «Lagre nytt emne» for å opprette en tråd i forumet, som vist i Figur 9.4. Etter å ha skrevet et innlegg som skulle postes i en ny tråd i forumet var tre av testdeltagerne usikre på om «Lagre nytt emne» var riktig knapp å trykke på.



Figur 9.3: Sosiale spørsmål-siden



Figur 9.4: Lagre nytt emne-dialog



Vi ser helt klart hvordan dette kan misforstås. For eksempel kan brukerne være vant til fra e-post-klienter at «lagre» betyr å lagre en kladd eller utkast, men ikke sende e-posten. Et bedre alternativ til navn på denne knappen ville nok vært «opprett emne», «post emne» eller «opprett tråd».

Det er også verdt å trekke fram brukbarhetsproblem B6 hvor en av deltagerne (T1) ikke oppfattet meldingssystemet i forumet som privat, og derfor nølte med å sende melding til doktoren på denne måten. Det er vanskelig å peke på konkrete årsaker til dette, men mangel på erfaring i bruk av forum med meldingssystem kan være én av grunnene. Hun oppfatter antagelig alt på forumet som synlig for alle registrerte brukere, inkludert sendte og mottatte meldinger.

### 9.1.3 Brukerkontroll og frihet

Denne kategorien omhandler prinsippet om at systemet skal gi brukeren mulighet til å angre og å gå tilbake til tidligere tilstander til enhver tid. Brukeren skal også kunne navigere relativt fritt fra alle systemets deler.

Det var få problemer som oppstod under brukbarhetstestene som kan relateres til denne kategorien. Likevel kan vi si at problemene B21, B22 og til en viss grad B8 har noe med dette å gjøre. Felles for disse problemene var at brukeren trykket på et valg eller en link relatert til kalenderen som førte til at systemet åpnet et nytt vindu i fullskjerm-modus, uten noen åpenbar mulighet for å navigere tilbake. Da dette skjedde måtte fasilitator bistå testpersonen i å avslutte fullskjerm-moduset og lukke vinduet.

Etter nærmere undersøkelse tror vi at denne feilen kan ha skyldtes nettleserinnstillingene på PCen som ble brukt under testingen og den kan derfor karakteriseres som en metodefeil.

### 9.1.4 Gjenkjenning før erindring

Brukbarhetsproblemer som skyldes at valg og funksjoner er lite synlige i systemet hører inn under denne kategorien. Dette inkluderer også funksjoner som er lite forståelige eller som må huskes av brukeren.

I oppgaven hvor testpersonen skulle legge inn en aktivitet med påminnelse i kalenderen oppstod problemene A21 og B9 som er relatert til dette. Nesten samtlige brukere la inn aktiviteten i kalenderen uten å spesifisere påminnelse, og oppfattet det som å ha løst oppgaven. Årsaken til dette kan være at de

ikke leste oppgaven godt nok, men vi tror også at mye skyldes at valget om påminnelse er lite synlig i kalenderen eller at brukerne tror at alle aktiviteter automatisk legges inn med påminnelse.

Som vi ser av skjermbildet i Figur 9.5 ligger valget om påminnelse i en egen fane i opprett aktivitet-menyen. I denne fanen må brukeren merke av for at han ønsker påminnelse (denne er satt til «Nei» som standard) og så spesifisere nærmere tidspunkt for påminnelsen. Innstillingene her er i overkant avanserte og kunne med fordel vært gjort enklere og plassert i samme dialog som aktivitetsbeskrivelse og tidspunkt legges inn. Hadde brukerne fått valget om å legge inn påminnelse «Ja» eller «Nei» i dialogen som skjermbildet viser tror vi at betraktelig flere hadde lagt inn påminnelse.



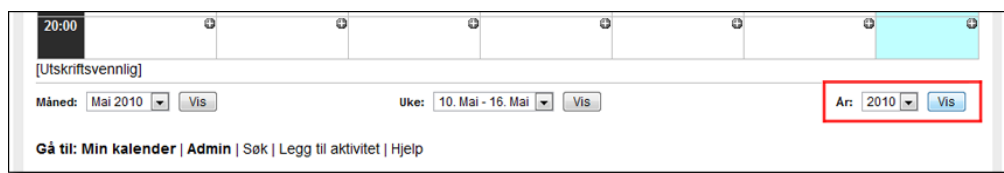
The screenshot shows a web-based dialog titled "Legg til aktivitet ( aktivitet )". It has three tabs: "Detaljer", "Gjentagelse", and "Påminnelser", with "Påminnelser" highlighted by a red box. The "Kort beskrivelse:" field contains "Ny aktivitet". Below it is a large text area for "Full beskrivelse:". The "Sted:" field is empty. The "Dato:" field shows "25", "Mai", and "2010" with a "Velg..." button. Below that is a dropdown menu set to "Aktivitet med tidsangivelse". The "Klokken:" field shows "09" and "00". The "Varighet:" field shows "0" and "00" with the text "(timer: minutter)". A "Lagre" button is at the bottom left.

**Figur 9.5:** Legg til aktivitet med påminnelse

Et annet problem mange hadde var å finne årsvisningen til kalenderen. Dette gjelder brukbarhetsproblemene A3 og B29. Valget for dette er vist i Figur 9.6. Vi tror de aller fleste så dette valget, men at de oppfattet det som at det gjorde noe annet enn å vise hele året. En teori er at brukerne trodde de tre boksene for måned, uke og år under kalenderen hang sammen og at om man trykte «Vis» ved en av de ville man få fram den spesifiserte uka i den spesifiserte måneden i det valgte året. Det kunne nok vært gjort tydeligere at hver knapp viste en måned, en uke eller et helt år.

## 9.1. ANALYSE OG KATEGORISERING I HEURISTIKKER

---



**Figur 9.6:** Kalender med årsvisnings-valg

Vi kan i tillegg samle brukbarhetsproblemene B12 og B20 inn under kategorien «Gjenkjenning før erindring». Dette var problemer som omhandlet at brukerne ikke fant valgene de lette etter i systemet.

### 9.1.5 Synliggjør statusen til systemet

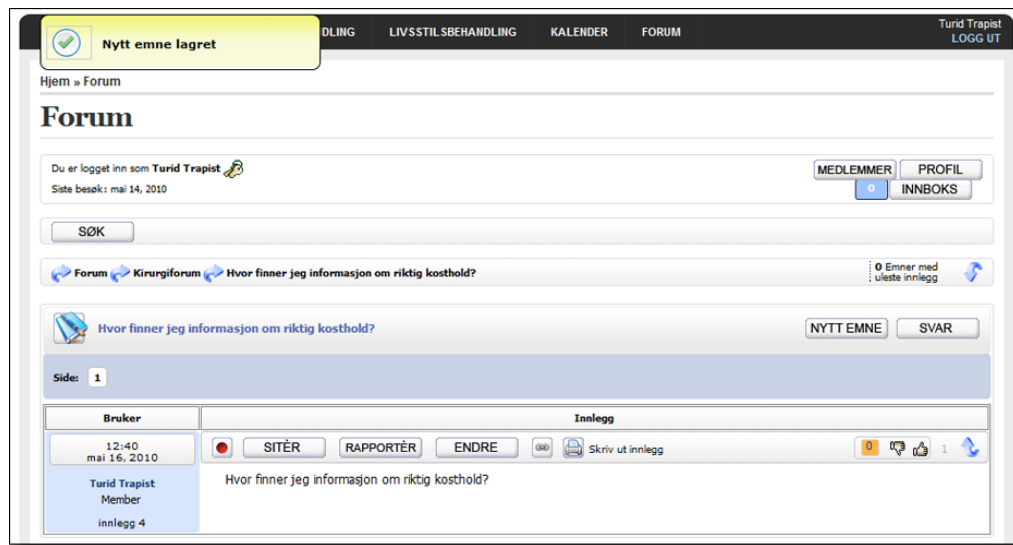
Denne kategorien vedrører de feilene som går på at brukeren ikke får tilstrekkelig tilbakemelding om hva som skjer «i kulissene» i systemet. Det oppstod relativt få feil som kunne plasseres i denne kategorien. Feilene som kan knyttes opp mot denne kategorien er B14, B33, B34 og B35.

B14 var et problem som oppstod én gang, da testperson T2 skulle legge inn påminnelse for en oppføring i kalenderen. Etter at T2 lagret påminnelsen kom T2 direkte til kalenderforsiden uten noen form for bekreftelse eller tilbakemelding om at påminnelsen var lagret. T2 kommenterte at han ikke visste om påminnelsen nå var lagret.

B33 er et annet godt eksempel fra denne kategorien, der brukeren ikke får god nok tilbakemelding. Testperson T5 skulle her poste et innlegg i forumet. Den norske forumoversettelsen bruker «lagre innlegg» som knapp for å poste et innlegg. Når brukeren har trykket på denne, får han opp en liten boks som sier at innlegget er lagret, og en blir sendt videre til sitt nye innlegg. T5 var her usikker på om innlegget faktisk var postet. Hun så ikke den lite synlige tilbakemeldingen fra systemet, og hun var også usikker på om «lagre» faktisk postet innlegget. Skjermen hun får opp etter at hun har trykket «Lagre nytt emne» er vist i Figur 9.7.

### 9.1.6 Estetisk og minimalistisk

Denne kategorien omhandler problemer som oppstår fordi designet ikke er fokusert på det som er viktig, eller at det forstyrrer brukeren med unødvendig



Figur 9.7: Notifikasjonen som viser at et emne er postet.

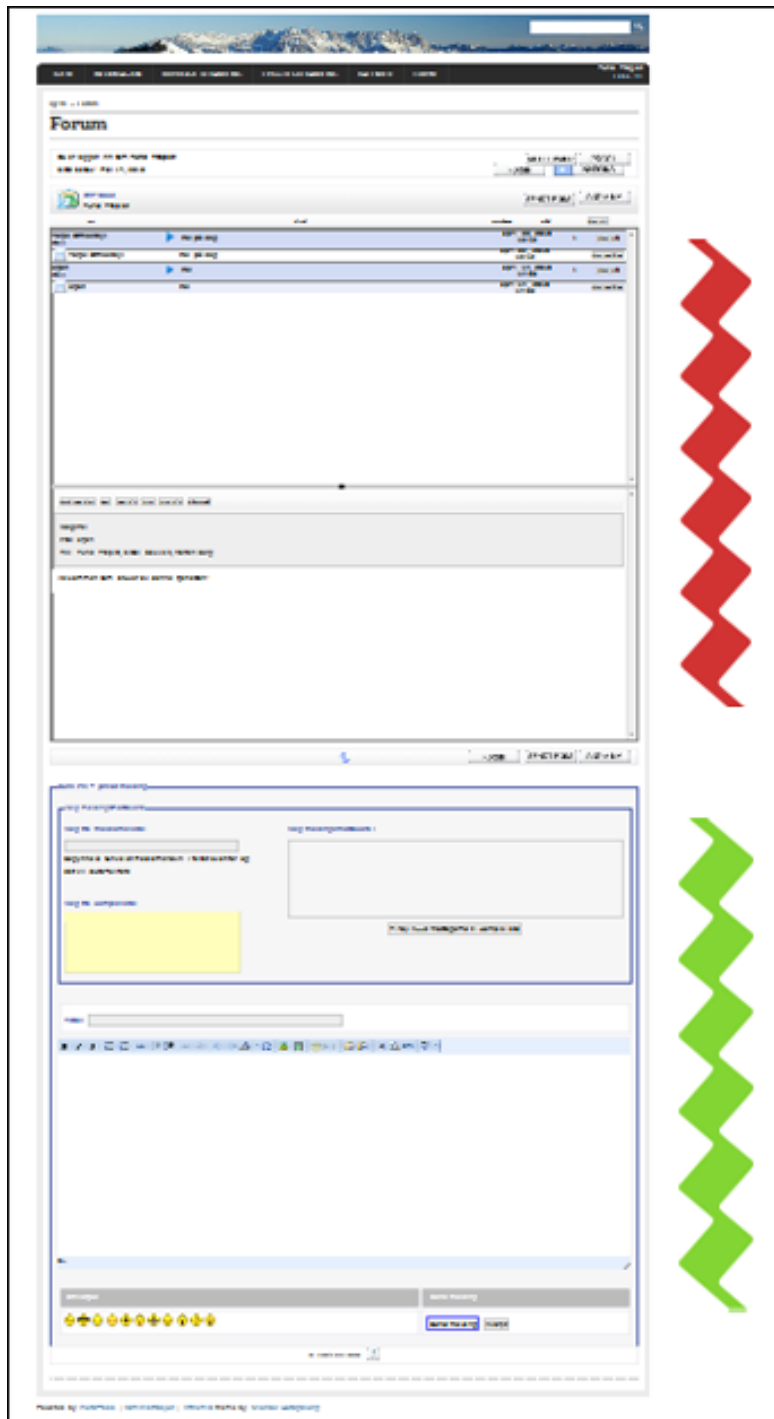
informasjon eller valg. Problemene A2, A14, A17, B16, B18, B25, B37 og B39 kan knyttes til denne kategorien.

Et problem som gikk igjen i begge iterasjonene med brukertesting var problem A2. Når testpersonen er inne i meldingssystemet i forumet, vises alltid innboksen og den øverste meldingen øverst på sida, selv om brukeren ønsker å gjøre noe annet, for eksempel skrive ny melding. Dette tar opp mye skjermplass, og det er ikke gunstig at funksjonaliteten de ønsker finnes under meldingen og langt nede på siden.

Problem A2 er vist i Figur 9.8, her har brukeren valgt å skrive ny melding. Den grønne markeringen på høyre side av figuren viser funksjonaliteten brukeren ønsker å få opp (vindu for å skrive ny melding). Den røde markeringen viser innboksen og nyeste melding, i tillegg til mye unødvendig hvitt tomrom.

Et annet eksempel er problem A17, der T5 synes det er mange knapper og valg i forumet, og blir forvirret. Som standard har en blant andre valgene «watch» og «subscribe», men for en uerfaren forumbruker vil ikke disse knappene gi mening.

## 9.1. ANALYSE OG KATEGORISERING I HEURISTIKKER



Figur 9.8: Innboks-visningen i forumet

### 9.1.7 Fleksibilitet og anvendbarhet

Denne kategorien omhandler feil som kan knyttes til tilretteleggelse for erfarne brukere ved for eksempel snarveier og alternative løsningsveier. Problemene A12, B24, B26, B27, B30 og B36 kan kobles til denne kategorien.

Et fint eksempel er problem A12 som omhandler innloggingen. Etter å ha skrevet inn logg inn-informasjonen så vil brukeren trykke enter-knappen for å gå videre. Dessverre så fungerer ikke dette, og brukeren må heller ta i bruk musa for å klikke på en knapp. Dette er selvfølgelig ingen alvorlig feil, men det er slike feil som er irriterende for en avansert bruker.

Et problem som gikk igjen i den andre brukertesten var B27, der flere av brukerne forsøkte å skrive rett inn i datofeltene i kalenderen. Dette støttes ikke av kalenderen. Grunnen til at flere forsøkte dette er nok at de er vant med å bruke Excel-ark, og siden datofeltene kan minne om Excel-celler antok de at dette var mulig også her.

### 9.1.8 Systemfeil

Noen få av brukbarhetsproblemene som oppstod under testingen omhandlet typiske «bugs» eller mangler ved systemet og kan karakteriseres som systemfeil.

Et eksempel er problemene A8 og A10 fra den første brukbarhetstesten hvor vi ba brukerne finne informasjon om Røros Rehabiliteringssenter og de ble sendt til en side med en død link.

En testdeltager avdekket også at emnetag-støtten som ble lagt inn i forumet i forkant av brukbarhetstest II ikke fungerte som det skulle. I problem B3 forsøkte brukeren å trykke på en av taggene under et emne i forumet og ble så «kastet» tilbake til hjemmesiden. Et gjennomgående, men kosmetisk problem (A6) som én av testpersonene kommenterte, var den trege autoforslagslisten i forumets «Skriv melding»-dialog.

Brukbarhetsproblem B8 var den eneste kritiske systemfeilen som ble avdekket under brukbarhetstestene. På dette tidspunktet hadde vi trykkbare lenker i oversikten over neste kalenderaktiviteter på hjem-siden. Under oppgaven som gikk ut på å finne neste legetime trykket en av testdeltagerne på denne linken, og ble så videresendt til kalendervisningen for denne aktiviteten utenfor selve kalenderfanen i Wordpress.

### 9.1.9 Metodefeil

Dette er feil som opptrådte under testingen grunnet konteksten, mangler ved metoden eller utførelsen av denne, og ikke feil ved selve systemet.

Et problem som gikk igjen noen ganger, spesielt i den første brukertesten, var at noen av oppgavene var dårlig formulert. Dette førte til misforståelser, og at testpersonen ikke gjorde det vi ville han skulle gjøre. Dette var selvfølgelig feil fra vår side, og kan ikke lastes systemet.

En annen metodefeil som gikk igjen var at tastaturet var tregt og dårlig som følge av eye-tracking programvaren. Dette førte til irritasjon og frustrasjon fordi testpersonen skreiv feil flere ganger. For eksempel ved problem B2 forsøkte testperson T1 å søke på nettsiden, men fikk ikke opp noe resultat fordi hun hadde skrevet feil. Vi ble tvunget til å avslutte eye-trackingen for å kunne gjennomføre alle brukertestene. Problem B7 omhandler også dette.

En annen feil vi gjorde var relatert til oppgave 4, der vi sendte en privat melding til alle brukerne. Dette førte derimot til problem B13, der testpersonen ikke oppfattet meldingen som privat siden den var sendt til flere brukere. Vi skulle selvfølgelig sendt en melding til hver enkelt bruker for å forhindre denne misforståelsen.

Problem B21 og B22 var også metodefeil. Disse er forklart i under kategorien *brukerkontroll og frihet* i Seksjon 9.1.3, da de hørte inn der også.

## 9.2 Relasjon til fri programvare

Denne oppgaven dreier seg mye om i hvilken grad brukbarhetsproblemer ved det utviklede systemet skyldes bruk av fri programvare. I denne seksjonen har vi derfor forsøkt å samle de identifiserte brukbarhetsproblemene fra Kapittel 7 og 8 i tre kategorier:

- Feil som ikke er knyttet til bruk av åpen kildekode-komponenter.
- Feil som er knyttet til hver enkelt åpen kilde-komponent.
- Feil som er knyttet til integrasjonen av de ulike åpen kildekode-komponentene.

Hver kategori er også delt inn i underkategorier som vi vil utdype nærmere i de påfølgende avsnittene. Det er verdt å merke seg at problemene som ble karakterisert som metodefeil er utelatt her da disse ikke kan lastes systemet.

### 9.2.1 Feil som ikke er knyttet til åpen kildekode-komponenter

Brukbarhetsproblemer som ikke direkte skyldes åpen kildekode-komponenter hører inn under denne kategorien.

Eget design-seksjonen omhandler hvordan vi selv har organisert nettsiden; det vil si den logiske inndeling av siden og struktureringen av informasjon. Hva innholdet er på de forskjellige sidene og hvor disse ligger hører også inn under her.

Egenutviklede komponenter-seksjonen gjelder hovedsakelig single sign-on-løsningen samt enkelte widgets<sup>1</sup> som er plassert på hjem-siden.

I seksjonen som heter annet har vi plassert brukbarhetsproblemer som ikke kan relateres til noen av de andre kategoriene.

#### Eget design

Denne seksjonen omhandler problemer knyttet til vår egen strukturering og organisering av nettsiden og dens komponenter. Problemer som er knyttet til dette må derfor lastes oss, og ikke de ulike åpen kildekode-komponentene i seg selv. Wordpress som et CMS tilbyr god støtte for å publisere og strukturere innhold etter brukerens ønske og vi har gjort bruk av denne plattformens verktøy i vår utforming av nettsiden.

De fleste av problemene under denne kategorien omhandler at testpersonene brukte lang tid og hadde problemer med å finne rett informasjon i oppgavene. Dette har helt klart sin årsak i hvordan informasjonen ble strukturert på nettsiden. På bakgrunn av problemene vi fant i den første iterasjonen med brukertesting endret vi struktureringen av informasjonen radikalt til den andre brukertesten. Likevel fant vi omtrent like mange problemer knyttet til dette i begge brukertestene. Vi tror dette kan komme av at det var vesentlige forskjeller i tekniske ferdigheter og erfaringer mellom brukergruppene i brukbarhetstest I og II. Dersom dette nivået hadde vært likt for begge gruppene, har vi grunn til å anta at antall problemer ville gått ned.

Problemene som ikke er knyttet til informasjonsstruktureringen omhandler at testpersonene har forventinger til siden som ikke blir innfridd, eller at de misforstår begreper og oppbygningen av siden. For eksempel i problem A16 leter en av testpersonene etter muligheten til å legge ut spørsmål på en av

---

<sup>1</sup>Instikk eller små kodeelementer som kan plasseres i sidemenyen på en Wordpress-side.



## 9.2. RELASJON TIL FRI PROGRAMVARE

---

informasjonssidene, da denne var strukturert som en «Ofte stilte spørsmål»-side.

Alle problemene i denne kategorien er vist i Tabell 9.1.

<b>Eget design</b>	A4, A5, A7, A9, A13, A16, A18, A22, B1, B10, B17, B18, B23, B24, B26, B32
--------------------	---

**Tabell 9.1:** Problemer knyttet til eget design

### Egenutviklede komponenter

Det eneste problemet i denne kategorien var A12, der Enter-knappen ikke fungerte ved innlogging. Dette er et kosmetisk problem som ble utbedret før den andre brukertesten, og som utelukkende skyldes vår egen implementasjon av innloggingsmekanismen.

### Annet

Her havner brukbarhetsproblemene som ikke passer inn under andre kategorier. Disse problemene kan ikke direkte lastes systemet, og er ikke hensiktsmessig å diskutere ytterligere på grunn av problemenes natur. Problemene her består av ubetydelige småfeil, misforståelser og små kommentarer fra brukerne. Tabell 9.2 viser en oversikt over denne kategorien.

<b>Andre problemer</b>	A8, A10, B31, B34, B38
------------------------	------------------------

**Tabell 9.2:** Andre problemer

### 9.2.2 Feil som er knyttet til hver enkelt åpen kildekode-komponent

Brukbarhetsproblemer som kan spores tilbake til bruk av en enkelt åpen kildekode-komponent havner i denne kategorien.

Vi har valgt å dele opp denne kategorien i to underkategorier. I første seksjon ligger problemene som skyldes våre egne tilpasninger av komponentene. Den neste seksjonen omhandler komponentenes standard, ut-av-boksen funksjonalitet og utseende.

### Egne tilpasninger

Noen av problemene er knyttet til våre egne tilpasninger av åpen kildekode-komponentene. Dette innebærer endringer gjort på utseendet til komponentene, og skjuling av funksjonalitet for brukeren. Problemene i denne kategorien er vist i Tabell 9.3.

<b>Egne tilpasninger</b>	A14, A23, B11, B27
--------------------------	--------------------

**Tabell 9.3:** Problemer knyttet til egne tilpasninger av komponenter

### Standard funksjonalitet

Problemene i denne kategorien er knyttet til komponentenes opprinnelige funksjonalitet og utseende. Vi ser at dette er den kategorien med flest problemer. Problemene her spenner over alle de ulike brukbarhetskategoriene; Alt fra dårlig språk og uforståelige valg til mangelfull tilbakemelding fra systemet. De spenner også over alle de ulike åpen kildekode-komponentene, selv om majoriteten tilhører kalenderen og forumet. Tabell 9.4 viser problemene i denne kategorien.

	Wordpress	Kalender	Forum
<b>Standard funksjonalitet</b>	B16	A3, A21, B9, B14, B15, B28, B29, B30, B37, B39	A1, A2, A6, A17, A20, B4, B5, B12, B20, B25, B33

**Tabell 9.4:** Problemer knyttet til standard komponentfunksjonalitet

### 9.2.3 Feil som er knyttet til integrasjonen av de ulike åpen kildekode-komponentene

I denne seksjonen har vi samlet brukbarhetsproblemene som skyldes integrasjonen og sammenkoblingen av åpen kildekode-komponentene vi tok i bruk. Problemene er vist i Tabell 9.5.

Vi brukte Wordpress som hovedplattform for nettsiden, og integrerte kalender og forum inn i denne. Forumet lot seg integrere nokså sømløst, da dette var utviklet som et innstikk til Wordpress. Kalenderen derimot, førte med

### 9.3. TILFREDSHETEN BLANT TESTPERSONENE

---

seg noen problemer, da denne var beregnet for å fungere som en selvstendig nettside. Vi fant ingen annen mulighet enn å integrere kalenderen ved hjelp av iframes i en Wordpress-side, noe som er suboptimalt.

Dette førte blant annet med seg vanskeligheter med å linke til kalenderinnhold fra andre steder på nettsiden enn «inni» kalenderens egen iframe. Før brukbarhetstest II laget vi en klikkbar liste over neste aktiviteter i kalenderen for den innloggede brukeren på hjem-siden. Problemet med dette var at når brukeren klikket på aktiviteten ble han sendt til en kalender-side med info om aktiviteten utenfor selve kalender-fanen i Wordpress, med ingen annen vei tilbake enn å trykke «Tilbake» i nettleseren. Dette skapte forvirring rundt om brukeren fortsatt befant seg i systemet og om han var logget inn.

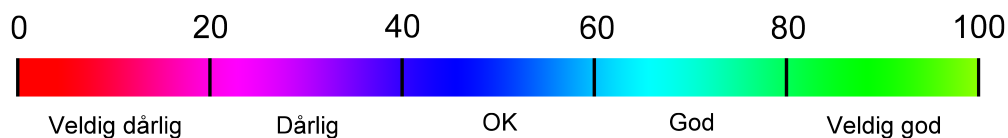
Et annet problem var at noen av testpersonene hadde vansker med å forstå skillet mellom hva som var privat og hva som var synlig for alle. Dette kan vi knytte til vår integrasjon av komponentene, men det er ikke i så måte en teknisk feil, heller en konseptuell misoppfatning av systemet som helhet. Blant annet gjelder dette enkelte brukeres forståelse av meldingssystemet.

<b>Integrasjonsproblemer</b>	B3, B6, B8, B19, B35, B36
------------------------------	---------------------------

**Tabell 9.5:** Problemer knyttet til integrasjon av komponenter

## 9.3 Tilfredsheten blant testpersonene

Fra brukertest I ser vi at brukerne ga systemet en SUS skår som ligger innenfor et 95% konfidensintervall på 73,9 til 92,3 og med en gjennomsnittlig skår på 83,1. I brukertest II ga resultatet et 95% konfidensintervall på 61,5 til 90,2 og en gjennomsnittlig skår på 75,8. Som en ser av Figur 9.9 betyr dette at testpersonene i brukertest I opplevde *veldig god* tilfredshet ved å bruke systemet, mens den andre testgruppen opplevde *god* tilfredshet.



**Figur 9.9:** SUS-skalaen

SUS-skåren fikk en liten nedgang etter den andre brukertesten. Vi tror dette kan ha noe å gjøre med det tekniske nivået til brukerne i den første iterasjonen, som lå høyt over den andre gruppen. Alder kan også ha hatt en betydning for dette resultatet. Testpersonene i den første iterasjonen var hovedsaklig i aldersgruppen 21-25 år, mens den andre brukertesten hadde et gjennomsnitt på rundt 50 år. Vi tror dette kan ha påvirket tilfredsheten da de som bruker mye Internett og data til vanlig har lettere for å sette seg inn i nye nettsider fordi de har sett mye likt før, og kan relatere til dette. Resultatene kan også ha vært påvirket av at de fleste fra brukertest I kjente medlemmer av prosjektgruppen, og kanskje ikke var så kritiske til systemet som de hadde vært om de var reelle framtidige brukere og ikke kjente noen av prosjektmedlemmene personlig.

For å finne ut om det er noen signifikant endring mellom de to resultatene kan en bruke Student  $t$ -Test med ulik størrelse og varians i de to prøvene. Dersom en antar at nullhypotesen er at de to tallgruppene med SUS-resultater er like vil en få en sannsynlighet på 28,9% for at nullhypotesen er riktig. Dette betyr at det er 29% sannsynlig at de to SUS-resultatene er like. Dette tallet sier at en ikke kan konstatere at de to resultatene hverken er like eller ulike.

# Kapittel 10

## Diskusjon

Dette kapitlet diskuterer de presenterte funnene fra denne studien ut ifra problemstillingen og forskningsspørsmålene. Disse funnene vil bli sett på i lys av tidligere presentert litteratur. Videre vil kvaliteten av resultatene diskuteres, og vi vil vurdere funnenes validitet, generaliserbarhet og reliabilitet. Til slutt vil forskningsmetodene vi har benyttet oss av diskuteres med fokus på nytteverdi og gjennomføring.

### 10.1 Problemstilling

I denne seksjonen vil vi diskutere og besvare oppgavens forskningsspørsmål og danne grunnlaget for å konkludere disse i neste kapittel.

Som beskrevet i Seksjon 1.3 var hovedmålet med dette prosjektet å se på *hvilke utfordringer knyttet til brukbarhet som kan oppstå ved bruk av fri programvare i utviklingsprosjekter, og hvordan disse kan reduseres.*

For å besvare dette spørsmålet vil vi nå diskutere de to underliggende forskningsspørsmålene.

#### 10.1.1 Forskningsspørsmål 1

Vi vil her diskutere FS1: *Hvilke brukbarhetsutfordringer fører utvikling basert på fri programvare med seg?*

For å kunne svare på dette gjennomførte vi to brukbarhetstester av et system vi utviklet basert på fri programvare. Gjennom dette ble brukbarhetsproblemer avdekket og vi fikk tilbakemeldinger til videreutvikling og redesign. Etter å ha analysert disse funnene har vi kategorisert de i forhold til relevante brukbarhetsheuristikker og forsøkt å identifisere hvilke problemer som kan knyttes til bruk av åpen kildekode-komponenter. Vi har også vurdert anvendbarheten og tilfredsheten til systemet.

### Type brukbarhetsproblemer

Det ble avdekket totalt 62 brukbarhetsproblemer under de to brukbarhetstestene. Av disse har vi knyttet 32 problemer opp i mot bruken av fri programvare. Vi kan så ta i betraktning disse problemene for å se hvilke brukbarhetsheuristikker som «syndes» mest og hvilke type problemer som oppstod oftest. I Seksjon 7.2.2 er kategoriene vi brukte for å gruppere brukbarhetsproblemer presentert. Tabell 10.1 viser hvor mange brukbarhetsproblemer relatert til fri programvare som forekommer i hver av disse kategoriene. Grunnen til at antall problemer her kan summeres opp til flere enn 32 er at enkelte kan knyttes til flere kategorier.

Kategori	Antall
Konsistens og standarder	11
Estetisk og minimalistisk	7
Gjenkjenning før erindring	6
Overensstemmelse med den virkelige verden	5
Synliggjør statusen til systemet	3
Fleksibilitet og anvendbarhet	3
Systemfeil	3
Brukerkontroll og frihet	1

**Tabell 10.1:** Antall åpen kildekode-problemer i hver kategori

Som vi ser av tabellen er det klart flest brukbarhetsproblemer relatert til brudd på prinsippet om at systemets design skal være konsistent og følge kjente designstandarder og konvensjoner. Brukerne opplever at nettsiden ikke oppfyller de grensesnitt-konvensjoner de er vant med fra kommersiell programvare, som for eksempel Microsoft Excel og Outlook.

Det kan være mange årsaker til at åpen kildekode-komponentene er svake på dette området. Det er nærliggende å tro at prosjektenes lange levetid og distribuerte natur, med mange involverte utviklere og ulike syn på brukbarhet

er en viktig årsak. Vi vet blant annet at fri programvare har mye å lære av kommersiell programvare hva gjelder standardisering og konvensjoner på visuelle og interaktive elementer [23]. Det er behov for et felles vokabular og en felles forståelse av anerkjente brukbarhetsprinsipper [30].

Det er verdt å merke seg at vi ikke ser like mange problemer relatert til den interne designkonsistensen i systemet og vi fikk få tilbakemeldinger fra brukerne når det gjaldt systemets «look and feel», farger og stil. Et mulig svar på dette er at vi lyktes med å tilpasse komponentenes utseende slik at det hele framstod som et helhetlig system i tilstrekkelig grad. En annen teori er at brukerne ikke hadde problemer med dette, og derfor ikke brydde seg med å kommentere hvordan det så ut.

Vi innser at det endelige grafiske designet ikke hadde vunnet noen designpris for nettsider. De ulike komponentene bruker ofte ulik font, ulike knapper, ulike farger og ulik informasjonsformidling. Det er likevel viktig å påpeke at vi ikke fant noen alvorlige problemer med grensesnittet, og gjennom brukbarhetstestene har vi funnet at nettsiden i stor grad er *anvendbar*. Tatt i betraktning de tilgjengelige ressursene og metodene som er brukt er det ikke å forvente at nettsiden kan tilby all funksjonaliteten som den gjør, og i tillegg ha et perfekt brukergrensesnitt. Ut ifra prosjektets rammer må vi si oss fornøyd med det endelige grafiske designet.

Ut ifra Tabell 10.1 ser vi at det også syndes noe når det gjelder estetisk og minimalistisk design i det brukertestede systemet. Sju av de åpen kildekode-relaterte problemene omhandler dette og brukerne kommenterer at enkelte komponenter har mange unødvendige og uforståelige knapper og funksjoner. En åpenbar grunn til dette er at komponentene vi har tatt i bruk ikke er skreddersydd brukergruppen. Dette er en typisk risiko ved bruk av fri programvare framfor å utvikle selv eller anskaffe programvare spesielt utviklet med tanke på brukernes behov.

Et annet poeng er at åpen kildekode-prosjekter ofte lider av at programvaren vokser i kompleksitet over tid. Ny funksjonalitet blir kontinuerlig lagt på, uten at eksisterende funksjoner fjernes. På denne måten vil grensesnittet til slutt kunne bestå av mange unødvendige funksjoner og oppleves som lite brukervennlig for «normale» brukere [21].

Når det gjelder kategorien *Gjenkjenning før erindring* har de aller fleste av de seks problemene sin årsak i at valg var lite synlige for brukerne. Det er rimelig å anta at dette skyldes at komponentenes brukergrensesnitt rett og slett er for dårlig utformet og ikke tilstrekkelig testet med representative brukere under utviklingsprosessen. Andreassen et al. [22] har blant annet funnet at det ofte

brukes lite ressurser på brukbarhetsmetoder, som for eksempel brukersentrert design, i fri programvare-prosjekter.

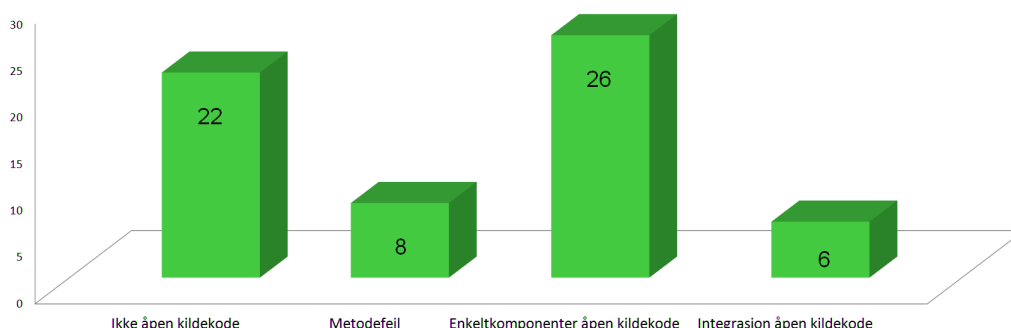
En ikke ubetydelig andel av problemene omhandler at brukerne opplever at systemet bruker andre ord og begreper enn de er vant med. I tillegg er det forskjell på brukernes og systemets oppfatning av hvordan informasjonen skal struktureres og formidles. Også her må vi legge mye av skylda på det faktum at åpen kildekode-komponentene ikke er utviklet spesielt med tanke på brukergruppens behov og tekniske ferdigheter.

Basert på diskusjonen i denne seksjonen om de ulike typene brukbarhetsproblemer som oppsto kan vi trekke ut de følgende viktigste punktene:

- Om lag halvparten av de avdekkede brukbarhetsproblemer kan knyttes direkte til bruken av åpen kildekode-komponenter.
- Det er aller mest brudd på prinsippet om å følge brukbarhetsstandarder og konvensjoner. Komponentene oppfyller ikke krav brukerne er vant med fra annen programvare.
- I tillegg synes det mye når det gjelder estetisk og minimalistisk design i systemet. Komponentene har for mye unødvendig og/eller uforståelig funksjonalitet og rotete grensesnitt.

### Brukbarhetsproblemer i komponentene

Som beskrevet i forrige seksjon har vi relatert 32 brukbarhetsproblemer til bruken av fri programvare. Av disse er 81.3% knyttet til enkeltkomponentene mens 18.7% kan knyttes mot integrasjonen av de ulike åpen kildekode-komponentene. Disse funnene er gjengitt i Figur 10.1.



**Figur 10.1:** Avdekkede brukbarhetsproblemer



Blant de 26 brukbarhetsproblemene relatert til hver enkelt åpen kildekode-komponent er det tolv problemer som skyldes forumkomponenten og samme antall skyldes kalenderen, mens kun to omhandler direkte Wordpress. Generelt kan denne skjeve fordelingen av brukbarhetsproblemer ha sin årsak i at det er stor variasjon blant komponentene hva gjelder utbredelse, community og modenhet. Wordpress er et mer populært produkt enn de to andre komponentene, med en stor utviklerstab og ressurser til å drive brukbarhetstesting under utvikling [58].

Når det gjelder vårt system er en av grunnene til få Wordpress-problemer at vi brukte Wordpress som en publiseringsløsning og mange av problemene relatert til dette gikk på hvordan innholdet var strukturert. Innholdsstrukturen er mer et resultat av våre egne designvalg (med visse føringer fra det visuelle temaet) og kan ikke lastes selve Wordpress-komponenten. Likevel er det klart at noen av kalenderproblemene er relatert til Wordpress, og man kan også se at mange av integrasjonsproblemene skyldes Wordpress sin måte å inkludere eksterne sider på. Dette betyr derimot ikke at det er feil ved Wordpress, heller et kompatibilitetsproblem.

Vi ser av Figur 10.1 at det totalt var få brukbarhetsproblemer knyttet til integrasjonen av åpen kildekode-komponentene. Spesielt gjelder dette problemer som omhandler «look and feel» og det rent visuelle ved systemet.

Åpen kildekode-komponenter og deres utviklere har ofte ulik oppfatning av brukbarhet [22] og følger få brukbarhetsstandarder [23], derfor skulle man tro at det ville oppstå flere slike problemer da vi integrerte disse. Men det at vi tok utgangspunkt i Wordpress sitt utseende og det tilhørende visuelle temaet gjorde at vi fikk et godt grunnlag for å tilpasse de andre komponentene. Både kalenderen og forumet hadde i utgangspunktet relativt nøytralt utseende og noe innstillingsmuligheter for å endre farge og stil.

Noen integrasjonsproblemer har sin årsak i måten vi inkluderte kalenderkomponenten som en egen side i Wordpress. Kalenderen var en selvstendig komponent beregnet for å kjøre som egen nettside i motsetning til forumkomponenten som fungerte som et innstikk til Wordpress. Integrasjonen av komponentene førte også til enkelte misforståelser angående hva som var privat, hva som var offentlig, hva som var offisiell informasjon og hva som var skrevet av forumbrukere. Vi innser at systemet for private meldinger burde vært tydeligere separert fra forumet slik at ingen brukere misforstår dette for å være allment tilgjengelig.

Basert på diskusjonen i denne seksjonen som angår brukbarhetsproblemer i komponentene kan vi trekke ut de følgende viktigste punktene:

- Det er en klar overvekt av brukbarhetsproblemer relatert til de enkelte åpen kildekode-komponentene.
- Wordpress har meget få problemer i forhold til de to andre komponentene.
- Det er få brukbarhetsproblemer som kan knyttes direkte til integrasjonen av komponentene.
- Samtlige komponenter hadde innstillingsmuligheter for en felles «look and feel».

### Anvendbarhet og tilfredshet

Som beskrevet i Seksjon 7.3 og 8.3 oppstod det åtte sammenbrudd, det vil si mislykkede oppgaver, i hver brukbarhetstest. Av de totalt 16 sammenbruddene kan vi relatere 13 til åpen kildekode-komponentene. Dette kan i utgangspunktet virke som et ganske høyt tall, men om vi ser litt bak tallene finner vi at hele ti av disse sammenbruddene er direkte relatert til kalenderkomponentens påminnelsesfunksjonalitet, slik den var «ut-av-boksen».

Dette kan karakteriseres som et «lokalt» problem da det er isolert til én meny og ét valg i systemet, og det er derfor rimelig å anta at å rette opp i dette problemet og gjøre påminnesvalget mer synlig burde la seg gjennomføre. Vi kan dessuten ikke være sikre på, som tidligere kommentert, at dette problemet skyldes komponenten alene. Det kan være at testdeltagerne antok at påminnelser ble lagt inn automatisk eller at de ikke oppfattet oppgaven fullstendig.

Vi kan også se på hvor stor andel av de avdekke brukbarhetsproblemene relatert til fri programvare som kan karakteriseres som kritiske, alvorlige og kosmetiske. Av de 32 problemene finner vi at fire problemer er kritiske, tolv er alvorlige og seksten kosmetiske. Dette er noenlunde lik fordeling som innenfor brukbarhetsproblemene som ikke relateres til fri programvare, og gir ingen grunn til å tro at bruk av åpen kildekode-komponenter fører til spesielt alvorlige brukbarhetsproblemer.

Den generelle tilfredsheten til systemet ble også målt under brukbarhetstestene og gir oss en pekepinn på hvor store brukbarhetsutfordringene er i dette systemet. Vi fikk en målt tilfredshet blant brukerne i brukbarhetstest I og II på henholdsvis 83.1 og 75.8. Dette kan karakteriseres som *veldig god* og *god* tilfredshet. Vi er likevel i tvil over, som vi var inne på i Seksjon 9.3, hvor mye hold det er i disse resultatene. Det var stor forskjell i alder og tekniske

ferdigheter mellom de to brukergruppene. I tillegg hadde den første gruppen personlig tilknytning til oss, mens den andre gruppen var faktiske framtidige brukere, så de kan vanskelig sammenlignes. Vi valgte derfor å legge større vekt på de kvalitative dataene og problemene vi observerte.

Basert på diskusjonen i denne seksjonen om anvendbarhet og tilfredshet kan vi trekke ut de følgende viktigste punktene:

- Det var få store, globale problemer relatert til åpen kildekode og en majoritet av sammenbruddene som oppstod under brukbarhetstestene har sin årsak i et lokalt problem i kalender-komponenten.
- Fordelingen av brukbarhetsproblemer i alvorlighetsgrad gir ingen grunn til å tro at bruk av åpen kildekode-komponenter fører til spesielt alvorlige problemer.
- Den målte tilfredsheten til systemet kan karakteriseres som *veldig god* og *god* for de to brukbarhetstestene.

### 10.1.2 Forskningsspørsmål 2

Vi vil her diskutere FS2: *Hvordan kan en redusere problemene knyttet til brukbarhet ved utvikling basert på fri programvare?*

Våre erfaringer fra deltakelsen i MOBESITY-prosjektet og utviklingen av et system basert på åpen kildekode-komponenter er hovedutgangspunktet for å kunne besvare dette spørsmålet. I tillegg baserer vi oss på relevant bakgrunns-litteratur og tidligere studier på området.

Det er viktig å understreke at våre egne erfaringer og anbefalinger baserer seg på caset vi har deltatt i, og ikke nødvendigvis kan generaliseres til andre utviklingsprosjekter.

#### Forberedelser

Et første steg for å redusere problemene knyttet til brukbarhet er å tilnærme seg det aktuelle prosjektet på riktig måte. Tar vi i betraktning ISO-definisjonen av brukbarhet [8] ser vi at det er veldig viktig å ha en klar forståelse av brukskonteksten ved utvikling av brukervennlige løsninger. Brukskonteksten inkluderer en beskrivelse av de aktuelle brukerne, oppgavene som skal gjøres, utstyret og de fysiske og sosiale omgivelsene løsningen skal brukes i. Dette er noe som ofte ikke blir gjort tilfredsstillende nok, spesielt i åpen

kildekode-prosjekter [28]. Når det mangler et klart bilde av brukskonteksten er det vanskelig å si akkurat hva sluttbrukeren trenger og ønsker, og en kan ende opp med manglende treff mellom utviklet funksjonalitet og ønsket funksjonalitet.

Vi hadde en klar definisjon av brukergruppen og denne gruppens behov og ønsker til systemet etter høstens fordypningsprosjekt og vår fremstilling av brukskonteksten kan leses i Seksjon 2.1.4. Samtidig var de ulike åpen kildekode-komponentene vi benyttet oss av utviklet for et mye større publikum. Det var derfor en utfordring å tilpasse disse til brukergruppen. Dersom vi hadde utviklet systemet fra bunnen av ville det antageligvis sett ganske annerledes ut. For eksempel om vi skulle utviklet forumet fra bunnen av ville vi gjort dette mye enklere funksjonalitetsmessig. Slik som det er nå finnes det veldig mye funksjonalitet som vi rett og slett har fjernet for brukeren da det ikke var behov for det, og fordi det var et poeng å gjøre det så enkelt som mulig. Heldigvis hadde forumet støtte for å utføre disse operasjonene gjennom administratorgrensesnittet. Kalenderen, derimot, hadde ikke slike muligheter, og vi måtte derfor inn i kildekoden og manuelt kommentere ut kode der vi ville skjule funksjonalitet og valg.

Mens forumet i prinsippet hadde passende struktur og utseende, var kalenderen ikke optimal for vårt bruk. Vi ville fokusert på å gjøre den enklere å bruke, og fokusert på ulike typer oppføringer, for å gjøre den mer passende til en personlig *min plan*. For eksempel ville vi likt å ha ulike oppføringstyper, slik at en kan både legge inn varslinger og også oppføringer av typen treningsdagboknotater og andre dagbokoppføringer. Da vi valgte komponenter hadde vi de overnevnte betraktningene i bakhodet, men et begrenset utvalg gjorde at kalenderen vi valgte var det beste alternativet vi kom over.

Basert på diskusjonen i denne seksjonen kan vi trekke ut de følgende viktigste poengene:

- Det er viktig å ha en klar forståelse av systemets brukskontekst, som inkluderer brukere, oppgaver, omgivelser og utstyr.
- Gjennomfør en evaluering av aktuelle komponenter opp i mot brukskontekst.

### **Valg av komponenter**

Ved valg av komponenter tok vi selvfølgelig brukernes behov i betraktning, i tillegg til at vi så på mulighetene for å tilpasse brukergrensesnittet til å

ha en felles gjennomgående stil. Som nevnt tidligere ble det lagt føringer på at Wordpress var komponenten å basere seg på, men vi valgte ikke Wordpress kun på bakgrunn av dette. Wordpress hadde den funksjonaliteten som var ønsket, i tillegg til mange innstikk, temaer og modifiseringsmuligheter. Samtidig er Wordpress en anerkjent plattform med et stort community.

Hoffmann og Cook [36] påpeker at det er en fordel å bruke én komponent som plattform og som en slags stilveiledning når man setter sammen et system bestående av ferdigkomponenter. Siden vi var mest bundet til Wordpress-komponenten og det tilhørende visuelle temaet ble de andre komponentene evaluert opp mot dette med det overordnede målet om å oppnå et konsistent grensesnitt og enklest mulig integrasjon.

En faktor vi burde lagt mer vekt på ved valg av komponenter var arkitekturen og kodestrukturen til komponentene. Dette er noe vi ser stor nytte av og som vi tar lærdom av fra dette prosjektet. Vi burde sett på i hvilken grad det var mulig å modifisere arkitekturen og koden for å tilpasse funksjonalitet og utseende til å matche brukergruppen. Gjennom prosjektets tidsløp ble vi kjent med komponentenes oppbygning og kodekvalitet, og det er trygt å si at vi ville sett mer etter alternativer til kalenderløsningen vi endte opp med. WebCalendar hadde noe uferdig kode og komplisert og vanskelig oppbygning som gjorde den unødvendig problematisk å modifisere.

Et uttrykk som blir brukt i programvareutvikling er «Kill your darlings» [59]. Betydningen er enkel; Ikke bli for knyttet til koden din. Dette er også relevant for bruk av åpen kildekode-komponenter i utvikling. Det kan fort bli en fallgrube å ikke gi slipp på en komponent en har valgt, fordi en har lagt mye tid og ressurser i å få til noe. En kan derfor ignorere andre muligheter som kanskje er bedre. Vi opplevde selv dette dilemmaet i vårt valgt av forumløsning. Vi fikk satt opp forumet ganske raskt, men da vi opplevde problemer med integreringen av forumet med plattformen vår, så vi etter andre, enklere, alternativer. Kanskje burde vi også sett etter alternativer til de andre komponentene underveis, før vi ble for avhengige av løsningene vi valgte.

I etterkant ser vi at vi kanskje burde gjort en grundigere undersøkelse av komponentene vi tok i bruk før vi satte i gang integreringen av disse. Som beskrevet i Kapittel 4 fikk vi problemer med noe av integrasjonen, og ble for eksempel nødt til å bytte ut phpBB med et forum-innstikk til Wordpress. I tillegg var enkelte komponenter i overkant avanserte til vårt bruk, med mye unødvendig funksjonalitet i forhold til vår målgruppe, noe som gikk ut over brukbarheten.

Basert på diskusjonen i denne seksjonen kan vi trekke ut de følgende viktigste punktene:

- Evaluer de aktuelle komponentenes brukbarhet, arkitektur, modifiserbarhet, utbredelse, dokumentasjon og community ved utvelgelse.
- Bruk én komponent som plattform og stilveiledning ved evaluering av andre aktuelle komponenter.
- Ikke bli for knyttet til egen kode og komponenter, «Kill your darlings».

### Utviklingsmetode

Valg av utviklingsmetode er også viktig når man skal utvikle et system basert på ferdigkomponenter og vil unngå brukbarhetsproblemer. Tidligere studier viser at vannfalls-metoden er lite egnet i slike prosjekter [33, 34], blant annet fordi den baserer seg på at kravene bestemmer hva som leveres i det ferdige systemet. Bruk av ferdigkomponenter, både åpen kildekode og kommersielle, vil ofte legge føringer på hvilke krav som kan leveres. Derfor er det nødvendig å bruke en mer passende og fleksibel utviklingsmetode som tar høyde for at krav kan diskuteres.

Et annet viktig poeng er å legge til rette for tidlig integrering av komponenter i utviklingsløpet, da dette ofte er den mest usikre og risikofylte prosessen i slike prosjekter [33]. Dette gjelder ikke spesielt for å redusere brukbarhetsproblemer, men det er klart at å få opp et vellykket integrert system på et tidlig tidspunkt vil tillate mer evaluering, testing og forbedring av brukbarhet.

Under vår utviklingsprosess fokuserte vi på å få opp et fullt kjørende system så raskt som mulig for å kunne gå i gang med å utbedre brukbarheten. I et system bestående av ulike ferdigkomponenter vil det være behov for å gjøre utseendetilpasninger for å oppnå en helhetlig visuell profil før man gjør en fullstendig brukbarhetsevaluering. Til forskjell fra de aller fleste kommersielle ferdigkomponenter muliggjør fri programvare å gjøre endringer i arkitektur og kildekode for å redusere brukbarhetsproblemer. Dette er en klar fordel med åpen kildekode, men også en risiko, da kodeendringer kan ha uante ringvirkninger hvis ikke de gjøres riktig. Samtidig er det også stor variasjon mellom komponenter hva gjelder kodekvalitet og modifiserbarhet som påvirker hvor enkelt det er å gjøre slike endringer. Det vil ofte være en sammenheng mellom graden av modifiserbarhet og komponentenes community og utbredelse. Erfaringer fra vår utvikling viser at Wordpress la mer til rette for egne tilpasninger av kildekode enn de to andre komponentene.

Basert på diskusjonen i denne seksjonen kan vi trekke ut de følgende viktigste punktene:

- Bruk en passende utviklingsmetode som tillater fleksible krav.
- Gjennomfør integrering av komponentene tidlig i utviklingsløpet.
- Åpen kildekode muliggjør at brukbarhetsproblemer kan utbedres ved modifisering av kildekoden.

### **Brukbarhetsevaluering og testing**

Med utgangspunkt i Nielsens ti punkter [10] og Shneidermanns åtte gyldne regler [11] gjennomførte vi tidlig en rask gjennomgang av grensesnittet i vårt system. På denne måten identifiserte vi de mest synlige brukbarhetsproblemene som for eksempel inkonsistens i språk og fargebruk og kunne luke vekk disse. Antageligvis utbedret vi også flere lokale problemer som kanskje ikke ville blitt avdekket under en brukbarhetstest.

På dette stadiet hadde vi et første-utkast til et system med et noenlunde helhetlig utseende og den viktigste funksjonaliteten på plass. Vi så derfor muligheten for å gjennomføre en tidlig brukbarhetstest for å få avdekket ytterligere problemer med brukergrensesnittet og få tilbakemeldinger på dette til videre utvikling og redesign. Deltagerne til denne testen var i stor grad rekruttert blant våre medstudenter på datateknikk og hadde god kunnskap om brukbarhet og erfaring i å utforme lignende løsninger. Brukbarhetstesten fungerte derfor som en slags kombinasjon av en ekspertvurdering og en brukertest og gjorde at vi fikk avdekket enda flere problemer på et tidlig tidspunkt.

Terry et al. [25] foreslår å benytte seg av lignende metoder under utvikling av fri programvare hvor man ofte mangler ressurser til å gjennomføre offisielle brukbarhetstester med representative brukere. Ved å utnytte kunnskapsrike brukere med mye erfaring i å bruke programvare kan man få evaluert programvarens brukbarhet på et generelt nivå før endelige utgivelser. Risikoen med både denne metoden og vår egen tidlige brukbarhetstest er at grensesnittet blir mer tilpasset disse brukerne enn de egentlige brukerne, og at man avdekker falske problemer.

Vi følte likevel at dataen fra den første brukbarhetstesten var relevante og samsvarte på mange måter med vår egen oppfatning og evaluering av nettsiden med tanke på brukbarhet. På grunnlag av dette gjorde vi derfor en redesign av siden og satte i gang videreutvikling fram mot brukbarhetstest II med representative, framtidige brukere av systemet.

Basert på diskusjonen i denne seksjonen kan vi trekke ut de følgende viktigste punktene:

- Gjør tidlig heuristisk evaluering eller enkel gjennomgang av komponentenes brukbarhet.
- Gjennomfør tidlig brukbarhetstest eller ekspertvurdering etter integrering, dersom ressurser og tid strekker til.
- Det kan være gunstig med flere brukbarhetstester med få brukere på ulike stadier i utviklingsprosessen.

## 10.2 Resultatkvalitet

I denne seksjonen vil vi diskutere kvaliteten til resultatene vi kom fram til gjennom den utførte praktiske studien. Vi vil diskutere resultatenes validitet, generaliserbarhet og reliabilitet.

For vår del kan en tenke seg to aspekter av resultatkvalitet; Det første aspektet er i hvilken grad resultatene fra brukbarhetstestene har avdekket majoriteten av problemene i systemet, og hvorvidt disse resultatene er valide, generaliserbare og reliable. Det andre aspektet er hvorvidt resultatene fra denne studien sett under ett, ut i fra problemstillingen, er valide, generaliserbare og reliable.

### 10.2.1 Validitet

Validiteten til et studie sier i hvilken grad våre konklusjoner basert på resultatene er gyldige. Vi tar her for oss indre validitet, eller kredibilitet, som eksisterer om vi kan begrunne at for eksempel hypotesen A er den riktige grunnen til en handling B [37], og ikke påvirket av andre faktorer. Hughes [60] sier at en brukbarhetstest er kredibel dersom testscenarioene reflekterer brukerens realistiske bruk av produktet.

#### Brukbarhetstesten

Vi forsøkte det vi kunne for å gjenskape et realistisk bruksscenario. Oppgavene vi utformet tror vi var realistiske, og kunne vært typiske bruksområder



i en hverdagslig situasjon. Selv om et par av oppgavene kunne vært bedre formulert tror vi ikke dette spilte negativt inn på validiteten til testene.

Under begge brukbarhetstestene var det mange spesifikke problemer og problemområder som gjentok seg. Dette kan tyde på at antallet brukere vi benyttet oss av var tilfredsstillende. Det kan også kommenteres at i brukbarhetstest II fant den siste testpersonen kun to unike problem, hvor det kan argumenteres for at ett av disse var et falskt problem.

Gjennomføringene av brukbarhetstestene fokuserte på kvalitative data, slik som «Tenke høyt-prinsippet» og oppfølgingsintervju. På denne måten fikk vi avdekket problemene utover at testpersonen ikke klarte en oppgave. Dette øker kredibiliteten til de brukbarhetsfunnene vi fant. Nielsen [61] og Agyris et al. [62] er enige om at det å få brukeren til å snakke høyt kan være den viktigste metoden for å avdekke problemer. På denne måten blir problemene grunnfestet i brukernes tilbakemelding og synspunkter.

### Studien under ett

Det vi forsøker å måle gjennom dette systemutviklingscaset er hvilke brukbarhetsproblemer som oppstår ved bruk av åpen kildekode-komponenter, og hvordan disse kan reduseres. Siden brukbarhetstestingen i stor grad ligger til grunn for denne diskusjonen, vil den overnevnte diskusjonen også gjelde for studien sett under ett. Det er likevel andre faktorer som også spiller inn.

Komponentene vi valgte å integrere trenger ikke nødvendigvis være representative for alle åpen kildekode-komponenter. Det var et tydelig skille mellom de ulike komponentene både hva gjelder brukergrensesnitt, arkitektur og kodekvalitet, spesielt fremstilt i Seksjon 4.5.1. Det er derfor ikke mulig å si om komponentene vi valgte er representative for fri programvare sett under ett.

Det påvirker også validiteten at prosjektet, og da spesielt utviklingen, er utført av studenter. Profesjonelle systemutviklere ville nok hatt en annen tilnærming og oppnådd et annet resultat. Men selv innenfor dette miljøet vil det være ulike måter å utvikle på og ulike tilnærminger til brukbarhetsheving. Det ville vært nyttig å undersøke hvordan andre prosjekter har angrepet lignende prosjekt, og undersøke hvilke tiltak de har gjennomført for å redusere brukbarhetsproblemer. Ikke minst hvordan de har lyktes med å sette sammen en endelig løsning med et helhetlig og anvendbart brukergrensesnitt.

## 10.2.2 Generaliserbarhet

Generaliserbarhet, eller ytre validitet, sier noe om resultatene fra en studie med et mindre utvalg mennesker kan generaliseres til å være gyldige for andre tilfeller og et større antall mennesker.

### Brukbarhetstestene

Vi fikk et greit antall testpersoner til å kunne avdekke majoriteten av brukbarhetsproblemene, selv om vi ideelt sett ønsket flere. Nielsen [44] argumenterer for at 5-6 personer er nok til å avdekke en stor majoritet av brukbarhetsproblemene. Andre studier [46, 47] har likevel vist at et større antall testpersoner er ønskelig.

Ut ifra et systemutviklingsperspektiv var antall testpersoner relativt bra. Vi kunne ønsket oss flere testpersoner for å kunne dra gyldigere konklusjoner ut i fra dataene. Seks testpersoner i brukbarhetstest II er ikke nok til å kunne generalisere resultatene med en viss sikkerhet. Med flere deltagere ville vi avdekket flere problemer, og våre resultater ville vært et bedre grunnlag for å svare på problemstillingen.

Vi har grunn til å trekke i tvil om resultatene fra vår første brukbarhetstest er helt generaliserbare. Som vi har vært inne på flere ganger tidligere, så er ikke personene fra brukbarhetstest I hverken representative eller realistiske brukere. Dette kan ha påvirket resultatet vi fikk, og kan ha gitt utslag i både antall problemer funnet og dekningsgraden av problemer funnet. Flere faktorer kan trekkes inn som setter resultatene fra brukbarhetstest I i tvil; Testpersonenes lave alder, tekniske kyndighet og personlige tilknytning til prosjektgruppen kan alle ha påvirket resultatene.

Under innsamlingen av data var ikke situasjonen, eller «settingen», slik den ville vært ved reell bruk. Utstyret, altså datamaskinen med tilbehør, skiller seg ikke ut fra en typisk hjemme-pc, men en kommer ikke vekk ifra at konteksten ved en brukbarhetslab med en fasilitator til stede ikke er realistisk. Det er vanskelig å si noe om hvorvidt dette har påvirket resultatene vi fikk. I et slikt kunstig bruksscenario vil ofte brukeren oppføre seg forskjellig fra virkelig bruk. Likevel peker studier på at det ofte ikke er verdt å utføre en felt-studie i forhold til en laboratorietest under evaluering av brukbarhet [63].

Opgavene som ble gitt i brukertestene kan også ha spilt inn på dekningsgraden til problemfunnene. Det er mulig oppgavene burde vært gjort på en

## 10.2. RESULTATKVALITET

---

annen måte, og vi er ikke sikre på om de favner over all funksjonalitet eller bruksområder. Det er derfor mulig at noen av problemområdene vil gå upåaktet hen, selv om oppgavene for oss virket meget realistiske.

### Studien under ett

Det er vanskelig å si om de resultatene og konklusjonene vi har kommet fram til gjennom denne praktiske studien kan generaliseres til å gjelde i en større skala. For brukbarhetsproblemene vi har funnet vil vi tro at de kan generaliseres til å være gyldige for denne brukergruppen i sin helhet, og kanskje andre pasientgrupper og marginale grupper i en lignende situasjon.

Det er derimot vanskelig å si noe om hvorvidt våre erfaringer med bruk av åpen kildekode-komponenter kan generaliseres for å gjelde alle slike utviklingsprosjekter. Våre konklusjoner og erfaringer er trukket ut ifra dette systemutviklingscaset, med de begrensninger som gjør seg gjeldende. Det er for eksempel naturlig å tro at et profesjonelt programvareutviklingsfirma ville tilnærmet seg utviklingsprosjektet annerledes enn oss, og sannsynligvis oppnådd et annet resultat. I tillegg er det stor varians når det gjelder komponentene vi har brukt, og våre resultater er i stor grad preget av enkeltkomponenters egenskaper. Sett ut ifra vårt perspektiv tror vi likevel vi har avdekket mange av de utfordringene som oppstår ved bruk av åpen kildekode-komponenter i mindre utviklingsprosjekter. Vi tror også de ulike tiltakene vi har identifisert for å redusere brukbarhetsproblem vil være gyldige for andre, lignende, prosjekt.

### 10.2.3 Reliabilitet

Reliabiliteten til et studie sier noe om hvorvidt en ville fått de samme resultatene dersom en gjentok studiet flere ganger. Ved en slik gjenskapelse bør en bruke den samme brukergruppen, testmetodikk og ellers ideelt sett helt identiske omstendigheter [37].

### Brukbarhetstestene

For brukbarhetstestene, spesielt brukbarhetstest II, tror vi reliabiliteten var rimelig god. Vi brukte like oppgaver gjennom hver iterasjon, og i brukbarhetstest II brukte vi også en fast fasilitator. Vi så også at mot slutten av testen

var det relativt få unike problemer, og vi tror vi ville avdekket omtrent de samme problemene om vi skulle utført brukbarhetstestene på nytt.

### **Studien under ett**

Dersom vi hadde gjentatt dette prosjektet med de samme forutsetningene og begrensningene er det fullt mulig vi ville endt opp med en annen endelig løsning. Tidlig i prosjektet hadde vi liten kjennskap til de ulike åpen kildekode-komponentene, og vi kunne endt opp med andre komponenter ved tilfeldigheter. Våre funn tilsier at valg av komponenter kan påvirke det endelige resultatet i stor grad.

Andre personer med annen bakgrunn innenfor datateknikk kunne ha tilnærmet seg problemstillingen på andre måter, og oppnådd andre resultat. Dersom en av prosjektmedlemmene hadde hatt erfaring med en annen plattform enn Wordpress, ville sannsynligheten vært stor for at denne komponenten heller ble valgt, og dette ville påvirket den endelige løsningen betraktelig.

### **10.2.4 Falske problemer**

Ofte vil metoder for å evaluere brukbarheten til et system avdekke problemer som ikke er problemer ved vanlig bruk, såkalt falske problemer [56]. I statistikken kalles dette type I-feil eller falske positive, det vil si at man observerer en forskjell når det egentlig ikke er noen [64].

Slike problemer kan oppstå både i forbindelse med heuristisk evaluering og brukbarhetstesting i laboratorium og det er vanskelig å si noe klart om hvilken metode som genererer flest falske problemer. Det er likevel verdt å ha dette fenomenet i tankene når man diskuterer resultat kvaliteten og analyserer brukbarhetsproblemene.

Generelt avdekker brukbarhetstesting ofte flere problemer enn andre evalueringsmetoder og det er naturlig å spørre seg om det høye antallet problemer kan ha sin årsak i falske problemer [63]. Slik testing vil aldri bli helt realistisk, og så lenge konteksten er konstruert vil det være en risiko for at resultatene er falske positive. Likevel er det grunn til å tro at en brukbarhetstest med representative brukere vil avdekke ekte problemer i større grad enn for eksempel en heurtisk evaluering av brukergrensesnittet [65].

Den manglende realismen i konteksten av vår brukbarhetstesting kan ha gitt utslag i enkelte falske problemer. Antagelig er et system av denne typen

noe brukeren vil benytte seg av hjemme, foran sin egen pc, uten spesielt tidspres eller noen som ser på. Det at fasilitator er tilstede og følger med på hva brukeren gjør under oppgavene i testingen kan gjøre at brukeren opplever et visst press og falske problemer kan ha oppstått som følge av dette.

Enkelte ganger under testingen fikk vi også kommentarer på at testpersonen visste hva informasjonen de fikk i oppgave å søke opp dreide seg om. Dersom brukeren mislyktes i å lete fram denne informasjonen i systemet kan man ikke kalle dette et reelt problem. Det er alltid en risiko for slike tilfeller ved utforming av denne typen oppgaver til en brukbarhetstest. Problemer som oppstod som følge av at testpersonen skrev feil er et resultat av konteksten og bør heller ikke legges vekt på. Det samme gjelder også for andre metodefeil, for eksempel uklare formuleringer i oppgavene.

## 10.3 Metodediskusjon

I denne seksjonen vil vi vurdere gjennomføringen og nytteverdien av forskningsmetodene som ble brukt i dette prosjektet. Hovedforskningsmetoden i dette prosjektet var et systemutviklingscase hvor vi deltok i forskningsprosjektet MOBESITY og utviklet et egenomsorgssystem for overvektspasienter basert på åpen kildekode-komponenter. Som en del av dette caset gjennomførte vi også brukbarhetstesting av systemet hvor vi tok i bruk intervju og spørreskjema.

### 10.3.1 Systemutviklingscase

Systemutviklingscaset ga oss verdifulle erfaringer i å sette sammen åpen kildekode-komponenter til et ferdig system. Vi fikk brukt vår kunnskap om brukbarhet til å tilpasse disse slik at de dannet et helhetlig system visuelt sett. Våre tekniske ferdigheter kom godt med til å modifisere komponentene og utbedre brukbarhetsproblemer og vi fikk god innsikt i hvordan en kan redusere slike problemer.

Systemet ble utviklet i iterasjoner med to brukbarhetstester som satte frister. Vi fokuserte på å få opp et fullt kjørende system så raskt som mulig, med alle åpen kildekode-komponentene integrert. Integrasjon er ofte det mest usikre og risikofylte i et prosjekt basert på ferdigkomponenter [33], og selv om dette var et åpen kildekode-prosjekt og vi hadde mer kontroll over komponentene var det viktig å få avdekket integrasjonsproblemer så tidlig som mulig. Dette

gjorde blant annet at forum-komponenten ble byttet ut på et tidlig stadium i prosessen og skapte få problemer.

Det eneste av stor funksjonalitet som var planlagt, men som vi ikke fikk tid til å implementere, var dagbok-delen av nettsiden. Utover dette gikk gjennomføringen av utviklingsprosessen etter planen og vi hadde systemet ferdigstilt til testing for de to brukbarhetstestene som ble planlagt. Med unntak av dagbok-komponenten var alle de store funksjonalitetskravene innfridd. Dette var viktig for å kunne gi oss relevante brukbarhetstestresultater og innblikk i brukbarhetsutfordringer som skyldes åpen kildekode.

Til slutt er det også verdt å nevne at deltagelsen i dette utviklingsprosjektet ga oss kunnskap og ferdigheter i ny teknologi, som PHP og oppsett og drift av web-tjener og nettside.

### 10.3.2 Brukbarhetstesting

Brukbarhetstestene hadde et todelt formål i dette prosjektet. For det første var det en viktig del av systemutviklingsprosessen for å avdekke problemer og få tilbakemelding til redesign og videreutvikling av systemet. På den andre siden var det en del av forskningsmetoden og ga oss en mengde data og resultater for analyse og diskusjon av forskningsspørsmålene.

Selve forberedelsene og gjennomføringen av brukbarhetstestene gikk stort sett fint. Vi kan likevel nevne enkelte utfordringer og problemer vi støttest på underveis som er av betydning for prosjektet.

Før den andre brukbarhetstesten hadde doktogradstudenten som jobber med MOBESITY-prosjektet ansvar for å rekruttere deltagere til testen fra pasientgruppen. Det var vanskelig å rekruttere mange til dette og vi endte til slutt opp med seks testpersoner til å teste systemet. Vi både ønsket og hadde ressurser til å gjennomføre flere tester, noe som ville gitt gyldigere resultater å basere oss på for å svare på forskningsspørsmålene. Sett ut i fra systemutviklingscasets perspektiv var det likevel et tilstrekkelig antall til å avdekke majoriteten av brukbarhetsproblemene, noe som støttes av tidligere studier [44].

En annen viktig del av forberedelsene var å utarbeide oppgavene som testpersonene skulle gjennomføre for å få testet alle de viktige delene av systemet. Disse fungerte etter planen, men vi så oss nødt til å gjøre noen endringer mellom de to testene på grunn av at enkelte deloppgaver ble for spesifikke

eller var formulert uklart. De øvrige praktiske forberedelsene med å sette opp brukere i systemet og legge inn testdata foregikk uten store problemer.

Et stort problem underveis var forbundet med eye-trackingen. Først hadde vi problemer med å fange opp øynene til enkelte deltagere og kalibreringen fungerte dårlig. Vi fant også ut etter hvert at programvare som tok opp eye-tracking-dataene reduserte ytelsen til testmaskinen betraktelig og førte blant annet til at tastaturet ikke oppfattet alle tastetrykk. Til slutt så vi oss nødt til å stoppe all eye-tracking da det hadde for stor påvirkning på testen. Utover dette fungerte lokasjonen og det tekniske oppsettet bra, med unntak av små problemer med nettleserinnstillingene på testmaskinen.

Som vi var inne på ga brukbarhetstestene oss mye data, hvor vi har valgt å fokusere spesielt på de kvalitative dataene. Det er også naturlig å gjøre dette siden brukbarhetstestene ble utført med relativt få testpersoner. I så måte var videopptakene av testingen og intervjuet i etterkant veldig nyttig under vårt arbeid for avdekke og analysere brukbarhetsproblemer og vi fikk mye ut av nettopp dette.

#### 10.3.3 Intervju

Oppfølgingsintervjuene av testdeltagerne som ble avholdt etter hver brukbarhetstest fungerte godt for å få dypere innsikt i deltagerens inntrykk av systemet. Deltagerne kom med kommentarer til problemene de hadde og ga oss en bedre forståelse av årsakene til problemene. På denne måten fungerte intervjuene som et supplement til de andre dataene fra brukbarhetstesting slik at vi bedre kunne analysere problemene. Vi fikk også mange verdifulle innspill i forhold til endringer i utseendet og innholdsstrukturen på nettsiden, noe som var spesielt nyttig for videreutviklingen som ble gjort etter den første brukbarhetstesten.

#### 10.3.4 Spørreundersøkelse

De to spørreundersøkelsene som ble gjennomført, bakgrunnsskjema og SUS-skjema, fungerte uten problemer. Bakgrunnsskjemaet var meget nyttig for å kartlegge og definere brukerne i de to brukbarhetstestene. Det bekreftet vår mistanke om at det var store forskjeller mellom gruppene og at det derfor var vanskelig å sammenligne resultatene fra testene. SUS-skjemaet ga oss et godt inntrykk av brukergruppenes tilfredshet med systemet og gjør at vi kan argumentere for at systemet hadde tilstrekkelig brukbarhet.





# Kapittel 11

## Konklusjon

I dette kapitlet vil vi oppsummere arbeidet som er gjort og konkludere svarene på problemstillingen og forskningsspørsmålene, samt se på mulighetene for videre arbeid med prosjektet og temaet.

### 11.1 Problemstilling

Hovedmålet med dette prosjektet var å se på *hvilke utfordringer knyttet til brukbarhet som kan oppstå ved bruk av fri programvare i utviklingsprosjekter, og hvordan disse kan reduseres.*

For å svare på denne problemstillingen ble det gjennomført en praktisk studie hvor vi utviklet et system for overvektspasienter basert på åpen kildekodekomponenter. Videre ble dette systemet brukbarhetstestet og funn fra disse testene ble analysert og diskutert.

Den kommende besvarelsen av forskningsspørsmål 1 og 2 utgjør til sammen svaret på problemstillingen. Oppsummert fant vi at de fleste utfordringene er knyttet til konsistens og standarder innenfor unike åpen kildekodekomponenter, samt at designet kunne vært mer estetisk og minimalistisk. For å redusere disse utfordringene er det hensiktsmessig å gjennomføre en vurdering og integrasjon av komponentene på et tidlig stadium, for å avdekke risikomomenter og svakheter. Slik vil en kunne sette i gang fokuset på brukbarhet tidlig i utviklingsprosessen.

### 11.1.1 Forskningsspørsmål 1

*Hvilke brukbarhetsutfordringer fører utvikling basert på fri programvare med seg?*

For å undersøke dette spørsmålet kategoriserte vi alle problemene som ble avdekket under brukbarhetstestene opp mot etablerte brukbarhetsheuristikker og så på hvilke som kunne knyttes til bruk av fri programvare. Ut i fra dette fant vi at om lag halvparten av de avdekkede brukbarhetsproblemer kunne knyttes direkte til bruken av åpen kildekode-komponenter. Blant disse problemene var det oftest brudd på prinsippet om å følge brukbarhetsstandarder og konvensjoner. Dette skyldes stort sett at komponentene ikke oppfyller krav brukerne er vant med fra annen programvare. I tillegg synes det mye når det gjelder estetisk og minimalistisk design i systemet da brukerne opplever at komponentene har for mye unødvendig og/eller uforståelig funksjonalitet.

Det er en klar overvekt av brukbarhetsproblemer relatert til de enkelte åpen kildekode-komponentene. Wordpress har meget få problemer i forhold til kalenderen og forumet. Samtidig er det få brukbarhetsproblemer som kan knyttes direkte til integrasjonen av komponentene.

Alt i alt var det få store, globale problemer relatert til bruk av åpen kildekode-komponenter. Vi ser det som positivt at majoriteten av sammenbruddene som oppstod under brukbarhetstestene har sin årsak i et lokalt problem i kalender-komponenten. Fordelingen av brukbarhetsproblemer i alvorlighetsgrad gir ingen grunn til å tro at bruk av åpen kildekode-komponenter fører til spesielt alvorlige problemer. Samtidig kan den målte tilfredsheten til systemet karakteriseres som *veldig god* og *god* for de to brukbarhetstestene.

Selv om tilfredsheten og anvendbarheten til den endelige løsningen viser seg å være på et akseptabelt nivå, er det også verdt å merke seg at det endelige brukergrensesnittet ikke skiller seg ut som spesielt helhetlig eller perfekt på noen måte. Dette er en ulempe ved å bruke fri programvare da en har begrensede muligheter til å endre brukergrensesnittet og skape en helhetlig «look and feel». En må derfor akseptere at brukergrensesnittet ikke er så pent som en ønsket, og heller glede seg over de mange fordelene denne typen utvikling fører med seg, og at brukbarheten tross alt er *bra nok*.

### 11.1.2 Forskningsspørsmål 2

*Hvordan kan en redusere problemene knyttet til brukbarhet ved utvikling basert på fri programvare?*

Gjennom våre erfaringer fra dette prosjektet kan vi identifisere en rekke tiltak for å redusere brukbarhetsproblemer i utvikling basert på åpen kildekode.

Det er viktig å ta i betraktning ISO-definisjonen av brukbarhet for å definere riktig brukskontekst. Aktuelle åpen kildekode-komponenter bør evalueres opp mot denne. I tillegg bør komponentenes arkitektur, modifiserbarhet, utbredelse, dokumentasjon og community vurderes ved utvalgelse. Dersom det er hensiktsmessig bør én komponent velges som plattform og andre komponenter evalueres mot denne med tanke på grafisk grensesnitt og integrerbarhet. En annen mulighet er å bruke en felles stilveiledning som alle aktuelle komponenter evalueres opp mot.

Under utvikling bør det være fokus på å integrere komponentene på et tidlig stadium for å minimere risiko og belyse problemområder. Dette muliggjør at man raskt kan ta i bruk ulike metoder for å øke brukbarheten til systemet. Både ekspertbrukere og representative brukere kan involveres for å gjennomføre flere iterasjoner med heuristiske evalueringer og brukbarhetstester. Det vil derfor være naturlig å benytte seg av en iterativ utviklingsmodell i denne type prosjekter, som tillater å holde kravene så fleksible som mulig, da ferdigkomponenter ofte vil påvirke systemkravene.

Bruk av åpen kildekode tilbyr større muligheter for å redusere og utbedre brukbarhetsproblemer gjennom å utføre direkte endringer av komponenters arkitektur og kildekode. Dette er en klar fordel med fri programvare, men graden av modifiserbarhet varierer ofte over ulike åpen kildekode-komponenter.

De overnevnte tiltakene for de ulike fasene av utviklingsprosessen vil alle bidra til å minimere brukbarhetsproblemer ved utvikling basert på fri programvare. Spesielt vil vi trekke fram viktigheten av å utføre en tidlig vurdering og integrering av aktuelle åpen kildekode-komponenter for å oppnå et tilstrekkelig nivå av brukbarhet.

## 11.2 Videre arbeid

Videre i dette prosjektet hadde det vært nyttig å gjennomføre en ny brukbarhetstest etter at problemene er utbedret og utviklingen av systemet er fullført. En slik test bør gjøres med flere testpersoner enn det som var tilfelle under forrige gjennomføring, for å gi gyldigere resultat. I tillegg bør helsepersonell involveres i en slik test for å avdekke problemer relatert til deres bruk av systemet. Det hadde vært interessant å gjennomføre en grundig, heuristisk evaluering av grensesnittet med brukbarhetsekspertene for å se hvilke problemer de ville funnet. Andre forskningsmetoder, som for eksempel intervjuer og feltstudier, kan også tas i bruk som et supplement eller alternativ til brukbarhetsevalueringer.

Vi ser klart at det er behov for mer forskning på temaet som er behandlet i denne oppgaven. Det kunne vært interessant å ta i betraktning andre tilsvarende utviklingsprosjekter av samme størrelse, samt større prosjekter i industrien, med tanke på brukbarhet og benyttelse av fri programvare. Disse prosjektene ville benyttet seg av andre åpen kildekode-komponenter og utviklingsmetoder, og opparbeidet seg annen erfaring. Som et naturlig neste steg innenfor dette temaet vil vi foreslå å gjennomføre en empirisk studie av slike prosjekter for å undersøke hvordan «state-of-the-practice» fokuserer på brukbarhet.

# Bibliografi

- [1] Bekk. Bekk open source. <http://boss.bekk.no/display/BOSS/Bekk+Open+Source>. Sist besøkt 10. Mai, 2010.
- [2] Accenture. Open source solutions. [http://www.accenture.com/Global/Technology/Open\\_Source/default.htm](http://www.accenture.com/Global/Technology/Open_Source/default.htm). Sist besøkt 10. Mai, 2010.
- [3] C. Benson. Meeting the challenge of open source usability. *Interfaces*, pages 9–12, 2004.
- [4] Wikimedia. Wikipedia usability initiative. [http://usability.wikimedia.org/wiki/Wikipedia\\_Usability\\_Initiative](http://usability.wikimedia.org/wiki/Wikipedia_Usability_Initiative), 2010. Sist besøkt 03. Mai, 2010.
- [5] Peter Brown. Chris hofstader appointed as gnu access technology director. <http://www.fsf.org/news/chris-hofstader-gnu-access-technology-director>, 2010. Sist besøkt 12. Mai, 2010.
- [6] Free Software Foundation. Gnu operating system. <http://www.gnu.org/>. Sist besøkt 12. Mai, 2010.
- [7] Anita Das. Kravspesifikasjon mobesity. Part of Ph.D., NTNU, 2009.
- [8] International Organization for Standardization. Iso 9241-11. <http://www.it.uu.se/edu/course/homepage/acsd/vt09/ISO9241part11.pdf>, 1998. Sist besøkt 06. Mai, 2010.
- [9] J. Nielsen. Usability 101: Introduction to usability. *Jakob Nielsen's Alertbox*, 2003.
- [10] J. Nielsen. Ten usability heuristics, 1994.

- [11] Ben Shneiderman. *Designing the user interface: strategies for effective human-computer interaction*. Addison-Wesley Longman Publishing Co., Inc., 2009.
- [12] Open Source Initiative. The open source definition. <http://www.opensource.org/docs/osd>. Sist besøkt 23. April, 2010.
- [13] Wikipedia. Fri programvare. [http://no.wikipedia.org/wiki/Fri\\_programvare](http://no.wikipedia.org/wiki/Fri_programvare). Sist besøkt 23. April, 2010.
- [14] Free Software Foundation. The free software definition. <http://www.gnu.org/philosophy/free-sw.html>. Sist besøkt 23. April, 2010.
- [15] Wikipedia. Usage share of web browsers. [http://en.wikipedia.org/wiki/Usage\\_share\\_of\\_web\\_browsers](http://en.wikipedia.org/wiki/Usage_share_of_web_browsers). Sist besøkt 26. April, 2010.
- [16] Netcraft. April 2010 web server survey. [http://news.netcraft.com/archives/2010/04/15/april\\_2010\\_web\\_server\\_survey.html](http://news.netcraft.com/archives/2010/04/15/april_2010_web_server_survey.html). Sist besøkt 28. April, 2010.
- [17] The Standish Group. Free open source software is costing vendors \$60 billion. [http://standishgroup.com/newsroom/open\\_source.php](http://standishgroup.com/newsroom/open_source.php). Sist besøkt 26. April, 2010.
- [18] B. Fitzgerald. The transformation of open source software. *Management Information Systems Quarterly*, 30(3):587, 2006.
- [19] N. Viorres, P. Xenofon, M. Stavarakis, E. Vlachogiannis, P. Koutsabasis, and J. Darzentas. Major HCI challenges for open source software adoption and development. *Online Communities and Social Computing*, pages 455–464.
- [20] Henrik Hedberg and Netta Iivari. Integrating HCI Specialists into Open Source Software Development Projects. 2009.
- [21] D.M. Nichols and M.B. Twidale. The usability of open source software. *First Monday*, 8(1):21, 2003.
- [22] M.S. Andreasen, H.V. Nielsen, S. Schrøder, and J. Stage. Usability in open source software development: opinions and practice. *Information technology and control*, 25(3A):303–12, 2006.
- [23] N. Frishberg, A.M. Dirks, C. Benson, S. Nickell, and S. Smith. Getting to know you: open source development meets usability. In *CHI'02 extended abstracts on Human factors in computing systems*, pages 932–933. ACM, 2002.

- [24] L. Zhao and F. Deek. Improving open source software usability. In *Proc. of the 11th Americas Conference on Information Systems*, pages 923–928, 2005.
- [25] Michael Terry, Matthew Kay, and Ben Lafreniere. Perceptions and practices of usability in the free/open source software (foss) community. In *CHI '10: Proceedings of the 28th international conference on Human factors in computing systems*, pages 999–1008. ACM, 2010.
- [26] D.M. Nichols, K. Thomson, and S.A. Yeates. Usability and open-source software development. In *Proceedings of the Symposium on Computer Human Interaction*, pages 49–54. Citeseer, 2001.
- [27] D.M. Nichols and M.B. Twidale. Usability processes in open source projects. *Software Process Improvement and Practice*, 11(2):149–162, 2006.
- [28] C. Benson, M. Muller-Prove, and J. Mzourek. Professional usability in open source projects: GNOME, OpenOffice. org, NetBeans. In *CHI'04 extended abstracts on Human factors in computing systems*, pages 1083–1084. ACM, 2004.
- [29] OpenUsability. Where open source meets usability. <http://www.openusability.org/>, 2010. Sist besøkt 03. Mai, 2010.
- [30] Alexander Faaborg and Daniel Schwartz. Using a distributed heuristic evaluation to improve the usability of open source software. In *CHI '10: Proceedings of the 28th international conference on Human factors in computing systems*, 2010.
- [31] Wikipedia. Commercial off-the-shelf. [http://en.wikipedia.org/wiki/Commercial\\_off-the-shelf](http://en.wikipedia.org/wiki/Commercial_off-the-shelf). Sist besøkt 30. Mai 2010.
- [32] P. Di Giacomo. COTS and open source software components: are they really different on the battlefield? *COTS-Based Software Systems*, pages 301–310.
- [33] B. Boehm and C. Abts. COTS integration: Plug and Pray? *Computer*, 32(1):135–138, 1999.
- [34] J. Li, F.O. Bjørnson, R. Conradi, and V.B. Kampenes. An empirical study of variations in COTS-based software development processes in the Norwegian IT industry. *Empirical Software Engineering*, 11(3):433–461, 2006.

- 
- [35] D. Garlan, R. Allen, and J. Ockerbloom. Architectural mismatch: Why reuse is so hard. *IEEE software*, 12(6):17–26, 1995.
- [36] J. Hoffman and C.A. Cook. Designing for Usability with Cots: How Useful is a Style Guide. In *Human Factors and Ergonomics Society Annual Meeting Proceedings*, volume 42, pages 1295–1299. Human Factors and Ergonomics Society, 1998.
- [37] B.J. Oates. *Researching information systems and computing*. Sage Publications Ltd, 2006.
- [38] D.I.K. Sjoeberg, T. Dybaa, and M. Joergensen. The future of empirical methods in software engineering research. *Future of Software Engineering, 2007. FOSE'07*, pages 358–378, 2007.
- [39] D.I.K. Sjoeberg, B. Anda, E. Arisholm, T. Dybaa, M. Joergensen, A. Karahasanovic, E.F. Koren, and M. Vokac. Conducting realistic experiments in software engineering. In *Proceedings of the 2002 International Symposium on Empirical Software Engineering (ISESE'02)*, pages 17–26. Citeseer, 2002.
- [40] Usability.gov. Heuristic evaluations. [http://www.usability.gov/methods/test\\_refine/heuristic.html](http://www.usability.gov/methods/test_refine/heuristic.html). Sist besøkt 26. April, 2010.
- [41] J.S. Dumas and J. Redish. *A practical guide to usability testing*. Intellect Ltd, 1999.
- [42] Usability.gov. Usability testing. [http://www.usability.gov/methods/test\\_refine/learnusa/index.html](http://www.usability.gov/methods/test_refine/learnusa/index.html). Sist besøkt 26. April, 2010.
- [43] Dag Svanæs. Brukbarhetstesting. <http://www.idi.ntnu.no/emner/it3402/F6Brukbarhetstest.pdf>, 2001. Forelesning.
- [44] J. Nielsen and T.K. Landauer. A mathematical model of the finding of usability problems. In *Proceedings of the INTERACT'93 and CHI'93 conference on Human factors in computing systems*, pages 206–213. ACM, 1993.
- [45] J. Nielsen. Why you only need to test with 5 users. *Jakob Nielsen's Alertbox*, 19, 2000.
- [46] J. Spool and W. Schroeder. Testing web sites: five users is nowhere near enough. In *CHI'01 extended abstracts on Human factors in computing systems*, page 286. ACM, 2001.



## BIBLIOGRAFI

---

- [47] A. Woolrych and G. Cockton. Why and when five test users aren't enough. In *Proceedings of IHM-HCI 2001 conference*, volume 2, pages 105–108. Citeseer, 2001.
- [48] International Organization for Standardization. Iso/iec 25062:2006. Technical report, ISO, 2006.
- [49] J. Nielsen and K. Pernice. *Eyetracking web usability*. New Riders Pub, 2009.
- [50] J. Brooke. SUS-A quick and dirty usability scale. *Usability evaluation in industry*, pages 189–194, 1996.
- [51] J.S. Dumas and J.E. Fox. Usability Testing: Current Practice and Future Directions. *Human-Computer Interaction: Development Process*, page 231, 2009.
- [52] Wikipedia. Single sign-on. [http://en.wikipedia.org/wiki/Single\\_sign-on](http://en.wikipedia.org/wiki/Single_sign-on). Sist besøkt 03. Mai, 2010.
- [53] Michael Hutagalung. Arthemias free: Wordpress magazine theme released. <http://michaelhutagalung.com/2008/05/arthemia-magazine-blog-wordpress-theme-released/>. Sist besøkt 10. Mai, 2010.
- [54] k5n. Webcalendar. <http://www.k5n.us/webcalendar.php>, 2010. Sist besøkt 03. Mai, 2010.
- [55] Wikipedia. Soap. <http://no.wikipedia.org/wiki/SOAP>. Sist besøkt 1. Juni 2010.
- [56] R. Molich. *Usable web design*. Nyt Teknisk Forlag, 2007.
- [57] H.B.L. Duh, G.C.B. Tan, and V.H. Chen. Usability evaluation for mobile device: a comparison of laboratory and field tests. In *Proceedings of the 8th conference on Human-computer interaction with mobile devices and services*, page 186. ACM, 2006.
- [58] Wordpress Blog. Usability testing report: 2.5 and crazyhorse. <http://wordpress.org/development/2008/10/usability-testing-report-25-and-crazyhorse/>. Sist besøkt 30. Mai 2010.
- [59] Jay Fields. Kill your darlings. <http://blog.jayfields.com/2009/03/kill-your-darlings.html>, 2009.

- [60] M. Hughes. Rigor in usability testing. *TECHNICAL COMMUNICATION-WASHINGTON-*, 46:488–494, 1999.
- [61] J. Nielsen. Usability engineering. 1993. *AP Professional, Boston*.
- [62] C. Argyris, R. Putnam, and D.M.L. Smith. *Action science:[concepts, methods, and skills for research and intervention]*. Jossey-Bass San Francisco, 1985.
- [63] J. Kjeldskov, M.B. Skov, B.S. Als, and R.T. Høegh. Is it worth the hassle? Exploring the added value of evaluating the usability of context-aware mobile systems in the field. *Mobile Human-Computer Interaction–MobileHCI 2004*, pages 61–73.
- [64] Wikipedia. Type i and type ii errors. [http://en.wikipedia.org/wiki/Type\\_I\\_and\\_type\\_II\\_errors](http://en.wikipedia.org/wiki/Type_I_and_type_II_errors). Sist besøkt 30. Mai 2010.
- [65] R. Jeffries and H. Desurvire. Usability testing vs. heuristic evaluation: was there a contest? *ACM SIGCHI Bulletin*, 24(4):39–41, 1992.

**Tillegg A**

**SUS-spørreskjema**

## Noen spørsmål om systemet du har brukt.

Vennligst sett kryss i kun en rute pr. spørsmål.

	Sterkt uendig				Sterkt enig
1. Jeg kunne tenke meg å bruke dette systemet ofte.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	1	2	3	4	5
2. Jeg synes systemet var unødvendig komplisert.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	1	2	3	4	5
3. Jeg synes systemet var lett å bruke.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	1	2	3	4	5
4. Jeg tror jeg vil måtte trenge hjelp fra en person med teknisk kunnskap for å kunne bruke dette systemet.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	1	2	3	4	5
5. Jeg syntes at de forskjellige delene av systemet hang godt sammen.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	1	2	3	4	5
6. Jeg syntes det var for mye inkonsistens i systemet. (Det virket "ulogisk")	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	1	2	3	4	5
7. Jeg vil anta at folk flest kan lære seg dette systemet veldig raskt.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	1	2	3	4	5
8. Jeg synes systemet var veldig vanskelig å bruke	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	1	2	3	4	5
9. Jeg følte meg sikker da jeg brukte systemet.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	1	2	3	4	5
10. Jeg trenger å lære meg mye før jeg kan komme i gang med å bruke dette systemet på egen hånd.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	1	2	3	4	5

SUS  
Norsk versjon ved Dag Svanæs  
NTNU 2006

Figur A.1: SUS-spørreskjema

# Tillegg B

## Oppgaver: Brukertest I

- Oppgave 1.  
Logg inn. Du har fått utdelt brukeravn og passord.  
En engangskode vil bli sendt til mobiltelefonen du har fått utdelt.
- Oppgave 2.
  - a) Finn ut om hvor mye man må gå ned i vekt før operasjon.
  - b) Finn ut om du kan få skattefradrag for utgifter grunnet overvekt.
  - c) Hvor mange måltider bør du spise de første ukene etter operasjon?
- Oppgave 3.
  - a) Du lurer på om du kan drikke alkohol etter operasjon. Hva har andre pasienter skrevet om dette?
  - b) Legg ut spørsmålet «Hvor mye spiste dere i tiden etter operasjonen?»
  - c) Få fram en liste over alle deltakere som tilhører «Kirurgi»-brukergruppen.
- Oppgave 4.  
Din venn Dag har sendt deg en melding. Les denne meldingen.
- Oppgave 5.  
Skriv en melding til Dr. Oysnes og spør hvor mange ganger det er normalt å spise om dagen etter operasjon.
- Oppgave 6.

- a) Legg inn påminnelser i kalenderen om å huske å ta vitaminer, søndag 14.03 kl 12.00.
  - b) Finn ut når du har legetime neste gang.
  - c) Du ønsker å få en oversikt over hele året i kalenderen.
- Oppgave 7.
    - a) Du ønsker å ringe overvektsklinikken. Hva er telefonnummeret dit, og hvilke telefontider har de?
    - b) Du ønsker å finne ut mer om Røros Rehabiliteringssenter.

# Tillegg C

## Oppgaver: Brukertest II

- Oppgave 1.  
Logg inn med ditt utdelte brukernavn og passord.
- Oppgave 2.
  - a) Finn ut om hvor mye man må gå ned i vekt før operasjon.
  - b) Finn ut om du kan få skattefradrag for utgifter grunnet overvekt.
  - c) Du lurer på om hva spirometri er. Finn ut av dette.
- Oppgave 3.
  - a) Du lurer på om du kan drikke alkohol etter operasjon. Hva har andre pasienter skrevet om dette?
  - b) Du har lyst å begynne på svømmekurs etter overvektsbehandling, og lurer på om noen andre har gått på dette. Skriv et innlegg hvor du spør om noen vet om et aktuelt svømmekurs i Trondheimsområdet.
  - c) Du ønsker å se hvem som er del av nettstedet, få fram en oversikt over alle deltakerne.
- Oppgave 4.  
Du har mottatt en privat melding, les denne.
- Oppgave 5.  
Du har glemt datoen for neste kontroll. Send en melding til Dr. Oysnes og spør om hvilken dato/tid du skal til kontroll neste gang.
- Oppgave 6.

- a) Finn neste gang du har legetime.
  - b) Legg inn en påminnelse om at du skal ta B12 sprøyte fredag i neste uke.
  - c) Du ønsker å få en oversikt over hele året i kalenderen.
- Oppgave 7.
    - a) Du ønsker å finne mer ut om kostholdet etter behandlingen du har gjennomgått.
    - b) Du ønsker å ringe til obesitasklinikken. Finn telefonnummeret.



Tillegg D

Bakgrunnsskjema

## SPØRRESKJEMA. MOBESITY

Hvor gammel er du?

- 18- 25       26- 30       31- 35       36- 40       41- 45  
 45- 50       51- 55       56- 60       61- 65       66- 75

Kjønn

- Mann               Kvinne

Hva er din høyeste oppnådde utdanning?

- Grunnskole/ Folkeskole               Videregående skole  
 Universitet/ høyskole opptil 3 år       Universitet/ høyskole, mer enn 3 år

Hva beskriver best din livssituasjon?

- Er student  
 Er i arbeid/ jobb  
 Er arbeidsledig  
 Er uføretrygdet  
 Er pensjonert  
 Er langtidssykmeldt  
 Annet \_\_\_\_\_

Din sivilstatus:

- Ugift/ Enslig  
 Enke/ Enkemann  
 Samboer/ Gift

## **ERFARING MED DATA**

Har du datamaskin hjemme?

- Ja                       Nei

Kan du bruke et skriveprogram på en datamaskin (f.eks word)?

- Ja                       Nei

Hvor ofte bruker du en datamaskin?

- Daglig     Ukentlig     Månedlig     Sjeldnere     Aldri

Hvor mange fingre bruker du når du skriver på en datamaskin?

- 1               2               4               Flere               Alle

Hvilket av følgende utsagn beskriver best din erfaring med data:

- Har ingen erfaring med data  
 Har litt erfaring med data  
 Har mye erfaring med data

## **ERFARING MED INTERNETT**

### Bruker du internett?

- Ja                       Nei

### Hvor ofte bruker du internett?

- Daglig     Ukentlig    Månedlig    Sjeldnere    Aldri

### Bruker du internett via datamaskin på noen av følgende steder:

(sett flere kryss hvis aktuelt)

- Hjemme  
 På jobb  
 På universitetet/skolen  
 Hos venner/ familie  
 På biblioteket, kafé, restaurant, hotell el.l  
 Annet sted \_\_\_\_\_  
 Har ikke tilgang til internett

### Hvilke av følgende internettbaserte tjenester bruker du?

- E- post  
 Bestille billetter (f.eks flybilletter, kinobilletter)  
 Handle på internett (f.eks bøker, musikk, klær, sko)  
 Lese nettavis  
 Være del av et nettsamfunn (som f.eks facebook, nettby, biip)  
 Chattekanaler/ Chatterom (snakke skriftlig over internett, for eksempel via facebook, twitter, msn etc)  
 Nettbank (bruke banktjenester over internett)  
 'Surfe' på internett  
 Bruker ingen av disse tjenestene

Hvor mange timer ukentlig bruker du internett?

---

Hvor ofte bruker du nettbank?

- Daglig     Ukentlig     Månedlig     Sjeldnere     Aldri

Hvilket av følgende utsagn beskriver best din erfaring med internett:

- Har ingen erfaring med Internet  
 Har litt erfaring med Internet  
 Har mye erfaring med Internet

### ***BRUK AV MOBILTELEFON***

Har du mobiltelefon?

- Ja                       Nei

Hva bruker du din mobiltelefon til? (sett flere kryss hvis aktuelt)

- |                                       |   |
|---------------------------------------|---|
| <input type="checkbox"/> Ringe        | <input type="checkbox"/> Sende sms (meldinger)    |
| <input type="checkbox"/> Kalender     | <input type="checkbox"/> Lese sms                 |
| <input type="checkbox"/> Klokkealarm  | <input type="checkbox"/> Sende mms (f.eks bilder) |
| <input type="checkbox"/> Internett    | <input type="checkbox"/> Ta i mot mms             |
| <input type="checkbox"/> Spille spill | <input type="checkbox"/> Annet _____              |

Hvor ofte bruker du din mobiltelefon?

- Daglig     Ukentlig     Månedlig     Sjeldnere     Aldri

Om du har noe du ønsker å tilføye i forhold til spørsmålene ovenfor, vennligst skriv kommentar: \_\_\_\_\_

---

---



**Tillegg E**

**Bakgrunnsskjema: Brukertest I**

Testpersoner	T1	T2	T3
<b>GENERELT</b>			
Alder	26-30	18-25	18-25
Kjønn	K	K	M
Høyeste oppnådde utdanning	Universitet/høyskole > 3 år	Universitet/høyskole > 3 år	Videregående
Livssituasjon	I jobb	Student	Student
<b>ERFARING MED DATA</b>			
Har du datamaskin hjemme?	Ja	Ja	Ja
Kan du bruke skriveprogram på data?	Ja	Ja	Ja
Hvor ofte bruker du datamaskin?	Daglig	Daglig	Daglig
Hvor mange fingre bruker du når du skriver på en datamaskin?	Alle	Flere	Alle
Ingen/Litt/Mye erfaring med data?	Litt	Litt	Mye
<b>ERFARING MED INTERNETT</b>			
Bruker du Internett?	Ja	Ja	Ja
Hvor ofte bruker du Internett?	Daglig	Daglig	Daglig
På hvilke steder bruker du Internett?	Hjemme, jobb	Hjemme, skolen	Hjemme, skolen, jobb, venner og familie
Erfaring med Internett-tjenester (Skår 0-8)	6	8	8
Hvor mange timer ukentlig bruker du på Internett?	-	5	75
Ingen/Litt/Mye erfaring med Internett?	Litt	Litt	Mye



T4	T5	T6	T7	T8
18-25	51-55	18-25	18-25	18-25
M	K	M	M	M
Videregående	Universitet/høyskole > 3 år	Videregående	Videregående	Videregående
Student	Student, i jobb	Student	Student	Student
Ja	Ja	Ja	Ja	Ja
Ja	Ja	Ja	Ja	Ja
Daglig	Daglig	Daglig	Daglig	Daglig
Flere	Flere	Alle	Alle	Flere
Mye	Mye	Mye	Mye	Mye
Ja	Ja	Ja	Ja	Ja
Daglig	Daglig	Daglig	Daglig	Daglig
Hjemme, skolen, jobb, venner og familie, kafé e.l.	Hjemme, skolen, jobb, venner og familie, kafé e.l.	Hjemme, skolen, jobb, venner og familie, kafé e.l.	Hjemme, skolen, jobb, venner og familie, kafé e.l.	Hjemme, skolen, venner og familie
8	7	8	8	8
50	Uendelig	60	70	60
Mye	Mye	Mye	Mye	Mye



## Tillegg F

# Bakgrunnsskjema: Brukertest II

Testpersoner	T1	T2	T3	T4	T5	T6
<b>GENERELT</b>						
Alder	51-55	51-55	41-45	31-35	51-55	45-50
Kjønn	Kvinne	Kvinne	Mann	Mann	Kvinne	Kvinne
Høyeste oppnådde utdanning	Univesitet/høyskole	Videregående	Grunnskole	Videregående	Universitet/høyskole > 3 år	Videregående
Livssituasjon	I jobb	50% jobb, 50% uføretrygdet	I jobb	I jobb	I jobb	I jobb
<b>ERFARING MED DATA</b>						
Har du datamaskin hjemme?	Ja	Ja	Ja	Ja	Ja	Ja
Kan du bruke skriveprogram på data?	Ja	Ja	Ja	Ja	Ja	Ja
Hvor ofte bruker du datamaskin?	Daglig	Daglig	Daglig	Daglig	Daglig	Daglig
Hvor mange fingre bruker du når du skriver på en datamaskin?	2	Flere	2	2	Alle	4
Ingen/Litt/Mye erfaring med data?	Litt	Mye	Litt	Mye	Mye	Litt
<b>ERFARING MED INTERNETT</b>						
Bruker du Internett?	Ja	Ja	Ja	Ja	Ja	Ja
Hvor ofte bruker du Internett?	Daglig	Daglig	Daglig	Daglig	Daglig	Daglig
På hvilke steder bruker du Internett?	Hjemme, jobb	Hjemme, jobb	Hjemme	Hjemme, jobb, venner/familie,	Hjemme, jobb	Hjemme, jobb
Erfaring med Internett-tjenester (Skår 0-8)	7	4	5	8	8	8
Hvor mange timer ukentlig bruker du på Internett?	14	7	30	25	20	5
Ingen/Litt/Mye erfaring med Internett?	Litt	Mye	Mye	Mye	Mye	Litt

# Tillegg G

## Kildekode-veiledning

Kildekoden til nettsiden som er utviklet i dette prosjektet er vedlagt i `website.zip`. Denne ZIP-filen er lagt ved den elektroniske innleveringen av denne masteroppgaven.

- Mappen `/basic` inneholder vår egenproduserte kode som inkluderer blant annet autentisering og autorisering gjennom en universal innloggingsløsning samt grensesnitt for administrering av brukere.
- Mappen `/webcal` inneholder kildekoden til kalenderkomponenten. Noe av koden har vi selv modifisert til å passe systemets behov.
- Mappen `/wordpress` inneholder kildekoden til Wordpress, samt alle innstikk og temafiler vi selv har installert. Noe av koden er også modifisert til å passe systemets behov.