



---

# INFO210 - Prosjekt i systemutvikling

Fisheries Daily System, Høsten 2009

---

Av Regine O. Sagstad, Åshild A. Torpe, Marianne Røsvik og Ingrid Nese

# Innhold

1.1 Takk.....	3
2.1 Innledning .....	3
2.2 Sammendrag.....	4
2.3 Introduksjon .....	5
3.1 Bakgrunnsinformasjon .....	5
4.1 Metoder og gjennomføring .....	6
4.2 Fossefallsmodellen.....	6
5.1 Analyse av resultat .....	7
5.2 Scenarier.....	7
5.3 Brukstilfellediagram.....	8
5.4 Kravspesifikasjon.....	9
5.5 Målgruppe .....	9
5.6 Testplan .....	9
5.7 Databasemodell.....	10
5.8 Prototype .....	10
5.9 Analyse av FDS.....	11
5.9.1 Analyse av FDS for fartøy .....	12
5.9.2 Analyse av FDS for Fiskeridirektoratet. ....	13
5.9.3 Manglende funksjoner .....	14
6.1 Evaluering .....	15
7.1 Konklusjon.....	17
8.1 Referanser og bibliografi .....	19
8.2 Inspirasjonsliste.....	19
8.3 Bildereferanser .....	19
9.1 Ordliste.....	20
10.1 Vedlegg.....	22

## 1.1 Takk

Følgende oppgave er utviklet i emnet "Prosjekt i Systemutvikling" ved Universitetet i Bergen. Fiskeridirektoratet har bidratt til stor nytte for gruppen, ved å stille til intervju, samt vise frem systemene som blir brukt i fiskeriindustrien i dag. Vi skylder dermed Guro Gjelsvik ved Fiskeridirektoratet en stor takk.

## 2.1 Innledning

Oppgaven vår går ut på å utvikle et system som skal forenkle og tilfredsstillende fiskeriindustriens elektroniske gjøremål. Målet vårt er å lage en prototype som skal vise systemets utseende og funksjonalitet. På denne måten kan vi presentere prosjektet med de løsninger vi synes passer best til oppgaven. Vi avgrensner prototypen til å være delvis funksjonell. Det vil si at vi ønsker å lage skjermbilder for å vise systemets hovedfunksjoner. Dermed vil resultatet ikke bli en ferdig programvare. For å lage og presentere prototypen vil Adobe Photoshop og Adobe Flash bli brukt. Systemet baseres på at teknologien som allerede finnes på fartøy blir tatt i bruk, deriblant ekkolodd, satellitt, gps og annen teknologi (se vedlegg1, krav #29). Systemet vil ha en underliggende database som inneholder data over alle fiskemottak, fartøy, fiskere, rapporter, statistikker, og lignende.

Det første vi gjorde etter at vi hadde valgt hvilken oppgave vi ville jobbe med, var å lage en Project Initiation Document (se vedlegg 11). Ved å lage dette dokumentet ble vi enige om hvilken retning prosjektet skulle ta, i tillegg fikk vi avgrenset prosjektet og blitt enige om mål og spesifikasjoner. Etter at Project Initiation Document var laget hadde vi et klart mål for prosjektet, og prosjektplanen (se vedlegg 6) ble dermed utformet. Her definerte vi de ulike oppgavene vi hadde foran oss. Hvor lang tid vi skulle bruke på hver oppgave, og hvem som skulle gjøre hva. I tillegg lagde vi et tidsskjema. Ettersom vi ikke har kjennskap til fiskeriindustrien måtte vi gjøre forundersøkelser for å tilegne oss mer kunnskap om industrien. Dette var viktig slik at vi kunne lage et system som var realistisk, og som hadde funksjoner som målgruppen hadde behov for.

Siden vi kun lager en prototype må vi ta en del antagelser og bestemme oss for hvordan vi ville ha gjort det dersom vi skulle ha laget hele systemet fullt ut. Det må også gjøres avgrensninger, og her gjaldt dette blant annet at prototypen begrenser seg til å gjelde innenfor norske farvann og kan kun brukes av norske fartøy. Et fullstendig system vil også inkludere utenlandske soner og fartøy.

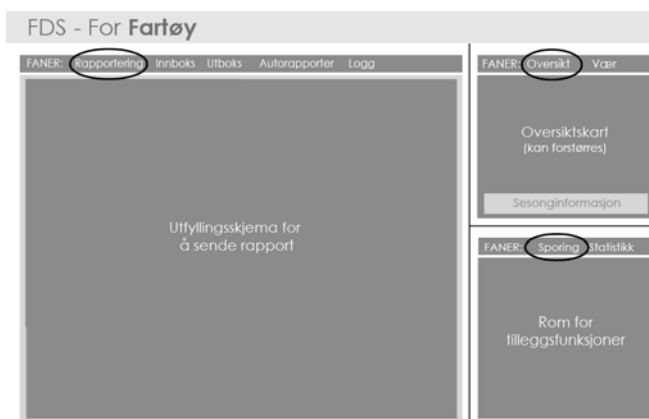
For å oppnå krav #32, 37 og 42 skal systemet være en programvare, ikke en nettside, og kjøre på eksisterende maskiner. Programvaren må altså installeres på alle maskiner som skal bruke systemet. Avgjørelsen er tatt ut ifra at systemet krever mye CPU, ettersom det skal være mulig å aktivere flertallige funksjoner. Datamaskinen programmet kjører på trenger internettforbindelse for å kunne

sende rapporter. I tillegg er et argument at flertallet av fartøy har utstyr om bord i båten som har kapasitet nok til å kjøre et slikt program.

I fellesskap utformet vi en visjon for prosjektet; *”Fisheries Daily System skal være et felles elektronisk overvåknings- og rapporteringssystem for både fartøy, direktorat og fiskemottak.”*

Systemet skal gjøre hverdagen enklere for fiskeriindustrien ved å samle arbeidsoppgaver i et dataprogram, samt forbedre overvåkningsmulighetene. Ved hjelp av Fisheries Daily System (heretter kalt FDS) skal Fiskeridirektoratet enklere kunne oppspore manglede rapporter, regelbrudd eller ulovlig fiske gjennom satellittovervåkning. Dette vil skje ved både jevnlig automatiske rapporteringer fra aktive fartøy, i tillegg til manuell rapportering, som også sendes til godkjenning ved fiskemottaket hvor fartøyet leverer sin fangst.

Ettersom Fiskeridirektoratet, fartøy og mottakene har ulike arbeidsoppgaver, må det lages tre ulike program tilpasset hver sin målgruppe. For å illustrere hvilke hovedfunksjoner de ulike programmene må ha, laget vi en skisse av hvert program med oversikt over inndelingen (Se vedlegg 9). Hovedfunksjonen for et fartøy vil være å kunne sende fangstrapporter (se krav #2), men de vil også ha mulighet for å legge til selvvalgte oppgavevindu etter behov. Mottakets hovedfunksjon vil være å godkjenne innkomne rapporter, samt utfylle prisinformasjon som beskrevet i krav #21, mens Fiskeridirektoratets hovedoppgave blir å overvåke alle innkomne rapporter, samt de manglede (Se krav #14 og #15).



## 2.2 Sammendrag

Vi vil i denne rapporten vise fra A til Å hvordan vi gikk frem for å lage et system for fiskeriet. Rapporten inneholder alle de viktige faktorene i utviklingsfasen. Dette innebærer oppsett av prosjektplan, research, valg av utviklingsmetode, utforming av krav, skisseringer, brukeranalyse, innholdsinnstilling, utforming av selve systemet, evaluering og analyse. Resultatet av prosessen ble en prototype av FDS, som viser systemets utforming og viktigste funksjoner. I tillegg til en flashbasert utgave av programvaren, har vi også utviklet brukstilfellediagram og en databasemodell som viser forholdet mellom de ulike oppgavene og enhetene.

## 2.3 Introduksjon

Oppgaven går ut på å utvikle et system som automatisk sender rapporter og melder i fra om fartøyets posisjon og status, samt sende et varsel ved rapportfravær. Målet med prosjektet vårt er først og fremst å vise en systemutviklingsprosess der arbeidet underveis og hvordan vi kommer frem til resultater vil være i fokus. Vi vil derfor bruke god tid i starten av prosjektet på å utforske eksisterende verktøy, hva som mangler, og hva vi kan gjøre for å forenkle rapporterings- og overvåkningsprosessen. Vi vil også forske på hvordan vi best mulig kan skape et brukergrensesnitt som når hver enkelt bruker; Fiskeridirektoratet, fartøy og fiskemottakene. Etersom vi har begrenset tid på denne oppgaven, vil vi kun lage prototyper for programvare til Fiskeridirektoratet og fartøy, selv om det i utgangspunktet finnes tre brukergrupper av systemet.

## 3.1 Bakgrunnsinformasjon

For å kunne utvikle en prototype måtte vi ha rikelig med bakgrunnsinformasjon om emne. Anskaffelse av dette begynte først og fremst med å gå inn på Fiskeridirektoratets nettside; <http://fiskeridirektoratet.no>. Denne nettsiden gir utfyllende informasjon om systemer som er i bruk på fartøy og på land, og inneholder blant annet kartinformasjon, kvoter, regler og oppdaterte nyheter. På nettsiden fikk vi tilgang til et dokument som omtalte systemet *SatRap*. Dette systemet utfører noen av de samme oppgavene vi ønsker at vårt system skal utføre.

*SatRap* er nemlig et fangst- og aktivitetsrapporteringssystem, som er utviklet av Fiskeridirektoratet. Dette bruker telefaks og e-post for rapportering. Omtrent 100 fartøy bruker i dag *SatRap* til å rapportere fangst- og aktivitetsdata. Dette er dermed en interessant ressurs for oss.

*«Grunnen til at det bare er omtrent 100 fartøy som bruker det er at reglementet i Norge sier at man bare er meldepliktige når man fisker i utenlandske soner.»* (personlig kommunikasjon, G. Gjelsvik, 25.september 2009).

Guro Gjelsvik, som er ansatt på Fiskeridirektoratet, gav oss bakgrunnsinformasjon om systemet *SatRap*. Hun informerte også om hvordan andre systemer fungerer, både med tanke på mottak og fartøy. Blant annet har Fiskeridirektoratet et prøvesystem hvor alt foregår elektronisk gjennom systemet eFangst.

I følge Guro Gjelsvik vil systemene som er under utvikling inkludere at fartøy også rapporterer innenfor norske soner. I vårt system, har vi tatt utgangspunkt i at dette er tilfellet. På denne måten blir det mulig for Fiskeridirektoratet å få en fullstendig oversikt over all aktivitet som skjer i norske soner. Sannsynligheten for å oppdage ulovlig fiske vil dermed øke.

I tillegg til møtet, benyttet vi også flere internettressurser. Deriblant for å finne statistikker,

informasjon om fiskesoner og forsøk på å analysere målgruppene. I forbindelse med dette forarbeidet var vi blant annet inne på <http://fiskaren.no>, som er en viktig nyhetskilde for fiskeriet. Flere nettkilder blir nevnt i inspirasjonslisten (se avsnitt 8.2).

## 4.1 Metoder og gjennomføring

En modell er et forenklet bilde av virkeligheten, og hensikten med å bruke den er å redusere kompleksiteten i den virkelige verden. En systemutviklingsmetode består av teknikker for å utføre de ulike trinnene i systemutviklingsprosessen, og beskrivelsen av hvordan disse teknikkene skal anvendes i forskjellige trinn i utviklingsprosessen. (Braadland, 2002)

Stadiene alle utviklingsprosjekter går gjennom er svært like, uavhengig av metoder, teknikker og verktøy som brukes. For at vi skulle få en best mulig prosess i utviklingen av systemet valgte vi å bruke de ulike trinnene som er involvert i en systemutviklingsprosess. Systemutvikling er prosessen med å utvikle datamaskinbaserte informasjonssystemer. Den mest kjente modellen for systemutvikling er livssyklusmodellen, som gir et rammeverk for håndtering og strukturering av et prosjekt. De tre modellene som brukes oftest i studentprosjekter er fossefallsmodellen, spiralmodellen, og den inkrementelle modellen (Weaver, 2004).

## 4.2 Fossefallsmodellen

Ettersom fossefallsmodellen gir et godt utgangspunkt for å strukturere et studentprosjekt har vi valgt å bruke denne modellen i vårt prosjekt. Det ligger i modellen at hver fase skal være avsluttet før den neste kan begynne. Dokumentasjonen fra en fase danner grunnlaget for den neste, og dette gjør at arbeidet innen hver fase må "fryses" for at det skal være mulig å fortsette. For hver fase produseres det dokumentasjon og delprodukter som brukes i neste fase. Den ferdige konstruksjonen er lett å bygge en enkel plan rundt, og slutten av hver fase gir studentene og veileder en pekepinn på at prosjektet går fremover (Weaver, 2004).

Når man er ferdig med en fase, kan man ikke gå tilbake. Ettersom dette er et studieprosjekt og det ikke investeres penger i det, vil vi praktisk talt ha en mulighet for å gå tilbake. For å skape et virkelighetspreg over prosjektet, ønsket vi i utgangspunktet å forholde oss til en stegvis prosess i henhold til prinsippene modellen bygger på. Men, da vi fikk en avtale med Fiskeridirektoratet, måtte vi gå tilbake et steg i modellen for å tilpasse systemet den nye informasjonen vi hadde fått. Prosjektplanen stemte dermed heller ikke overens og vi måtte utvide tiden vi hadde satt av til research.

I fossefallsmodellen skal alt ved systemet dokumenteres på forhånd, og det skal settes ned god tid

til kravspesifisering. Den kan ses på som urealistisk, ettersom brukeren forandrer krav og meninger underveis i prosjektet. En av fordelene med denne modellen er at den er rigid og forutsigbar. Ved å bruke denne modellen vil det forenkle fordelingen av arbeidsoppgaver ettersom den foregår steg for steg. Dette gjør det også enklere å beregne tid for inndelinger av oppgaver, noe som passer oss bra i og med at et studentprosjekt har begrenset tidsomfang. Modellen vil hjelpe oss å få en god struktur over prosjektet, slik at vi hele tiden har kontroll på hvor langt vi har kommet og hvor mye som gjenstår. Denne arbeidsmetoden sikrer også arbeidsprosessen og kvaliteten for vår egen del.

## **5.1 Analyse av resultat**

Formålet med analysen er å få en helhetlig vurdering av systemet.

## **5.2 Scenarier**

Vi har valgt å ta i bruk scenarier for å hente ut krav til systemet. Denne teknikken blir også brukt for å vise hvordan systemet skal hjelpe de ulike brukerne. Scenarier er en god måte å få frem forskjellige aspekter i systemet.

### **1. Scenario: Fiskeridirektoratet**

Atle jobber på Fiskeridirektoratet og har som oppgave å overvåke fartøy langs Norskekysten. Han er fortvilet over dagens system som ikke tilfredstiller de kravene Asle gjerne kunne tenke seg at systemet gjorde. Et eksempel på dette er at når et fartøy ikke melder seg aktiv i en sone vil det være vanskelig for Asle å finne igjen fartøyet uten omfattende satellittøk. I tillegg er det vanskelig for han å holde oversikt over alle som har overskrevet fangstknoten. Dette er bare to av flere problemer som Asle møter på en vanlig arbeidsdag. Problemene nevnt over har sammenheng med hva fartøyet melder fra om. Fiskeridirektoratet vil også trenge å gi beskjeder, nyheter og varsler om regelbrudd ut til ulike fartøy. Noen ganger er det også vanskelig for Asle å finne igjen rapporter som blir sendt til fiskeridirektoratet.

### **2. Scenario: Fartøy og fiskere**

Mannskapet på Havglans sliter med å få gjennomført rapportskrivningen som inneholder viktig informasjon om fangst og feil. Dette viser seg å være tidkrevende. Fartøyet er også meldepliktige. De har ansvar for å melde fra om hvor de befinner seg og om de er passiv, eller aktiv i en fiskesone. Mannskapet vil også planlegge fiskesesongen, og derfor er de avhengig av å vite hvilke fisker det er sesong for. Dette har vist seg å være tungvint og problematisk å få gjennomført. Annen informasjon som vil være nyttig og interessant for

mannskapet på Havglans, er statistikk, værmeldinger, havstrømninger, sporing og kommunikasjon. Den eksisterende situasjonen i dag er at man må bruke forskjellige system til de ulike gjøremålene. Dette fører til at de bruker mer tid på det enn ønskelig.

### **3. Scenario: Mottak**

Staben på fiskemottaket i Bergen har store problemer med å holde oversikt over rapporteringen fra fartøy. De opplever at det er unødvendig tungvint å fylle ut fangstskjemaer, samt sjekke at fangstopplysningene fra fartøy stemmer. Problemene som da oppstår er at mottaket i Bergen feilrapporterer videre til Fiskeridirektoratet. Slike ringvirkninger vil mottaket unngå. Mottaket vil også få ut den informasjonen som kan være nyttig for fartøy, og det er ofte vanskelig med tanke på hvor de skal sende informasjonen slik at flest mulig får relevant informasjon.

### **4. Scenario: Løsning- Fisheries Daily System**

Atle, mannskapet på Havglans og staben på fiskemottaket i Bergen har en ting til felles, de savner et system som sammenfatter oppgavene som skal utføres. Det nye systemet Fisheries Daily System er blitt lansert og hevder å ha løsningen på alle deres problemer. Informasjonen blir samlet på et sted og vil være mer effektiv, enn å ha forskjellige komponenter til å utføre oppgavene. Mottaket kan nå på en enklere måte fylle ut og videresende godkjente rapporter til fiskeridirektoratet. Mannskapene kan lettere registrere fangst, status, avvik og posisjon. Fiskeridirektoratet får samlet informasjonen de behøver på en plass. Ettersom flere verktøy har blitt brukt tidligere har det tatt lengre tid enn nødvendig å få oversikt over hva som skjer til havs. Med det nye systemet holder det med noen få skjjermer for å få en total oversikt. Automatiske og manuelle rapporter vil bli håndtert på en mer oversiktlig måte og lagret i en omfattende, underliggende database. For å oppnå krav #25 vil databasen jevnlig ta backup på maskiner lokalisert på Fiskeridirektoratet. Fisheries Daily System erstatter telefaks og e-post med et samlet system.

## **5.3 Brukstilfellediagram**

For å gi et overblikk over systemet har vi valgt å ta i bruk brukstilfeller. Det er viktig å kunne forstå hvordan systemet skal fungere og interagerer med verden (personlig kommunikasjon, A. Babic, 24.september 09). Et brukstilfelle er assosiert med en aktør, og det er aktørens mål ved å bruke systemet brukstilfellet vil fange opp (Sharp, Rogers, Preece, 2007). Brukstilfelle kan – som vi har gjort – bli fremstilt grafisk. Vårt brukstilfellediagram viser hver av brukergruppene, samt en illustrasjon som viser hvordan systemene skal integrere med hverandre, og hvilken informasjon som

skal sendes mellom dem. Se vedlegg 3.

## 5.4 Kravspesifikasjon

Et krav er en påstand om et tenkt produkt som spesifiserer hva det skal gjøre og hvordan det skal fungere. For at systemet skal fungere optimalt må vi stille krav til både brukeren og systemet. Det finnes to typer krav: funksjonelle krav og ikke- funksjonelle krav. Funksjonelle krav sier hva systemet skal gjøre, mens ikke- funksjonelle krav beskriver ytelse og begrensninger i systemet. (Weaver, 2004)

Vi har hentet ut krav ved hjelp av scenarier og brukstilfellediagram. For å spesifisere funksjonaliteten til systemet laget vi en liste over kravspesifikasjoner, både funksjonelle og ikke-funksjonelle krav. Ettersom det finnes tre ulike brukergrupper av systemet, har vi laget egne tabeller for hver enkelt gruppe. Se vedlegg 1.

## 5.5 Målgruppe

Ethvert system har en målgruppe som tidligere nevnt er det både fiskere, fiskeridirektorat og fiskemottak som skal inkluderes i vårt system. For å kunne lage et produkt som på en best mulig måte skal kunne forbedre hverdagen til disse arbeidsgruppene er det viktig å drive research. På denne måten vil vi som utviklere vite mer om hvordan systemet blir brukt nå, og hvordan vi på en best mulig måte kan lage et system som er tilrettelagt den målgruppen vi ønsker å ha. Det er viktig å tenke på hvordan, og av hvem systemet skal bli brukt av, slik at sluttproduktet ikke ender opp som en overflødig programvare. Se vedlegg 2.

## 5.6 Testplan

Det finnes mange måter å teste et system på. Vi vil teste vårt system i en systemtest. Det går ut på at de på gruppen som har arbeidet med analysering av systemet også vil teste systemet. (Weaver 2004) Grunnen til at vi valgte å gjennomføre en systemtest er fordi vi har et system som ikke fungerer totalt. Det er begrenset med funksjonalitet og de som har analysert systemet vet dette. Derfor er det mulig å ta en systemtest fordi vi vet hva vi vil at systemet skal utføre. Med en systemtest vil vi se om vi får våre forventede resultat. Vi har ikke laget en testplan på dette ettersom systemet vårt ikke er ferdig, men bare en prototype. Argumentene vi har for å ikke lage en plan er at vi har lagt inn de funksjonene vi trenger for å vise systemet. Etter en gjennomgang av alle funksjonene, ser vi at de fungerer slik de skal, og det ville være unødvendig å lage en egen testplan for å sjekke dem. For å sjekke brukervennligheten til prototypen vår, vil vi foreta en HCI-analyse. Det går ut på å gå gjennom de ulike sidene og elementene på sidene som har med brukergrensesnitt å gjøre. Resultatet

fra denne analysen er å finne i avsnitt 5.9, 5.9.1 og 5.9.2

## 5.7 Databasemodell

For å vise hvilke data som skal lagres og behandles laget vi en databasemodell. En databasemodell er strukturen eller formatet til en database, beskrevet med et formelt språk støttet av DMS. Databasemodellen er teorien eller spesifikasjonen som beskriver hvordan en database er strukturert og brukt (Wikipedia, 2009a). I denne oppgaven valgte vi å lage en semi-strukturert modell, SSM. I denne modellen er det ikke et klart skille mellom dataene og skjemaet, og mengden av struktur som blir brukt kommer an på meningen med modellen. Fordelene med en SSM-modell er at den kan presentere informasjonen om visse datakilder som ikke kan bli begrenset av et skjema, den kan lett bli endret på, og den kan være til stor hjelp for å se strukturert data som semi-strukturert (Wikipedia, 2009b). Se vedlegg 4.

## 5.8 Prototype

Etter at alle behovene er kartlagt må systemet designes, og første steget da er som regel design av en prototype. Designet må reflektere de funksjonelle og ikke-funksjonelle kravene som er satt for systemet (Braadland, 2002). Det å lage en prototype er til god hjelp for å se hvordan et system kan se ut. Prototyper svarer på spørsmål og hjelper designere i å velge mellom forskjellige tenkte alternativer (Sharp, Rogers, Preece, 2007).

En prototype kan enten være av low-fidelity eller high-fidelity. Dersom en prototype skal kategoriseres som en low-fidelity prototype vil det være en prototype som ikke ligner på det ferdige produktet. Det er ikke meningen at man skal ta vare på og integrere prototyper av denne typen i det ferdige produktet (Sharp, Rogers, Preece, 2007). En high-fidelity prototype bruker derimot det materialet man forventer seg i det ferdige produktet, og vil svare til forventningene for hvordan det ferdige produktet forventes å se ut. High-fidelity prototyper er til god hjelp for å selge ideer til personer og for å teste ut tekniske spørsmål (Sharp, Rogers, Preece, 2007).

Vår prototype er en blanding av begge de nevnte typene for prototyper. Vi har benyttet oss av Flash fordi vi ønsker å gi brukeren mulighet til å se for seg hvordan man kan interagere med systemet, og for å vise hvordan vi ser for oss, og ønsker at funksjonaliteten skal være. Vår prototype er av low-fidelity fordi man ikke kan se hvordan systemet skal se ut i virkeligheten med tanke på hvilket materiale den er laget i, og hvor den skal plasseres hos de forskjellige brukerne (i båten, på mottaket og hos fiskeridirektoratet). Men, man kan også se på prototypen som en high-fidelity prototype fordi man faktisk kan trykke seg rundt ved hjelp av musepeker, for å se hva som skjer når man

navigerer seg rundt på sidene, og ved hjelp av dette få et inntrykk av hvordan det ferdige systemet faktisk blir. Se vedlegg 5.

## 5.9 Analyse av FDS

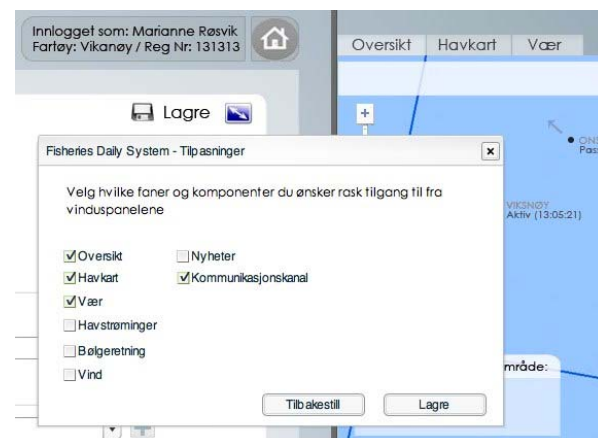
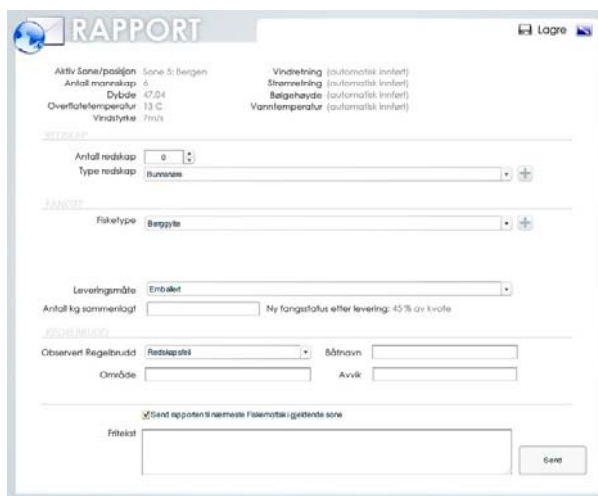
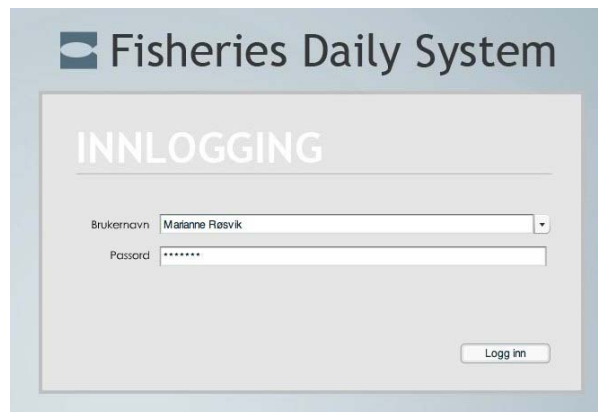
Når man presenteres for et nytt system er det viktig at det ikke kreves for mye av brukerne. Vi har derfor valgt å bruke *usability goals* i utviklingen av vårt system. Ved å gjøre dette vil en som utviklere kunne forsikre seg om at en har laget et system som er effektivt, raskt, trygt, har god funksjonalitet, er enkelt å lære og å bruke, samt enkelt å huske hvordan man bruker. Det er en stor fordel å tenke på disse målene i utviklingen av et nytt system fordi folk ikke liker å bruke lang tid på å forstå hvordan et system fungerer. De vil ha muligheten til å prøve ut produktet og de ulike funksjonene uten for mye anstrengelser (Rogers, Preece og Sharp, 2007).

Vi har også valgt å følge noen grunnleggende designprinsipper under utviklingen av prototypen. Designprinsipper er generaliserte abstrakte begreper som er ment for å orientere designere mot å tenke over forskjellige aspekter av designene deres (Sharp, Rogers, Preece, 2007). *De forskjellige designprinsippene er synlighet, tilbakemeldinger, begrensninger, konsistent og rettleiding.*

Vår prototype synliggjør for brukerne av FDS hvordan de skal gå frem for å utføre de ulike oppgavene systemet tilbyr. Knapper og tekstfelt er hovedfaktorer her, disse er synlige, og de er elementer som på en tydelig måte viser hvilke oppgaver som kan utføres. Disse grensesnittkomponentene er designet for å utføre visse oppgaver, og hver enkelt har en egen rolle i grensesnittet (Heim, 2007). Man må derfor være bevisste på valg av disse komponentene, og hvor man plasserer de. Vi har vært bevisste med bruken av ikoner og knapper gjennom hele designprosessen vår. Brukerne får også en tilbakemelding på oppgavene de utfører, for eksempel når de har sendt en rapport, får de opp et popup-vindu som gir en bekreftelse på at de har sendt rapporten. I tillegg er det fargekoder. Rød farge indikerer at noe mangler og grønn farge indikerer at noe er gjennomført på riktig måte. I systemet er det også begrensninger som hjelper brukerne å handle på riktig måte, for eksempel rullegardinmeny. Det kan blant annet være med på å hindre brukeren å registrere feil fisketype, sone, og lignende. Et annet eksempel på en begrensning, som ligger implementert i systemet, er at man ikke kan være aktiv i flere soner samtidig. I tillegg kan man ikke sende samme rapport to ganger fordi hver rapport har et unikt rapportnummer. Systemet benytter funksjoner og design som er gjenkjennelig fra andre systemer. Eksempler på dette er piler, haker, combobokser og lignende elementer. Ikoner som er brukt i systemet er for advarsler, godkjenninger, svar, fullskjerm og lagre. Grunnen til bruken av gode og presise ikoner er fordi ikoner er lettere å lære og gjenkjenne enn tekst (Sharp, Rogers, Preece, 2007).

### 5.9.1 Analyse av FDS for fartøy

Det første man møter når man åpner systemet for fartøy er et innloggingsvindu. Dette er en viktig faktor for å ha et sikkert system. Prototypen viser bare hvordan innloggingen i det endelige systemet ville vært visuelt, men er ikke reel. Vi tenker oss at alle registrerte fiskere, ansvarlig på fiskemottak, og ansatte på Fiskeridirektoratet får tildelt egne passord og brukernavn. Vi ser for oss at det fastsettes en gruppe administratorer for systemet som har ansvar for rolletildeling og videre teknisk vedlikehold av systemet. Programmet for både direktoratet og fartøy har en verktøylinje øverst som vil gi valg som brukere er vant til fra andre system. Dette er for å sikre en større effektivitet og bedre brukervennlighet. Selve vinduet er delt inn i tre forskjellige deler. Den største delen, rapport, ligger i fokus fordi dette er hovedbruksmålet for denne delen av systemet. I rapportdelen finner man både ferdigutfylte opplysninger, og felt som må implementeres manuelt av fiskerne. Dette mener vi gir grunnlag for en nøyaktig rapport. Felter som posisjon, vindstyrke, vindretning og så videre, blir automatisk fylt inn ved hjelp av sensorer i fartøyet som fanger dette opp. Rapportfeltet tar i bruk



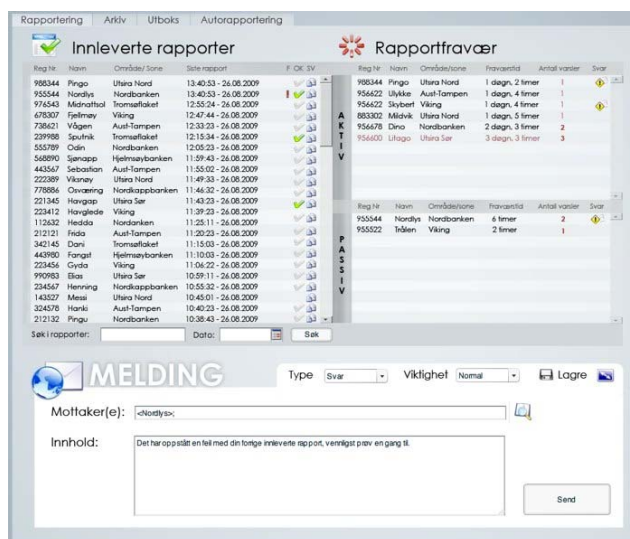
combobokser, rullegardinmenyer, samt tekstfelt som skal fylles inn med opplysninger. Rullegardinmenyer er til god hjelp når man skal velge mellom ulike valg. Det gir ikke rom for feil som for eksempel skrivefeil, eller misforståelser for hva som kan fylles inn. Man kan velge å lagre rapporten, eller å sende den direkte til Fiskeridirektoratet for så å få en tilbakemelding. På rapportensiden finner man også en fanemeny som gjør at fiskerne enkelt kan sjekke innboks, utboks, autorapportering og logg. Grunnen til at vi har valgt å ta med dette er fordi man på denne måten enkelt kan følge med på hva som er gjort før, og ha en god oversikt. I vinduene til høyre kan man

velge mellom hvilke faner man vil vise. De forhåndsvalgte fanene er oversikt, havkart, vær, sporing (av fisk ved hjelp av ekkolodd) og statistikk. Man kan velge å legge til ekstrasfaner som havstrømninger, bølgeretninger, vind, nyheter, kommunikasjonskanal, områdestatistikk, fangststatistikk og sesonginformasjon. I det forhåndsinnlagte oversiktvinduet vil man kunne følge med på hvilke andre fartøy som befinner seg i området, melde seg aktiv eller passiv, samt en oversikt over hvilke fisker det er sesong for i gjeldende sone. Man kan også velge å utvide alle vinduene til fullskjerm (ved å klikke på det blå ikonet med pil i begge retninger) dersom man ikke ønsker å ha programmet delt opp i tre deler. Dette er med på å øke brukervennligheten for systemet. Se vedlegg 5a.

### 5.9.2 Analyse av FDS for Fiskeridirektoratet.

Systemet for Fiskeridirektoratet inneholder et hovedvindu og mindre vinduer i høyre marg. Hovedvinduet har en statisk fanemeny som inneholder rapportoversikt, innboks, utboks, logg og autorapportering. Under fanemenyen følger «Innleverte rapporter». Her har hver innleverte rapport et unikt nummer, navn, område/sone og dato/klokkeslett. Det røde utropstegnet indikerer at rapporten varsler om et regelbrudd eller en feil.

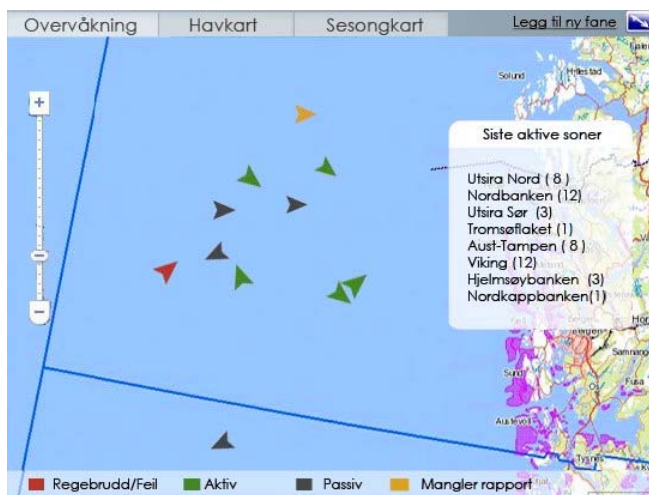
Den grønne haken indikerer at rapporten har gått gjennom fiskemottaket, merket med pris og er godkjent. Se vedlegg 3d. Konvoluttikonet bak haken gir brukeren en enkel snarvei til å svare på akkurat denne innleverte rapporten. Mottakernavnet vil da dukke opp i boksen for ny melding. En søkefunksjon er lagt til under de innleverte rapportene. På den måten kan brukeren for eksempel enkelt søke seg frem til eldre rapporter, eller rapporter fra et spesielt fartøy.



Til høyre i hovedvinduet vises «Rapportfravær» som er delt inn i to. En for aktive fartøy og en for passive fartøy. De som er aktive skal rapportere hvert døgn, mens fartøy som har meldt seg passive skal rapportere innen to timer. Aktive fartøy som ikke rapporterer vil få første advarsel etter ett døgn, andre advarsel når det har gått to døgn, og tredje og siste advarsel etter tre døgn. Passive får første advarsel etter to timer som nevnt over, andre advarsel får de etter seks timer og tredje advarsel etter ti. De som har fått tre advarsler får en melding merket med rød skrift. Nederst til høyre på siden finner vi meldingsboksen. Her er det mulig å sende meldinger med ulik innholdstype

og viktighet. Grunnen til at vi har valgt å legge inn ulike innholdstyper og viktighet er at det finnes forskjellige typer meldinger, og for at brukerne enkelt skal kunne skille de viktige fra de mindre viktige. Lagreikon er også selvsagt lagt til for at man kan lagre meldingen, og ta den frem igjen ved en senere anledning. De sendte meldingene blir lagret i utboks. Innkommende meldinger blir lagret i innboks.

Kartseksjonen til høyre har også fanemeny som er justerbar etter eget ønske. Kartet viser i nåværende vindu de ulike fiskesonene vi har langs norskekysten. En zoom-funksjon er lagt til for å optimalisere oversikten over alle fiskesonene. I tillegg er det en boks med siste aktive soner som oppdaterer antall fartøy i sonene, og eventuelle regelbrudd. Fordelen med å ha integrert et kart i systemet er at man får samlet informasjonen og posisjonen til



ulike fartøy i et og samme system. I tillegg er det godt visualisert med ulike farger på fartøy for å synliggjøre hvilken status fartøyet har. Under kartet finner man en fanemeny som viser regelbrudd og statistikk. Denne fanemenyen er dynamisk og kan tilpasses. Helt nederst er det også lagt til en søkefunksjon som gir mulighet for å søke blant regler og paragrafer. På den måten kan man alltid ha en oversikt over de nyeste reglene og i tillegg finne statistikk på saker som angår emnet.

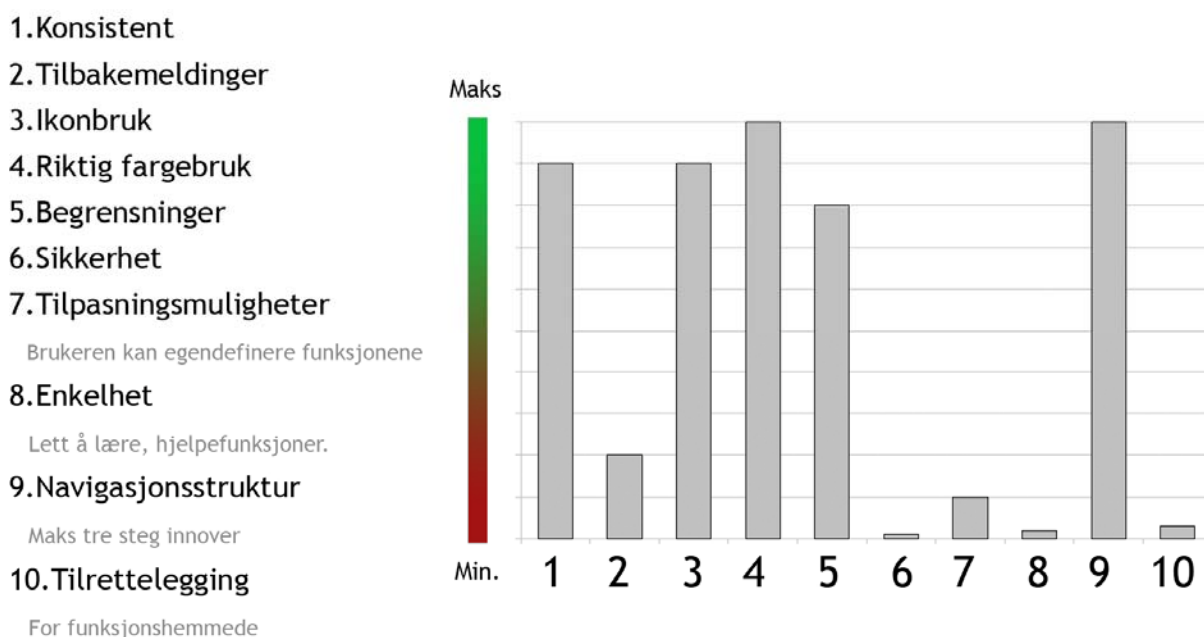
### 5.9.3 Manglende funksjoner

En prototype er ikke fullstendig, og har derfor flere mangler, som ville vært med i en ferdigutviklet programvare. Vi ønsker blant annet at systemet skal benytte seg av tastatursnarveier, som gjør systemet raskere å bruke. Et velutviklet grensesnitt bør imøtekomme både nybegynnere og erfarne brukere (Sharp, Rogers, Preece, 2007). Vi mener at tastatursnarveiene vil kunne øke brukervennligheten og tilfredsheten til mer erfarne brukere. Et system uten tastatursnarveier vil på mange måter kunne sies å være et ineffektivt system. Det å inkludere snarveiene i et system imøtekommer bruksmålprinsippet om at systemet skal være kjapt i bruk (Sharp, Rogers, Preece, 2007). Dette er ikke noe vi kan vise til i prototypen vår, men det er viktig å vise forståelse for disse elementene slik at det kan bli tatt med i en videreutvikling av systemet. Vi ville også hatt med noe som kalles for «tooltip». Enkelt forklart kommer det opp en beskrivelse av verktøyet når man holder musepekeren over en knapp. Grunnen til at vi ville hatt dette med er at det gir brukerne en hurtig forklaring om det skulle være noe som er uklart i systemet. På fanemenyene ønsker vi en

utheving, i form av en lysere farge, på den fanen som er aktiv. På den måten får brukerne en visuell bekreftelse på hvilken fane som faktisk er aktiv om det skulle være vanskelig å se. Endelig programvare skal også ha en inkludert brukermanual i form av innebygde hjelpefunksjoner. Disse skal man kunne få tilgang til i en hjelpemeny, samt at det dukker opp tips første gang man bruker systemet.

For å visualisere de manglende funksjonene i prototypen har vi valgt å lage et stolpediagram.

Fargebarometeret viser en skalering på kritiske og ukritiske mangler. Punktene som ikke når opp til den grønne fargen er kritisk. Punktene er nummerert og hvert tall viser til en funksjon.



Ut i fra stolpediagrammet kan vi finne ut at prototypen vår mangler god funksjonalitet på punktene tilbakemelding, sikkerhet, tilpasningsmulighet, enkelhet og tilrettelegging. I en fullstendig utgave av systemet ville disse funksjonene blitt forbedret. Slik som prototypen er nå blir de satt til kritisk og vi er klar over manglene.

## 6.1 Evaluering

Vårt system har mer likheter med det kommende systemet til Fiskeridirektoratet enn det systemet som blir brukt i dag. I dagens system må fiskerne sende rapporter på en tungvint måte. Først fyller de ut rapporten, deretter lagrer de den og går inn på e-posten for å sende rapporten som vedlegg. I vårt system har vi lagt opp til at fiskerne skal bruke minst mulig tid på unødvendig arbeid. Når fiskerne logger inn på systemet kommer de til hovedsiden, hvor rapportutfylling er hovedvinduet. Dette fordi vi anser det som den viktigste funksjonen for fiskerne, og også den oppgaven de gjør oftest. Etter at de har fylt ut rapporten kan de trykke på knappen "send" og rapporten blir sendt. Når

rapporten blir sendt får brukeren en tilbakemelding på at rapporten er sendt. I motsetning til dagens system som sender en e-post tilbake til brukeren om at rapporten er mottatt. Vårt system vil også i større grad ta i bruk god estetikk ved grensesnittet. Ved design av prototyper er det viktig at designerne tenker på hvordan brukerne på en best mulig måte skal sitte igjen med en positiv opplevelse av systemet/produktet. Empiriske studier viser at grensesnittets estetikk kan ha en positiv innvirkning på hvordan folk oppfatter systemets brukervennlighet (Tractinsky, 2000 referert i Sharp, Rogers, Preece, 2007, s184).

Når vi var på Fiskeridirektoratet fikk vi vite at systemet de utvikler også skal ha en send-knapp, slik vi hadde tenkt. I tillegg skal det nye systemet forvente at fiskerne sender inn rapport en gang i døgnet, i motsetning til dagens system hvor de manuelle rapportene sendes hver syvende dag. Ettersom vi også hadde basert oss på at fiskerne skal sende inn rapport hvert døgn, og forenklet denne prosessen, føler vi at vi har klart å svare på oppgaven. Systemet vårt har flere likhetstrekk med det kommende, i tillegg til ekstra funksjonalitet og bedre brukervennlighet.

Tilleggsfunksjonene i vår programvare gjør systemet annerledes fra det som brukes i dag og det som er under utvikling. Vi ønsker at brukeren skal få et alt-i-et system, slik at de slipper å bruke flere systemer. Dette skal ikke være forhåndsinnlagt, men en mulighet for de som ønsker det, på den måten at de kan tilgjengeliggjøre de funksjonene den syns er viktig å ha, eller bruke sine eksisterende program.

Vi har fokusert på at det skal være enkelt for Fiskeridirektoratet å ha oversikten over rapporter som blir sendt inn, i tillegg til at det skal være mulig å oppdage feil og regelbrudd, samt manglende rapporter. I tillegg er det enkelt i vårt system og svare på en rapport til et enkelt fartøy, eller sende ut en beskjed til flere samtidig. Systemet vårt gir også en god oversikt over fartøy som befinner seg i det norske farvannet, med ulike fargekoder, alt etter deres funksjon eller mangler. På den måten vil vi gjøre det enklere for Fiskeridirektoratet å oppdage regelbrudd og avvik. Fiskeridirektoratet har ikke et slikt sammensett system nå. De bruker andre programmer enn fiskerne, noe vi har gått bort fra. Vi syns det er viktig at alle, fiskere, Fiskeridirektoratet og fiskemottak bruker samme system, designet for hver enkelt brukergruppe, med deres behov.

Slik prototypen vår er nå, er det ikke vist hvordan systemet kan brukes utenfor norske fiskebanker. Vi valgte å begrense oppgaven til å gjelde innenfor Norge, men mener at dersom systemet skulle blitt utviklet videre og brukt, hadde det vært en styrke dersom systemet kunne blitt brukt av

utenlandske fartøy. I tillegg til at norske fartøy kunne brukt systemet til å rapportere når de befinner seg i utenlandske soner. I dag må de nemlig rapportere til EU, når de krysser grensen. Vi mener det hadde vært mer brukervennlig om fiskerne ikke måtte forholde seg til mer enn ett system.

## 7.1 Konklusjon

Målet med oppgaven var å utvikle et system som svarer på problemstillingen som ble gitt i oppgaveteksten. Etter å ha fulgt instruksjonene gitt i Philips Weaver's, *Success in your project* (2004) føler vi at vi har oppnådd et gjennomført og godt resultat. Det finnes allikevel alltid forbedringspotensial. Dersom vi skulle utført oppgaven på ny ville vi lagt mer vekt på å utforme fyldigere og mer spesifikke skisser, slik at utviklingen med selve prototypen ville tatt mindre tid. Her kunne vi mer bevisst fokusert på å planlegge valg av ikoner, plasseringer, og liknende detaljarbeid. En annen fordel ville vært å få noen utenforstående til å teste systemet vårt. Vi måtte da laget en prototype hvor dette var mulig å gjennomføre. Funksjonene måtte altså vært mulig å utføre i vilkårlig rekkefølge. Hjelpesfunksjoner kunne også med fordel blitt laget til prototypen, som nevnt i prosjektplanen. Dersom vi hadde hatt tilstrekkelig kapasitet og kunnskap, hadde det vært en fordel å utforme databasen for prototypen, eventuelt laget programmet i Java. Skulle vi ha implementert en reell database for prototypen i Flash, kunne vi gjort dette ved bruk av MySQL og PHP, som kobler database og Flash sammen. Et endelig system, som ville vært utviklet i Java, hadde bare trengt å forholde seg til MySQL direkte i Javakoden, ved hjelp av Javabiblioteket.

I begynnelsen på prosjektet utformet vi en prosjektplan. Det har vist seg at vi måtte gjøre noen endringer i denne underveis, da den ikke svarte helt til det vi først så for oss. Vi har byttet litt om på arbeidsoppgaver, da noen av dem viste seg å være mer tidkrevende enn først antatt. Endringer oppstod også da vi fikk til møtet med Fiskeridirektoratet, og måtte legge til litt ekstra tid på research-delen.

Ut i fra at dette bare er en prototype, og vi har beskrevet mangler i rapporten, mener vi at systemet svarer på problemstillingen. Vi har dermed oppnådd ønsket resultat med prosjektet. Under arbeidet har vi som gruppe jobbet jevnt gjennom hele prosessen, og prøvd så godt det lot seg gjøre å ha en lik arbeidsfordeling. Samarbeidet har fungert godt, og vi har vært flinke til å løse problemer som har dukket opp underveis i utviklingsprosessen. Vi er godt fornøyd med vårt arbeid og svar på problemstillingen i dette prosjektet.

Nåværende systemer, som *SatRap*, ville blitt utkonkurrert av vårt system hadde vi hatt muligheten til å implementere det i sin helhet. Problemet er at det ville krevd enorme ressurser og en

omveltning i alle delene av fiskeriindustrien. Systemet ville også trenge utviklere og administratorer, som kunne holdt systemet ved like og oppdatert det. Dette ville gjort det utfordrende for systemet å få en fremtid. *SatRap* kombinert med eFangst vil nok fremdeles være det gjeldende systemet. FDS kunne gjort jobben enklere, mer effektiv og sammenfattet oppgavene, men som nevnt over ville det krevd mye ressurser.

## 8.1 Referanser og bibliografi

Heim, S. (2007) *The Resonant Interface: HCI Foundations for Interaction Design*. Long Island University, Addison Wesley.

Sharp, H. Rogers, Y. & Preece, J. (2007) *Interaction design: beyond human-computer interaction*. 2. utg. Wiley.

Weaver, P (2004) *Success in your project – a guide to student system development projects*. Prentice Hall, Financial Time.

Wikipedia (2009a) *Database modell* [Internett]

Tilgjengelig fra: <[http://en.wikipedia.org/wiki/Database\\_model](http://en.wikipedia.org/wiki/Database_model)> [Nedlastet 23.oktober 2009].

Wikipedia (2009b) *Semi-structured model* [Internett]

Tilgjengelig fra: <[http://en.wikipedia.org/wiki/Semi-structured\\_model](http://en.wikipedia.org/wiki/Semi-structured_model)> [Nedlastet 23.oktober 2009].

## 8.2 Inspirasjonsliste

<[www.fiskeridirektoratet.no](http://www.fiskeridirektoratet.no)>

<[www.fiskaren.no](http://www.fiskaren.no)>

<[www.fiskemottaket.no](http://www.fiskemottaket.no)>

<[www.fiskeribladetfiskaren.no](http://www.fiskeribladetfiskaren.no)>

<[www.fisknett.no](http://www.fisknett.no)>

<[http://retro.met.no/kyst\\_og\\_hav/fiskefelt1a.html](http://retro.met.no/kyst_og_hav/fiskefelt1a.html)>

<<http://www.piscus.no/index.php?cmd=artikkel&id=169>>

<<http://www.baat.aller.no/baatweb/cms.nsf>>

## 8.3 Bildereferanser

Bilder og logoer som er brukt i vedlegg 3d er hentet fra;

<http://www.bulandet-grendalag.org/fiskemottak.htm>

<http://www.fiskeridir.no/>

<http://www.fiskeri.no/Foto2007/Diverse%20September%20i2007.htm>

## 9.1 Ordliste

### **Aktivitetmeldinger:**

Et fiskefartøy kan enten melde seg aktiv eller passiv i en sone. Melder fartøyet seg aktiv vil det være det samme som å si at fartøyet aktivt fisker.

### **Autorapportering:**

En rapport som blir sendt automatisk fra systemet. Den er tidsinnstilt.

### **Begrensninger:**

Designprinsippet som går ut på at et godt design begrenser mulige handlinger som kan utføres og forhindrer at brukeren tar uriktige valg.

### **Fangstdagbok:**

Fiskefartøy er underlagt en rekke rapporteringsbestemmelser. En av disse er kravet om føring av fangstdagbok, en annen er innsending av fangst- og aktivitetsmeldinger.

Mange norske fiskefartøy omfattes av kravet om å føre fangstdagbok. Ikke så mange fartøy har krav om å sende fangst- og aktivitetsmeldinger til Fiskeridirektoratet så lenge de fisker i norsk sone. Men når norske fartøy fisker i utenlandske soner er de pålagt rapportering til ulike utenlandske mottakere.

### **Fangstkvote:**

Et visst antall kilo/tonn som det er lov å fiske av en spesiell art. Kvoteene blir regulert hvert år, justert etter artbestand og andre vilkår.

### **Fiskesoner:**

Norskekysten er delt opp i soner som det er lov å fiske i for norske fartøy. Utenlandske fartøy må melde seg aktive og søke om tillatelse for å fiske i de Norske sonene. Det samme gjelder for norske fartøy som fisker i utenlandske soner.

### **HCI-analyse:**

Human Computer Interaction. Brukergrensesnittsanalyse.

### **Konsistent:**

Designprinsippet som viser til at like oppgaver skal ha like operasjoner og elementer.

### **Rettledning:**

Designprinsippet som går ut på at en skal invitere brukeren til å utføre oppgaver på en riktig måte.

**Synlighet:**

Designprinsippet som går ut på å synliggjøre for brukeren hva som er det neste steget for å komme videre i et system.

**Tilbakemelding:**

Designprinsippet som går ut på å gi brukeren tilbakemelding på hva som har blitt gjort, og hva som er i ferd med å skje, samt å gi brukeren en mulighet til å fortsette aktiviteten.

**Usability goals:**

Brukervennlighetsmål

## 10.1 Vedlegg

Veldegg:

1. Kravspesifikasjoner
2. Målgrupper
3. Brukstilfeller
  - a. Fartøy
  - b. Fiskeridirektoratet
  - c. Mottak
  - d. FDS relasjoner
4. Databasemodell
  - a. SSM
  - b. Attributter
5. Prototypen
  - a. Fartøy
  - b. Fiskeridirektoratet
6. Prosjektplan
7. Prosjektlogg
8. Intervju med Guro
9. Skisser
  - a. Skissefartøy
  - b. Skissefiskeridirektoratet
  - c. skissemottak
10. Risikoliste
11. Project Initiation Document