Sensur av hovedoppgaver

Høgskolen i Sørøst-Norge
Fakultet for teknologi og maritime fag

Prosjektnummer: **2016-02**
For studieåret: **2015/2016**
Emnekode: **SFHO3201**

Real Time Observation
Sanntid Observasjon

Utført i samarbeid med: Kongsberg Defence & Aerospace AS

Ekstern veileder: Arnt-Levi Alstad

Sammendrag: Kongsberg Defence & Aerospace har behov for å se på hvordan ny og fremtidsrettet teknologi kan anvendes innenfor deres forskjellige observasjonssystemer. Oppgaven går derfor ut på å se på nye innovative løsninger for hvordan bilde og informasjon fra et videokamera skal nå operatøren av systemet.

Stikkord:
- Software
- Video
- Server - Client

Tilgjengelig: DELVIS

Prosjekt deltagere og karakter:

Navn	Karakter
Vegard Smedsvik	
Svein Bekkevold	
Alexander Schinnes	
Carl Peder Jansen	

Dato: 9. Juni 2016

_____	_____	_____
Joakim Bjørk	Karoline Moholth	Arnt-Levi Alstad
Intern Veileder	Intern Sensor	Ekstern Sensor

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

TILGJENGELIGHET

Dette prosjektet er DELVIS tilgjengelig som innebærer følgende:

- All dokumentasjon i prosjektet er tilgjengelig for allmennheten
- Kildekoden er IKKE tilgjengelig og skal ikke publiseres videre

Retningslinjer er satt av oppdragsgiver: Kongsberg Defence & Aerospace AS den 19.05.2016

CD med kildekode vil bli levert separat fra dokumentasjonen.

INNHOLDSFORTEGNELSE

1. Visjonsdokument
2. Prosjektplan
3. Brukermanual
4. Teknologidokument
5. Designdokument
6. Analysedokument
7. Risikodokument
8. Kravspesifikasjon
9. Testspesifikasjon
10. Kommunikasjon
11. Etteranalyse
12. Testrapport
13. Iterasjonsplaner
14. Iterasjonsrapporter

Høgskolen i Sørøst-Norge

KONGSBERG

GRUPPE 2

VISJONSDOKUMENT – v 1.0

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENT 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 DOKUMENTHISTORIE 4
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 INNLEDNING 4
3.1 PROSJEKTETS DELTAKERE 5
4 PROBLEMBESKRIVELSE 6
4.1 OVERORDNET PRODUKTBESKRIVELSE 6
5 INTERESSENTER OG BRUKERE 7
5.1 INTERESSENTER 7
5.2 BRUKERE 7
5.3 BRUKERBEHOV 7
6 PRODUKTET 8
7 FORUTSETNINGER 8
8 RAMMER 9
VEDLEGG 9
TIDLIGERE DOKUMENTHISTORIKK 9
REFERANSER 9

1 DOKUMENT

Dette dokumentet inneholder visjonen for systemet Real Time Observation.

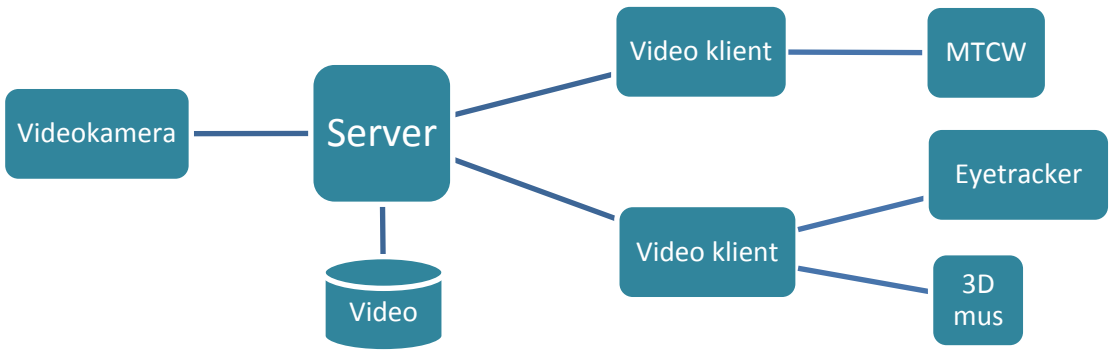
1.1 OVERSIKT

Dokumentet gir en grov beskrivelse av systemet og danner et grunnlag for en mer detaljert teknisk beskrivelse. Her finner vi en oversikt over interessenter, brukere, forutsetninger og rammer for prosjektet. Gruppen har fått tildelt hovedoppgave av Kongsberg Defence & Aerospace AS som gir grunnlag for ingeniørbachelor hos Høgskolen i Sørøst-Norge. Det er fire deltagere som danner gruppen, hvor alle går elektroingeniør. Interessenter i oppgaven er HSN med intern sensor/veileder og KDA med ekstern sensor/veileder.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 – Systemoversikt

1.3 DOKUMENTHISTORIE

Se vedlegg for tidligere dokumenthistorikk.

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
1.0	09.03.2016	Carl Peder Jansen	Dokumentgjennomgang før andre presentasjon

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man to Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 4
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4
Tabell 4 - Deltakerinfo 5
Tabell 5 - Interessenter 7
Tabell 6 - Dokumenthistorikk versjon 0 9

2.4 FIGURER OG BILDER

Figur 1 – Systemoversikt 3

3 INNLEDNING

Real Time Observation, heretter benevnt RTO, er et system utviklet for Kongsberg Defence & Aerospace. RTO er et hovedprosjekt som avslutning for bachelorgrad ved Høgskolen i Sørøst-Norge.

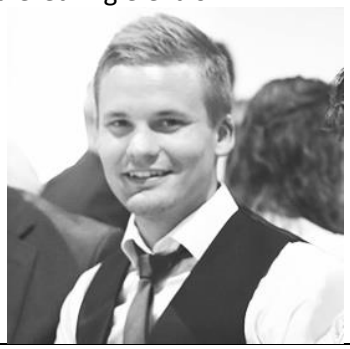
Dokumentets hensikt er å gi en grunnleggende beskrivelse av hvilket problem RTO skal løse, hvordan det skal løses og på hvilket grunnlag. Dokumentet skal også fortelle litt om hvilke forutsetninger og rammer prosjektet må forholde seg til, og hvilken målsetning prosjektgruppen har.

3.1 PROSJEKTETS DELTAKERE

RTOs forskning og utviklingsgruppe består av 4 deltakere med ulik bakgrunn. Felles for alle er en genuin interesse for å løse tekniske problemer, både praktisk og analytisk. Gruppens fagkompetanse består av elektro med spesialisering innen mekatronikk og kybernetikk.

Tabell 4 - Deltakerinfo

Deltaker	Info
Vegard Smedsvik Studieretning elektro 	**Rolle:** Prosjektleder **Bakgrunn:** Romfart og teknologi **Kontaktinformasjon:** Email: vsmedson@gmail.com Tlf: 47965007
Carl Peder Jansen Studieretning elektro 	**Rolle:** Testansvarlig **Bakgrunn:** Fagbrev automasjon **Kontaktinformasjon:** Email: carl.p.jansen@gmail.com Tlf: 47896477

Svein Bekkevold Studieretning elektro 	**Rolle:** Dokument Controller **Bakgrunn:** Tømrer Forsvaret /Hæren **Kontaktinformasjon:** Email: bekkevoldsvein@gmail.com Tlf: 93223115
Alexander Schinnes Studieretning elektro 	**Rolle:** System arkitekt **Bakgrunn:** Fagbrev elektriker **Kontaktinformasjon:** Email: aschinnes@gmail.com Tlf: 47264142

4 PROBLEMBESKRIVELSE

Oppdragsgiveren Kongsberg Defence & Aerospace har behov for å se på hvordan ny og fremtidsrettet teknologi kan anvendes innenfor deres forskjellige observasjonssystemer. Oppgaven går derfor ut på å se på nye og innovative løsninger for hvordan bilde og informasjon fra et videokamera skal nå operatørene av systemet. Systemet skal bestå av et kamera på utsiden som overfører videobildet via HDMI-kabel inn til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren for så å kunne avspilles ved en senere anledning.
Det skal også undersøkes nye løsninger for hvordan et motorisert kamera kan styres av en operatør med ulike MMI-enheter. De forskjellige MMI- enhetene er: Eye-tracker, 3D-mus og Multi-Touch Control Wheel. Et av hovedkravene er at vi skal lage en user interface i Qt Quick som skal kunne styre kameraet og tilrettelegge bruken av overlays.

4.1 OVERORDNET PRODUKTBESKRIVELSE

Produktet som skal utvikles må møte KDAs behov i henhold til problemstilling og overholde krav gitt av kravspesifikasjonen.

RTO skal være et brukervennlig verktøy med de funksjoner som kreves for å tilfredsstille kravspesifikasjonen, gitt de forutsetninger som er satt.

5 INTERESSENTER OG BRUKERE

Vår hoved interessent er Kongsberg Defence & Aerospace AS, ettersom de har fremstilt kravene for oss med ønske om en teknisk løsning for problemet. KDA er også eneste økonomiske interessent med en økonomisk ramme gitt til RTO.

Høgskolen i Sørøst-Norge (HSN) er en annen viktig interessent, selv om de ikke har brukere av systemet. Systemet vil bli vurdert av HSN i samspill med KDA.

5.1 INTERESSENTER

Tabell 5 - Interessenter

Navn	Rolle under prosjektet	Representant	Kontakt
Kongsberg Defence & Aerospace AS	Oppdragsgiver og ekstern sensor for vurdering av sluttprodukt og system	Arnt-Levi Alstad	Tlf: 92401588 Email:Arnt.Levi.Alstad@kongsberg.com
Kongsberg Defence & Aerospace AS	Ekstern veileder for RTO. Bistår med sin kunnskap, deltar på møter og følger opp.	Rune Værnes	Tlf: 99004209 Email:rune.flaten.vaernes@kongsberg.com
Kongsberg Defence & Aerospace AS	Oppdragsgiver og avdelingsleder.	Baard Johan Trooye	Tlf: 93057283 Email:baard.johan.troye@kongsberg.com
Høgskolen i Sørøst-Norge	Intern Veileder. Deltar på møter med RTO, gir veiledning og følger opp internt.	Joakim Bjørk	Tlf: 31009406 Email: Joakim.Bjork@hbv.no
Høgskolen i Sørøst-Norge	Intern sensor for vurdering av sluttprodukt og prosjekt.	Karoline Moholt	Tlf: 31008898 Email: Karoline.Moholt@hbv.no

5.2 BRUKERE

Operatører av systemet vil sitte ved en klient hvor de vil ha tilgang til alle funksjoner beskrevet i systembeskrivelsen. Tester og bruk vil bli utført på en stasjon innendørs.

5.3 BRUKERBEHOV

KDA har ønske om at produktet kan være en del av et større system som kan brukes i fremtiden. Ved utviklingen må vi derfor tenke på gjenbruk og at det skal være mulig å bytte ut deler av systemet mot oppgradert programvare, hardware og lignende.

6 PRODUKTET

Det skal lages en test applikasjon som presenterer HD-video fra en videoserver, hvor videoformatet skal være H.264. Det skal lages et «overlay» som oppdateres med input fra de forskjellige MMI-enhetene nevnt nedenfor.

Det skal være en funksjon for å ta opptak av video, samt avspille denne på et senere tidspunkt. I tillegg, lage en videoserver som sender data til sine klienter for presentasjon av video.

Det skal lages en API mot enhetene som gjør det mulig å integrere dem inn i en applikasjon.

Enheter:

- Picolo HD H.264 Videokort
- MTCW (Multi Touch Control Wheel)
- Tobii Eye Tracker
- 3D mouse space navigator
- Axis V5915 HD videokamera

7 FORUTSETNINGER

Forutsetninger fra KDA:

- Kostnader vedrørende videokamera 10.000,- til 20.000,-
- Det bør anvendes open source løsninger under koding
- Rammeverket skal være i Qt Quick

Forutsetninger fra RTO:

- Utviklingstid til ferdig system, ca 5mnd
- Det bør anvendes open source løsninger under koding
- Det kan benyttes eksterne tjenester (retting av dokumentasjon, teknisk hjelp osv.)

RTO skal ha følgende egenskaper:

- Funksjonalitet for å ta opptak av video
- Spille av video på et senere tidspunkt
- Funksjon for «overlays» over bildet
- Motorstyring av kamera (Pan, tilt og zoom)
- Kommunikasjon via videoserver

8 RAMMER

Prosjektet har visse rammer som ikke er mulig å endre.

- Prosjektet må være ferdigstilt og levert HSN innen 17.05.2016, arbeid etter dette teller kun for den avsluttende presentasjonen sin del.
- Fremvisningen av prosjektet er delt opp i 3 forskjellige presentasjoner.
- HSNs rammer for hovedoppgave.
- KDA står for utlevering av utstyr samt økonomiske rammer som er avtalt i kontrakten.
- HSN tilbyr grupperom og lab arealer.
- Kontrakt med HSN
- Kontrakt med KDA

VEDLEGG

TIDLIGERE DOKUMENTHISTORIKK

Tabell 6 - Dokumenthistorikk versjon 0

Versjon	Dato	Skrevet av	Beskrivelse
0.1	05.02.16	Svein Bekkevold	Første utkast
1.0	09.03.16	Carl Peder Jansen	Dokumentgjennomgang før andre presentasjon

REFERANSER

[1] Olaf Hallan Graven, *Veiledning for prosjekt 2016*, 05.02.16
[2] Bacheloroppgave Tangstad 2015

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

PROSJEKTPLAN – v 2.0

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENT 4

1.1 OVERSIKT 4

1.2 SYSTEMOVERSIKT 4

1.3 HISTORIKK 5

2 FIGURER OG TABELLER 5

2.1 AKRONYMER OG FORKORTELSER 5

2.2 ORDBOK 5

2.3 TABELLER 5

2.4 FIGURER OG BILDER 6

3 MÅL OG RAMMER 7

3.1 BAKGRUNN 7

3.2 PROSJEKTMÅL 7

3.3 RAMMER 7

4 OPPGAVESPESIFIKASJON 8

5 PROSJEKTORGANISERING 9

5.1 ANSVARSOMRÅDER 9

5.2 ØVRIGE ROLLER I PROSJEKTET 9

6 PROSJEKTERING OG GJENNOMFØRING 10

6.1 PROSJEKTMODELL 10

6.2 SCRUM-MØTER 10

6.3 FASER 11

6.3.1 ITERASJONER 12

6.3.2 FASE 1 - «INCEPTION» 12

6.3.3 FASE 2 - «ELABORATION» 13

6.3.4 FASE 3 - «CONSTRUCTION» 14

6.3.5 FASE 4 - «TRANSITION» 15

6.4 PLANLAGT GJENNOMFØRING 15

6.4.1 FASE 1 - «INCEPTION» 15

6.4.2 FASE 2 - «ELABORATION» 16

6.4.3 FASE 3 - «CONSTRUCTION» 16

6.4.4 FASE 4 - «TRANSITION» 17

7 OPPFØLGING OG RAPPORTERING 18
7.1 OPPFØLGINGSDOKUMENT 18
7.2 TIMELISTER 18
7.3 MØTESTRUKTUR 18
7.3.1 MØTE MED INTERN VEILEDER 18
7.3.2 MØTE MED EKSTERN VEILEDER 18
7.4 NETTSIDE 19
8 GJENNOMFØRING 19
8.1 ADMINISTRATIVE 19
8.2 MILEPÆLER 20
8.3 TIDS –OG RESSURSPLANER 20
8.4 KOSTNADER 20
VEDLEGG 21
AKTIVITETSLISTE 21
TIDLIGERE DOKUMENTHISTORIKK 27
REFERANSER 27

1 DOKUMENT

Dette dokumentet omfatter en prosjektplan for systemet Real Time Observation.

1.1 OVERSIKT

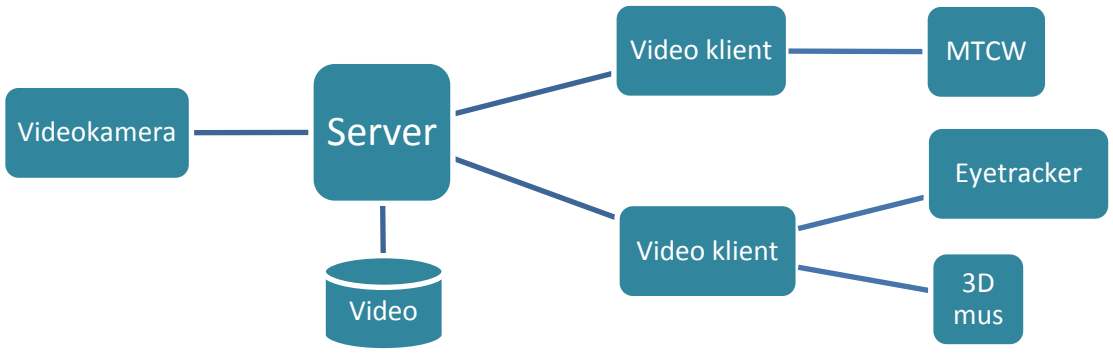
Prosjektplanen er et nyttig verktøy som skal gi oss en plan på hvordan vi skal jobbe mot et ferdig produkt som tilfredsstiller kravene fra arbeidsgiveren. Vår prosjektplan inneholder generelle mål og rammer for prosjektet, en beskrivelse av oppgaven, hvilke roller som finnes i prosjektet, og hvordan vi ser for oss å gå frem for å løse oppgaven. Det er også utarbeidet en tidsplan, som viser hvilke aktiviteter vi skal jobbe med til hvilken tid. Denne gir oss en god oversikt over ressursforbruket.

Det er først og fremst prosjektleder som jevnlig benytter seg av prosjektplanen for å følge opp at alle aktiviteter blir utført til rett tid, men det er også et meget godt verktøy for alle andre som er involvert i prosjektet. Her kan man få et godt overblikk over alt som foregår. Man oppdager om man ligger etter tidsplanen, og hvilke konsekvenser det da eventuelt kan få. Ved aktivt å benytte seg av prosjektplanen vil man kunne holde en veldig god flyt gjennom alle faser av prosjektet. Leverandører og kunder kan også holde seg oppdatert på hvordan prosjektet ligger an i forhold til tidsplanen. Prosjektplanen er en viktig informasjonskanal for alle parter.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 - Systemoversikt

1.3 HISTORIKK

Se vedlegg for tidligere dokumenthistorikk.

Tabell 1 - Dokumenthistorikk

Versjon	Dato	Skrevet av	Beskrivelse
2.0	19.05	Vegard	Gjennomgang, oppdatering

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorikk 5
Tabell 2 - Akronymer og forkortelser 5
Tabell 3 - Ordbok 5
Tabell 4 - Stillinger 9
Tabell 5 - Administrative aktiviteter 19
Tabell 6- Milepæler 20
Tabell 7 - Budsjettering 20
Tabell 8 - Dokumenthistorikk versjon 0 27

2.4 FIGURER OG BILDER

Figur 1 - Systemoversikt 4
Figur 2 - System 8
Figur 3 - Faseoversikt 11

3 MÅL OG RAMMER

3.1 BAKGRUNN

Vi er en gruppe på fire elektrostudenter som går siste år av en treårig bachelorgrad innenfor kybernetikk og mekatronikk på Høgskolen i Sørøst-Norge, avdeling Kongsberg (heretter kalt HSN), som skal gjennomføre en avsluttende hovedoppgave som vil gjøre oss best mulig forberedt på arbeidslivet som ingeniør. Prosjektoppgaven har vi fått av Kongsberg Defence & Aerospace AS (heretter kalt KDA).

Kunnskaper innen programmering og software blir en viktigere og viktigere del av hverdagen til ingeniører, dette er noe av grunnen til at vi har valgt denne oppgaven. Vi mener at denne oppgaven vil utvide vår breddekunnskap og innsikt innenfor elektronikk, som vil gjøre oss i stand til å løse oppgaver bedre i teknologibransjen.

3.2 PROSJEKTMÅL

- Målet med prosjektet er å lage et produkt som tilfredsstiller kravene gitt av KDA på en god og ryddig måte.
- Møte oppgaver som utfordrer de kunnskapene vi har, slik at vi kan utvikle oss praktisk og samtidig få en bedre teoretisk forståelse.
- Lage et produkt som er nyttig og innovativt som kan forenkle videre arbeid og utvikling for KDA.
- Holde oss til prosjektmodellen og de tidsplaner som er lagt.
- Lære å jobbe bedre som gruppe og utvikle samarbeidsevnene videre.
- Takle uforutsette problemer og uhell som kan oppstå på en profesjonell og effektiv måte.

3.3 RAMMER

Rammene rundt prosjektet er satt av HSN og KDA i form av tidsrammer, tidspunkter og kostnader. Presentasjonen av prosjektet er delt opp i tre deler, med en første presentasjon, andre presentasjon og en avsluttende tredje presentasjon. Frist for innlevering av selve prosjektet med all tilhørende dokumentasjon er satt av HSN til å være tirsdag 17. mai 2016. Arbeid etter denne tid teller kun for den avsluttende presentasjonen sin del.

KDA står for utlevering av alt materiell og utstyr samt de økonomiske rammer som er avtalt i kontrakten. HSN tilbyr grupperom og lab arealer.

4 OPPGAVESPESIFIKASJON

Oppdragsgiveren, Kongsberg Defence & Aerospace AS, har behov for et system som kan erstatte nåtidens overvåkning innen maritime applikasjoner.

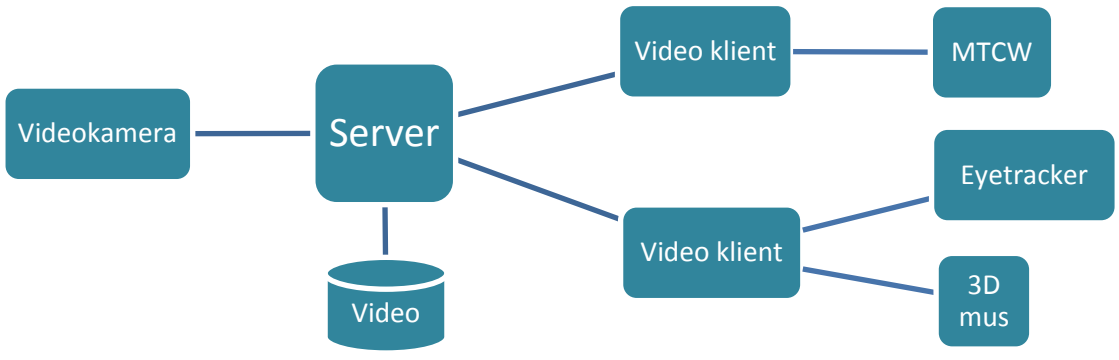
KDA har kommet med et forslag til hvordan de ser for seg dette produktet. Det skal bestå av et motorisert kamera som overfører videobildet via HDMI-kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Videoformatet skal være H.264.

Vi skal lage både videoserveren og en applikasjon på klientmaskinene som skal kunne kommunisere med serveren og vise videostrømmen. Applikasjonen skal lages i Qt Quick med programmeringsspråkene C++ og QML. Det er her ønskelig med lavest mulig forsinkelse på videostrømmen fra kamera til klient slik at opplevelsen for operatørene i klientapplikasjonen er instant og øyeblikkelig i forhold til virkeligheten. Beregninger på faktisk forsinkelse skal kunne beregnes, det samme med datatrafikk, prosessorkraft og minneforbruk. Det skal også gjøres detaljert analyse av løsninger, eventuelle alternativer, videostrøm, nettverk, kvalitet, enkoding/dekoding og lignende.

Klientene skal kunne styres med tre MMI-devicer som er utlevert fra KDA. Dette er en Tobii Eye Tracker, et Multi Touch Control Wheel (MTCW) og en 3D Connection Space Navigator mus. Det skal lages en felles API for enhetene som enkelt skal kunne integreres i Qt Quick applikasjoner. For MTCW må det også bli laget en driver for Windows 7 64-bit. Det er meningen at man skal kunne styre brukergrensesnittet på klientene ved hjelp av disse enhetene. I tillegg er det et ønske om å kunne styre motoren som skal være på kameraet ved hjelp av for eksempel eye trackeren eller en av de andre enhetene.

Videoavspillingen av både lagret video og video i nåtid skal være plassert i applikasjonen til klientene. Det må da lages et funksjonelt brukergrensesnitt for videoavspilling i tillegg til at det skal være mulig å legge et overlay over videoen. Dette overlayet bør kunne legge til effekter på videoen, som endring av farge og kontrast, og vise detaljert datainformasjon om videostrømmen.

Figur 2 - System



5 PROSJEKTORGANISERING

5.1 ANSVARSOMRÅDER

Oppdragsgiver: Kongsberg Defence & Aerospace AS
Prosjektleder: Vegard Smedsvik

Tabell 4 - Stillinger

Stilling	Beskrivelse	Gruppemedlem
Prosjektleder	Ansvar for at arbeidet utføres i henhold til kravene og at prosjektet gjennomføres etter planen og de tidsfrister som er satt. Fordeler oppgaver og følger opp gruppen.	Vegard Smedsvik
Økonomiansvarlig	Ansvar for alle økonomiske kostnader i prosjektet og sørger for at budsjetter overholdes.	
Systemarkitekt	Har ansvar for at utviklingen av software og kode bygges opp og utføres på en god og tilfredsstillende måte. Skal ha god oversikt og kjennskap til alle tekniske ledd i softwarestrukturen.	Alexander Schinnes
Testansvarlig	Har ansvar for alle ledd i forbindelse med testing av produktet. Testspesifikasjoner, testrapporter samt gjennomføringen av alle tester skal testansvarlig følge opp.	Carl Peder Jansen
Dokumentansvarlig	Har ansvar for at all dokumentasjon følges opp og utføres etter gode standarder. Passer på at maler følges. Setter dokumenter sammen og utfører dokumentkontroll.	Svein Bekkevold

5.2 ØVRIGE ROLLER I PROSJEKTET

Ansvarlig kontaktperson KDA:

- Baard Johan Troye

Eksterne veiledere:

- Arnt-Levi Alstad
- Rune Flaten Værnes

Intern veileder:

- Joakim Bjørk

6 PROSJEKTERING OG GJENNOMFØRING

6.1 PROSJEKTMODELL

Etter å ha satt oss godt inn i hva oppgaven går ut på, dens omfang, rammer og tidsfrister, så valgte vi prosjektmodell. Vi vurderte først å gå for en noe modifisert spiralmodell, men fant raskt ut at "Unified Process" – modellen ville passe prosjektet vårt enda bedre. Det er mange grunner til dette. Vi forventer at det vil oppstå uventede situasjoner underveis i prosjektet, som det ofte pleier å gjøre. "Unified Process" er en relativt fleksibel prosjektmodell med tanke på at man kan gjøre en del endringer i prosjektet, ettersom ting oppstår underveis. Samtidig er denne prosjektmodellen veldig enkel og klar å bruke når man skal jobbe mot bestemte mål, delmål og tidsfrister. Vi skal holde tre presentasjoner underveis i prosjektet, dette blir våre delmål og milepæler i tillegg til innlevering av hele prosjektet med all dokumentasjon den 17. mai.

Det er klare tidsfrister og aktiviteter å forholde seg til når man bruker denne modellen, det liker vi meget godt. Å arbeide mot konkrete mål under tidspress er den mest effektive formen å jobbe på for de aller fleste, inkludert oss. Gjennomføringen av selve prosjektet deles opp i faser med flere iterasjoner innenfor hver fase. Dermed har man alltid oversikt over hvor man er i prosjektet, hvilke oppgaver som skal gjøres, når de skal avsluttes og hvilke oppgaver som venter. Skulle man plutselig havne bakpå, oppdages dette raskt slik at man kan gjøre nødvendige tiltak.

6.2 SCRUM-MØTER

Vi har også valgt å hente inn et element fra prosjektmodellen «Scrum». Det er «scrum-møter» som skal holdes stående hver morgen. Møtet skal foregå effektivt og gi alle i møtet en kort oversikt over hvordan vi ligger an i prosjektet, samt hvilke oppgaver som skal prioriteres for dagen. Alle deltakere kan komme opp med spørsmål ved behov. Vi har også valgt å gå gjennom tre faste punkter per gruppemedlem hvert møte. Dette er:

- Hva jobber du med nå?
- Hva skal du gjøre frem til neste møte?
- Er det noen spesielle problemer eller utfordringer?

Her gir man en kort status på nevnte punkter. Hvis man står fast på oppgaver eller større utfordringer er dette fint å få avklart, slik at andre gruppemedlemmer kan hjelpe til hvis de har noe kunnskap å tilføre på dette området. Misforståelser blir også raskt oppfattet og rettet opp i. Møtet skal foregå i maksimalt 15 minutter. Etter erfaring så sparer vi mer tid i prosjektet enn vi bruker på å holde disse møtene.

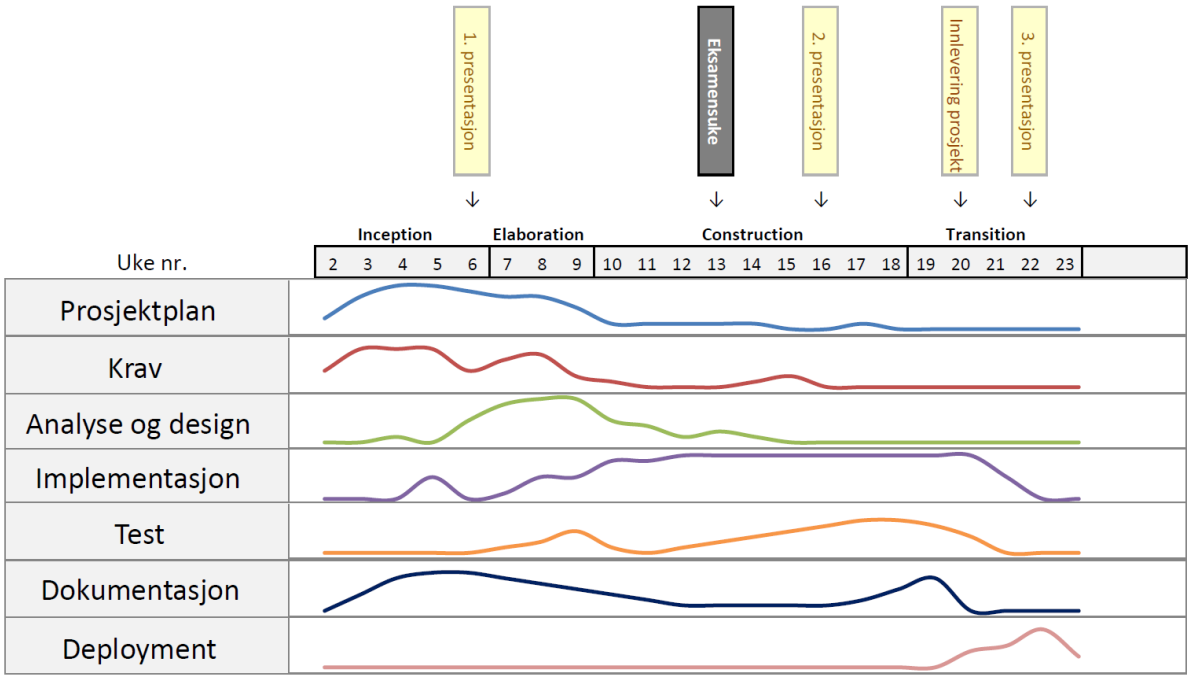
6.3 FASER

«Unified Process» er en prosjektmodell som baserer seg på fire faser som en grov inndeling for hva som prioriteres på overordnet nivå til hvilken tid. Disse fire fasene består av «Inception», «Elaboration», «Construction» og «Transition». «Inception» går ut på å samle inn all relevant info og komme smått i gang med dokumentasjon og det overordnede systemet. «Elaboration» er fasen hvor man tar utgangspunkt i det man gjorde i første fasen, for så å spikre dette mer fast og legge et godt arbeidsgrunnlag for neste fase med tanke på design og analyse. «Construction» er den største og viktigste fasen. Her bygger man videre på det grunnlaget man la i forrige fase og implementerer alt til et godt fungerende system. «Transition» er overleverings – og ferdigstillelsesfasen. Se punkt 6.3.2 til 6.3.5 for mer utfyllende informasjon.

Figur 6.3.1 viser en overordnet tidsplan for prosjektet. Her vises arbeidstype i kolonnen helt til venstre, mens faser og ukenummer fremkommer i den øvrige raden. Det er i tillegg lagt inn milepæler øverst. For oss er dette de tre presentasjonene, eksamensperioden og frist for innlevering av prosjektet med komplett dokumentasjon.

De fargede linjene som fremkommer av figuren viser prioritet på de forskjellige arbeidstypene avhengig av hvilken fase og periode vi er i. Er grafen i øvre nivå vil det si at vi arbeider mye med denne aktiviteten i den perioden. Er grafen i nedre nivå vil det si at vi jobber lite, eller ingenting, med denne aktiviteten i denne perioden. Det er også lagt inn en liten buffer på slutten hvor det er lite aktivitet. Alt som eventuelt skulle gjenstå blir da fortløpende prioritert etter behov.

Figur 3 - Faseoversikt



6.3.1 ITERASJONER

Gjennom hver fase vil vi ha flere iterasjoner. En iterasjon kan kalles et slags mini-prosjekt som er en del av hele fasen. Man velger seg ut ulike delsystemer eller temaer man arbeider med i de forskjellige iterasjonene. Hver iterasjon bør utføres systematisk. Vanligvis er man innenfor disse fem temaene i følgende rekkefølge:

Det vil være ulikt hvilke elementer man bruker mest tid på avhengig av hvilken fase man er innenfor i prosjektet. I «elaboration» fasen vil det naturligvis gå med mye tid til krav og analyse, mens det i «construction» fasen vil være stort fokus på design, implementasjon og test. Hver iterasjon vil ende opp med en ny versjon av applikasjonen når vi har inntatt den utførende fasen. Denne versjonen vil da inneha mer og bedre funksjonalitet sammenlignet med forrige ukes versjon.

6.3.2 FASE 1 - «INCEPTION»

Dette er fasen hvor hele prosjektet starter. Man kan grovt kalle det en «brainstormings fase». Hovedmålet med denne er å etablere en god forståelse for det planlagte produktet eller systemet. Man ser på systemet som en «black-box» og definerer innganger og resulterende utganger. Altså, hva skal skje hvis vi gjør «det og det». Man henter inn så mye informasjon som mulig i fra kunden. Dette kan komme i skriftlig format, eller bli omtalt i møter og telefonsamtaler. Man må samle inn nok info til at man får et godt overblikk over den faktiske oppgaven, og hva denne går ut på. Når oppgavedefinisjonen er relativt klar, må prosjektgjennomføringen avklares. Da finner man ut hvordan man skal løse oppgaven best mulig som gruppe. Det må lages tidsplaner med frister og milepæler, kravspesifikasjoner, testspesifikasjoner, aktivitetsliste med mer.

Fasens målsetninger er:

- Få alle berørte parter og kunder til å være enige om rammen og omfanget av prosjektet og produktet som er foreslått.
- Få på plass alle kritiske og viktige høy-nivå krav.
- Prosjektplanen og de økonomiske rammene er på plass slik at vi kan få grønt lys fra alle parter til å fortsette utviklingen av produktet.

Vi har delt opp «inception» fasen i to iterasjoner. I1 og I2. Iterasjon 1 er oppstartsperioden som vi har satt til å vare i tre uker, altså 15 arbeidsdager. Dette er en noe spesiell periode fordi det er mye som skjer på en gang. Gruppen må etableres, oppgaven spesifiseres, materiell og utstyr må hentes og

monteres, grupperom må reserveres også videre. Alt dette tar tid og det vil være en utfordring å kunne jobbe systematisk i denne perioden. Derfor er den satt til 15 arbeidsdager.

De resterende iterasjonene skal ha en varighet på en uke hver, altså 5 arbeidsdager. I realiteten vil dette være 3 til 4 arbeidsdager, fordi alle gruppens medlemmer har et annet fag å forholde seg til, frem til og med påske. Men for ordens skyld omtaler vi en uke som 5 arbeidsdager. Grunnen til at vi har valgt dette er at vi ønsker relativt korte iterasjoner som er enkle å forholde seg til. Det er lett å beholde god oversikt over iterasjonen så lenge den ikke strekker seg lenger enn den aktuelle arbeidsuken man befinner seg innenfor. Skulle det være slik at en ikke blir ferdig med oppgavene som er planlagt for iterasjonen i løpet av disse 5 arbeidsdagene, har en lørdag og søndag til å redde seg inn på ved behov. Ved å da benytte seg av helgen kan en slippe å endre statuser, tidsplaner og opprette tiltak fordi en aktivitet ikke ble ferdig. Det er en av de store fordelene med å bruke iterasjoner basert på uker.

Iterasjon 2 går ut på å forsøke å lande krav for system og ferdigstille dokumenter så langt det lar seg gjøre, men det vil sannsynligvis kunne oppstå situasjoner i neste fase som gjør at disse dokumentene da må endres. Dokumentene vi prioriterer å ferdigstille er kravspesifikasjonen, testspesifikasjonen og prosjektplanen som skal inneholde prosjektmodell, use caser, risikovurdering og andre administrative nødvendigheter. Vi vil også jobbe med forberedelser til første presentasjon. Vi tenker å bruke tilstrekkelig med tid for å lage en god og oversiktlig presentasjon. Alle dokumenter skal i utgangspunktet være ferdige nå, og vi kan fokusere på å sy alle dokumenter sammen og ferdigstille prosjektplanen for innlevering innen onsdag. Onsdag og torsdag vil gå med til å lage presentasjon, samt øve på denne. Fredag er presentasjonsdag, mye tid vil da gå med rundt dette. Det er også et ønske fra gruppen å kunne vise frem en prototype på hvordan vi ser for oss brukergrensesnittet. Dette vil da være et veldig grovt produkt med noe helt enkel funksjonalitet. Men hvis vi oppdager at tiden ikke strekker til, vil dette bli utsatt til neste fase. Denne iterasjonen varer i to uker.

6.3.3 FASE 2 - «ELABORATION»

Nå er vi forbi den grove begynnelsesfasen, og det er på tide å legge et mer fast grunnlag for å lage systemet. Kravspesifikasjoner og prosjektplan vil bli jevnlig vurdert og eventuelt endret hvis dette er nødvendig. Vi forsøker å fastslå så mange krav og planer som mulig, slik at vi allerede nå er relativt sikre på hvilken vei prosjektet går og hvilket produkt vi kommer til å ende opp med. Men vi kommer også til å være åpne for å flytte på tidsfrister og planer underveis hvis vi ser at det ender opp med et bedre sluttresultat og en god flyt.

Vi har valgt å ha en relativt kort og effektiv «elaboration» fase som skal strekke seg over tre uker. Det skal analyseres og utføres mye systemdesign i denne perioden, som skal legge grunnlaget for løsningene vi skal implementere i «construction» fasen. Det er ønskelig å få slått fast hvilke løsninger vi skal bruke for alle delsystemer og systemet i sin helhet. Vi tror at vi kan spare en del tid i neste fase ved å gjøre god «research» på alle løsninger og finne eventuelle fordeler og ulemper ved disse løsningene så tidlig som mulig. Da er vi bedre forberedt på situasjoner som kan oppstå senere, og har et bedre grunnlag for å gjøre gode beslutninger.

Når det nærmer seg enden av fasen vil vi også begynne med noe implementasjon og test av delsystemer. Alt som blir ferdig i denne fasen, skal gjennom en testprosedyre for å sikre at den aktuelle delen er klar for videre implementasjon og integrasjon. Også i denne fasen skal vi ha ukes iterasjoner som strekker seg over fem arbeidsdager. Totalt vil fasen da inneholde tre iterasjoner. Første iterasjon vil bli evaluering av alt som er oppnådd i første fase for å sikre at vi er på rett vei. De resterende to iterasjonene vil det bli stort fokus på design og analyse av system og delsystemer. Mer detaljerte aktiviteter innenfor hver iterasjon vil bli lagt når fasen påbegynnes.

Fasens målsetninger er:

- De fleste funksjonelle krav skal være på plass for det nye systemet og skal være beskrevet med use case modeller.
- Arkitekturen og rammeverket, i en liten forenklet utgave, for produktet skal være på plass og legge et solid grunnlag for videre utvikling.
- Prosjektet får grønt lys av alle parter for videre arbeid, og gruppen har en klar prosjektplan som beskriver hvordan «construction» fasen skal fortsettes.

6.3.4 FASE 3 - «CONSTRUCTION»

Dette er den aller største og viktigste fasen i prosjektet. Det er her produktet skal ta form og ting skal skje. Fasen er planlagt å vare i totalt 9 uker, men det er viktig å påpeke at innenfor denne fasen er det en uke som går med til eksamensforberedelser og en snau uke som går med til påskeferie. I praksis har vi da 7 iterasjoner med totalt 35 arbeidsdager. Vi fortsetter med ukes iterasjoner også i denne fasen, som alle andre faser.

Vi vil fortsette å bygge på alt grunnlaget vi har til nå, samt delsystemer som har blitt implementert gjennom forrige fase. Funksjonaliteten vil bli utvidet slik at flest mulig av kundens krav skal bli oppfylt. Fokuset er stort på å få et fungerende system uten større feil og mangler. Generelt blir det utført langt mer implementasjon på dette punktet enn i forrige fase. Detaljert innhold i iterasjonene vil bli satt i begynnelsen av fasen, men det er planlagt å holde av hver fredag til test og fiksing av feil. På denne måten vil vi holde testingen og testdokumentasjonen ved like, samtidig som vi vil ha et system og en kode med minst mulig feil ved arbeids start på mandag uken etter. All dokumentasjon skal vedlikeholdes på et godt nivå gjennom hele denne fasen, slik at vi ikke trenger å sette av så mye tid til dette i neste fase.

Fasens målsetning er:

- Å bygge et system som fungerer tilstrekkelig som en betaversjon av produktet i kundens miljø og omgivelser.

6.3.5 FASE 4 - «TRANSITION»

Dette er avslutningsfasen av prosjektet. Her skal det fokuseres på overlevering av et fungerende produkt til kunde. Feil kan rettes hvis løsningen er rask og forutsigbar, ellers skal feilene dokumenteres med full sporbarhet til kilden og kunden skal informeres.

Avsluttende systemtester blir utført, og demonstrasjoner for kunde blir utført. Deretter skal all dokumentasjon settes sammen og overleveres før en avsluttende presentasjon blir holdt. Hvis det er hensiktsmessig og kunden ønsker det, skal applikasjonen «deployeres».

Fasens målsetning er:

- Kunne levere ut et fullt fungerende system til kunden med alle funksjonaliteter som er ønsket.

6.4 PLANLAGT GJENNOMFØRING

Her vil den faktiske gjennomføringen av prosjektmodellen vises i forhold til hver fase. Informasjon her vil være ufullstendig i perioder, da det blir oppdatert ettersom tiden går og prosjektet skrider fram.

6.4.1 FASE 1 - «INCEPTION»

Vi har 5 uker som er satt av til inception fasen, som dermed er inndelt i to iterasjoner. Den første iterasjonen har vi satt til å vare i 3 uker. Denne iterasjonen er starten på selve prosjektet, det blir derfor mye fokus på å sette oss inn i oppgaven. Det vil også bli fordelt forskjellige ansvarsområder til hvert medlem og avtale ukentlige møter med intern og eksterne veiledere. Vi skal få tildelt et fast grupperom på skolen hvor vi kan sette opp utstyret vi får utlevert fra KDA. Når dette er på plass skal vi produsere den dokumentasjonen som er ønsket til første presentasjon.

Vi skal utarbeide følgende dokumenter i denne perioden:

- Prosjektmodell
- Prosjektplan
- Kravspesifikasjon
- Testspesifikasjon
- Aktivitetsliste
- Tidsplaner
- Risikoanalyse
- Use-caser

Hvis vi får tid får tid skal vi også lage en helt enkel illustrasjon av hvordan vi ser for oss hvordan systemet kan bli seende ut til slutt. Denne tenker vi å vise frem på den første presentasjon for å gi tilhørerne et litt mer grafisk bilde av oppgaven.

Den andre iterasjonen varer da i 2 uker. Her blir hovedfokuset å ferdigstille alle dokumentene slik at de er klare for innlevering til første presentasjon. Det blir selvfølgelig også noe tid som går bort til å forberede oss til selve presentasjonen.

6.4.2 FASE 2 - «ELABORATION»

Vi har tre uker som er satt av til Elaboration fasen, som dermed er inndelt i tre iterasjoner. Som tidligere nevnt skal det være fokus på analyse, design og arkitektur i denne fasen av prosjektet. I første iterasjon skal alt som er gjort i Inception fasen, spesielt kravspesifikasjonen, gås over en gang til for å konkretisere og rette inn det vi skal jobbe med videre. Deretter skal vi gjøre en del research og analyse på hvilke løsninger vi skal gå for med tanke på de viktigste delene som videostrøm og kommunikasjon. Dette er såpass viktige elementer av prosjektet, at vi vil ha en løsning før vi starter med implementasjon i Construction fasen.

Det skal også lages et utgangspunkt for design av systemet som vi kan bruke som implementasjonsgrunnlag senere. Vi skal gjøre design og analyse i form av klassediagram, komponentdiagram, sekvensdiagram, grafisk brukergrensesnitt og et eget analysedokument. Her skal valg av kommunikasjonsløsninger og protokoller avgjøres og komme fram. De løsninger vi kommer fram til vil vi formidle fortløpende til oppdragsgiver underveis i form av ukentlige møter og mail. Når de viktigste punktene for fasen blir utført vil vi gjøre test og research på de løsningene vi har valgt samt at vi skal ha satt opp grovstrukturen i prosjektet med revisjonskontroll.

6.4.3 FASE 3 - «CONSTRUCTION»

Totalt har vi 10 uker til rådighet i denne delen av prosjektet, men nærmere 2 av disse ukene vil gå med til felles påskeferie og forberedelser til eksamen. Så vi står igjen med 8 effektive arbeidsuker. Første og andre iterasjon i denne fasen vil gå med til å implementere grovstrukturen i koden, kunne sende og motta en videostrøm, kommunikasjon og forberedelser til andre presentasjon. Etter dette har vi da fem iterasjonene til implementering av resterende funksjonalitet og forbedre det som alt er implementert.

Under iterasjon 1 skal vi ha fokus på å lage grunnstrukturen til applikasjonen vår. Det vil være utvikling av GUI, samt enkel funksjonalitet til å styre kamera.

I iterasjon 2 kommer vi til å forberede oss til presentasjon nr 2. Hvis vi får tid til overs, så vil det være mulig og utbrede grunnstrukturen.
Det var litt problemer angående presentasjonen. Ekstern veileder var utenlands på jobb, mens tekniskveileder ble dessverre syk. Presentasjonen ble utsatt fra 23.03 til 31.03.

Til nå har vi arbeidet med ukes iterasjoner, dette ble diskutert og vi tok beslutningen med å øke den til 2 uker. Argumentasjonene for denne beslutningen, var at vi brukte for mye tid på å skrive planene og rapportene.

Hovedfokuset i Iterasjon 3 er å utvikle en API mot kamera og enheter. Samtidig så har vi et mål om at alle enhetene skal fungere i systemet. Hovedprioriteringen er å få på plass en skikkelig kamera API, slik at vi kan «kopiere» den til enhetene sin API. Det vil også bli utviklet grunnstruktur for kommunikasjon mellom server og klient.

Iterasjon 4 sitt fokus vil være å implementere alle enhetene i applikasjonen. Iterasjon 3 så hadde vi som mål å lage ene API til enhetene, denne utgår fordi det blir for vanskelig å legge til enheter skikkelig på senere tidspunkt. Denne avgjørelsen ble tatt sammen med teknisk veileder. Det vi skal gjøre istedenfor er å implementere enhetene på et samlet sted. Det vil også bli satt av tid til å legge til funksjonalitet til lagring og avspilling av video.

I iterasjon 5 så fokuserer vi på å sy hele systemet sammen til et komplett system. All funksjonalitet vil bli testet og alle bugs vil bli rettet opp i fortløpende. Manglende funksjonalitet som swipe og endring av mode på kamera vil bli utviklet. Kommunikasjonen vil også bli forbedret.

6.4.4 FASE 4 - «TRANSITION»

I denne fasen er det satt av 3 uker som er fordelt inn i 2 iterasjoner. I iterasjon 1 blir hovedfokuset å ferdigstille alt av dokumentasjon slik at den er klar til endelig innlevering 23.mai.

I iterasjon 2 kommer vi til å forberede oss til siste presentasjon. Hvis det viser seg å bli tid til overs er det mulig å utbedre tekniske utfordringer i denne iterasjonen.

7 OPPFØLGING OG RAPPORTERING

7.1 OPPFØLGINGSDOKUMENT

Hver uke blir et oppfølgingsdokument fylt ut, dette er for å få en oversikt over arbeidet som er gjort og hvilke arbeidsoppgaver vi skal jobbe med kommende uke. I hovedsak blir dette gjort for å se hvordan vi ligger an i henhold til prosjektplan. Dokumentet blir sendt inn til intern veileder minst et døgn før det ukentlige oppfølgingsmøtet, slik at han også kan følge med på framdriften i prosjektet. Timeliste blir vedlagt oppfølgingsdokument.

7.2 TIMELISTER

Hvert enkelt gruppemedlem fører timene sine inn i et felles Excel-dokument. Her skal det dokumenteres hvilken aktivitet som ble gjort samt antall timer som ble brukt. Dette blir igjen ført inn i et større dokument hvor det er oversikt over timeforbruk i forholdt til estimerte timer i prosjektplanen. Samtlige medlemmer er innforstått med viktigheten av å holde en slik oversikt og hvilke konsekvenser som kan oppstå hvis dette ikke opprettholdes.

7.3 MØTESTRUKTUR

7.3.1 MØTE MED INTERN VEILEDER

Møter med intern veileder, Joakim Bjørk blir holdt ukentlig hver onsdag klokken 11:00. Prosjektleder sender møteinnkalling til alle møtedeltakere med oppfølgingsdokument, ukentlig timelister og annen dokumentasjon som er relevant i henhold til møteagendaen, som vedlegg. Mailen skal sendes minimum 24 timer før møtet. Møteleder er prosjektleder og det refereres til møteagendaen. Etter hver møte blir det skrevet referat og sendes til alle møtedeltagere innen 24 timer.

7.3.2 MØTE MED EKSTERN VEILEDER

Ukentlige møter med KDA er blitt avtalt å holdes torsdag klokken 12:30. Disse møtene foregår i utgangspunktet hos HSN, med mindre annet er avtalt. Her sendes det møteinnkalling til Arnt-Levi Alstad og Rune Værnes. Møtene består av oppdatering fra arbeidsgiver og status i prosjektet. Referat blir skrevet etter hvert møte og sendt til alle møtedeltagere innen 24 timer.

7.4 NETTSIDE

Skolen har bedt oss å opprette en nettside for prosjektet. Nettsiden beskriver hva oppgaven går ut på, informasjon om gruppen og dokumentasjon. Siden oppdateres fortløpende slik at de som er interessert kan følge framdriften i prosjektet.

Nettside: https://home.hbv.no/web-gr2-2016

8 GJENNOMFØRING

I dette kapittelet her har vi beskrevet noen administrative aktiviteter. For mer informasjon, se vedlagt aktivitetsliste. I den vedlagte aktivitetslisten er noen aktiviteter merker med rød tekst, disse ble ansett som overflødige underveis i prosjektet og ble dermed satt som inactive, de ble ikke slettet da vi vil ha en god sporbarhet til alle aktiviteter.

8.1 ADMINISTRATIVE

Tabell 5 - Administrative aktiviteter

Kravspesifikasjon	En beskrivelse til kunden av hvilke behov som skal tilfredsstilles og hvilke resultater og mål som skal nås. Dokumentet vil også hjelpe oss å effektivisere prosessen mot et ferdig produkt.
Testplan	Et dokument som skal fylles ut etter hvert som et krav blir ferdig. Dette er for å teste kravet, slik at eventuelle «bugs» eller andre problemer som oppstår når man implementerer dette i et større system skal oppdages så tidlig så mulig.
Prosjektplan	Hensikten med en prosjektplan er å fordele og strukturere arbeidet gjennom prosjektet på en taktisk måte. Uten en prosjektplan vil et prosjekt ha en mye større risiko for å feile. En god prosjektplan inneholder tidsestimeringer per aktivitet, prosjektmodell, aktiviteter, budsjett, testplan, kravspesifikasjon osv. Det er et dokument for alt, og desto fler dokumenter som blir utnyttet riktig, desto høyere sjanse for å få et suksessfullt prosjekt.
Dokumentasjon	Loggføring under hele prosjektet er en vital del av et prosjekt. En god dokumentasjon gjør at vi kan spore tilbake hvem som har gjort hva, og når det ble gjort. Hvert prosjekt vil ha utfordringer og erfaringer som kan bli relevant i framtidige prosjekter. Det er derfor viktig å dokumentere slike situasjoner og avgjørelser som blir tatt.
Visjonsdokument	Her blir det estimert tidsforbruk samt ressurser og ut ifra det bestemmes milepæler underveis i prosjektet.

8.2 MILEPÆLER

Tabell 6- Milepæler

Milepæl	Hendelse	Forventet ferdig	Ferdig
Presentasjon nr. 1		12.02.16	X
Presentasjon nr. 2		18.03.16	X
Ferdigstilling		23.05.16	X
Presentasjon nr. 3		08.06.16	

8.3 TIDS –OG RESSURSPLANER

Framdriftsplanen oppdateres kontinuerlig.

8.4 KOSTNADER

Vi trenger ett nettverkskamera for prosjektet. Det finnes mange slike og vi har sett på forskjellige kameraer i prisklassen 10,000 – 22,000. Utstyret blir betalt av KDA, derfor kreves det en dialog med dem om hvilket kamera som passer for vårt prosjekt og eventuelt gjenbruk.

Tabell 7 - Budsjettering

Beskrivelse	Pris
Nettverkskamera	25000
Sum	25000

VEDLEGG

AKTIVITETSLISTE

Aktiviteter - oversikt					
			V	2.0	
A1 - Dokumentasjon og Prosjektstyring					
Aktivitet ID	**Beskrivelse**	**Krav ID**	**Ansvarlig**	**Tidsestimat i timer**	**Avhengighet**
A1-001	Prosjektplan		Svein	60	A1-002, A1-003
A1-002	Tidsplan		Alexander	30	A1-001
A1-003	Aktivitetsplan		Svein	15	A1-001, A1-005
A1-004	Use Case		Carl Peder	30	
A1-005	Kravspesifikasjon		Alexander og Vegard	60	A1-004
A1-006	Testspesifikasjon		Carl Peder og Svein	50	A1-005
A1-007	Kontrakter		Svein	5	
A1-008	Risikodokument		Alexander	10	
A1-009	Lage dokumentmaler		Alexander	5	
A1-010	Lage og oppdatere iterasjonsplan og rapport I1		Vegard	3	
A1-011	Lage og oppdatere iterasjonsplan og rapport I2		Alexander	3	
A1-012	Lage og oppdatere iterasjonsplan og rapport E1		Svein	3	
A1-013	Lage og oppdatere iterasjonsplan og rapport E2		Carl Peder	3	
A1-014	Lage og oppdatere iterasjonsplan og rapport E3		Alexander	3	
A1-015	Lage og oppdatere iterasjonsplan og rapport C1		Vegard	3	
A1-016	Lage og oppdatere iterasjonsplan og rapport C2		Svein	3	
A1-017	Lage og oppdatere iterasjonsplan og rapport C3		Carl Peder	3	
A1-018	Lage og oppdatere iterasjonsplan og rapport C2		Alexander	3	
A1-019	Lage og oppdatere iterasjonsplan og rapport C4		Vegard	3	
A1-020	Lage og oppdatere iterasjonsplan og rapport C5		Svein	3	

A1-021	Lage og oppdatere iterasjonsplan og rapport C5		Carl Peder	3	
A1-022	Lage og oppdatere iterasjonsplan og rapport C6		Alexander	3	
A1-023	Lage og oppdatere iterasjonsplan og rapport C7		Vegard	3	
A1-024	Lage og oppdatere iterasjonsplan og rapport C8		Svein	3	
A1-025	Lage og oppdatere iterasjonsplan og rapport C9		Carl Peder	3	
A1-026	Lage og oppdatere iterasjonsplan og rapport T1		Alexander	3	
A1-027	Lage og oppdatere iterasjonsplan og rapport T2		Vegard	3	
A1-028	Lage og oppdatere iterasjonsplan og rapport T3		Svein	3	
A1-029	Lage og oppdatere iterasjonsplan og rapport T4		Carl Peder	3	
A1-030	Visjonsdokument		Svein	10	
A1-031	Designdokument		Carl Peder	15	
A1-032	Implementasjon		Svein	6	
A1-033	Teknologidokument		Alexander	10	
A1-034	Analysedokument		Svein	26	
A1-035	Oppdatering av nåværende dokumenter		Alexander	150	
A1-036	Lage og oppdatere iterasjonsplan og rapport		Alle	30	
A1-037	Kommunikasjon dokument		Svein og Vegard	10	
			Sum	**522**	

A2 - Møter og presentasjoner					
Aktivitet ID	**Beskrivelse**	**Krav ID**	**Ansvarlig**	**Tidsestimat i timer**	**Avhengighet**
A2-001	Kongsberg Gruppen		Vegard	10	
A2-002	Veiledermøte internt		Vegard	30	
A2-003	Veiledermøte eksternt		Vegard	30	
A2-004	Andre presentasjoner		Alle	5	
A2-005	Presentasjon 1		Alle	60	
A2-006	Presentasjon 2		Alle	60	
A2-007	Presentasjon 3		Alle	80	
			Sum	**275**	

A3 – Research					
Aktivitet ID	**Beskrivelse**	**Krav ID**	**Ansvarlig**	**Tidsestimat i timer**	**Avhengighet**
A3-001	Software		Alle	10	
A3-002	Hardware		Alle	10	
A3-003	Prosjektstyring		Alle	50	
			Sum	**70**	
A4 – Kurs					
Aktivitet ID	**Beskrivelse**	**Krav ID**	**Ansvarlig**	**Tidsestimat i timer**	**Avhengighet**
A4-001	Qt		Vegard og Alexander	20	
A4-002	C++		Vegard og Alexander	20	
			Sum	**40**	
A5 - Administrasjon					
Aktivitet ID	**Beskrivelse**	**Krav ID**	**Ansvarlig**	**Tidsestimat i timer**	**Avhengighet**
A5-001	Henting og installering av utstyr		Alle	10	
A5-002	Oppdatere Oppfølgingsdokument		Svein	10	
A5-003	Skrive møtereferat		Svein	20	A2-001 til 003
A5-004	Forberedelse til møter og innkallinger		Vegard	10	
A5-005	Annet		Alle	10	
A5-006	Logo		Vegard	5	
A5-007	Plakat		Svein	10	
A5-008	Opprydding i kode		Svein	32	
			Sum	**65**	
A6 - Research og testing					
Aktivitet ID	**Beskrivelse**	**Krav ID**	**Ansvarlig**	**Tidsestimat i timer**	**Avhengighet**
A6-001	Annet - må dokumenteres		Svein	5	
A6-002	TCP kommunikasjon i Qt		Alexander	100	
A6-003	UDP kommunikasjon i Qt		Vegard	100	
A6-004	Videoanalyse		Vegard	50	
A6-005	VlcLib til Qt		Carl Peder	30	
A6-006	SQLlite i Qt, mySQL		Carl Peder	16	
A6-007	InterComDDS fra KDA		Alexander	16	
A6-008	Kode struktur		Alexander og Vegard	10	
A6-009	Prototype		Vegard	5	

A6-010	Testing		Carl Peder	100	
			Sum	**300**	
A7 - Design og arkitektur					
Aktivitet ID	**Beskrivelse**	**Krav ID**	**Ansvarlig**	**Tidsestimat i timer**	**Avhengighet**
A7-001	Arkitektur		Svein	20	
A7-002	Class diagram		Vegard	20	
A7-003	Design		Carl Peder	10	
A7-004	Hg Tortoise		Vegard	8	
A7-005	Hjemmeside		Svein	20	
A7-006	Sekvens diagram		Vegard	15	
A7-007	Component diagram		Svein	10	
A7-008	Optimalisering av nåværende system		Vegard	10	
			Sum	**103**	
A10 - System og server					
Aktivitet ID	**Beskrivelse**	**Krav ID**	**Ansvarlig**	**Tidsestimat i timer**	**Avhengighet**
A10-001	Motta video fra kamera på server	K01.003A	Alexander	10	
A10-002	Sende videostrøm til klienter	K01.001A	Alexander	50	
A10-003	Lagre video på server	K01.003A	Alexander	50	
A10-004	La klienter koble seg til server for videostreaming	K01.001A	Alexander	30	
A10-005	Feilsøking videoserver		Alexander	30	
A10-006	Utvikle funksjoner i server		Alexander	10	
			Sum	**140**	
A20 - Klient					
Aktivitet ID	**Beskrivelse**	**Krav ID**	**Ansvarlig**	**Tidsestimat i timer**	**Avhengighet**
A20-001	Koble seg opp mot server	K02.001A.01A	Alexander	50	
A20-002	Motta videostrøm fra server	K02.001A.02A	Alexander	50	
A20-003	Opprette kommunikasjon for datameldinger	K02.001A	Alexander	60	
			Sum	**160**	
A30 – GUI					
Aktivitet ID	**Beskrivelse**	**Krav ID**	**Ansvarlig**	**Tidsestimat i timer**	**Avhengighet**
A30-001	Lage videospiller som kan spille av videostrøm	K03.002A	Carl Peder	10	
A30-002	Videokontroller for start, stopp, pause etc.	K03.002A	Carl Peder	10	

A30-003	Se info om videodata, forsinkelse, CPU-forbruk etc.	K03.001A	Carl Peder	10	
A30-004	Utvikle GUI til systemet	K03.002A	Carl Peder	80	
A30-005	Spille av video fra server	K01.002A.03A	Carl Peder	20	
			Sum	**100**	
A40 - Eyetracker					
Aktivitet ID	**Beskrivelse**	**Krav ID**	**Ansvarlig**	**Tidsestimat i timer**	**Avhengighet**
A40-001	Lag en API for Qt Quick applikasjoner	K04.002A	Alexander	30	
A40-002	Skal kunne detektere hvor øynene ser	K04.001A	Alexander	10	
A40-003	Kalibrering for forskjellige brukere	K04.001A	Alexander	8	
A40-004	Integrering av enheten i systemet	K04.001A	Alexander	22	
			Sum	**48**	
A50 - 3D Mus Space Navigator					
Aktivitet ID	**Beskrivelse**	**Krav ID**	**Ansvarlig**	**Tidsestimat i timer**	**Avhengighet**
A50-001	Lag en API for Qt Quick applikasjoner	K05.001A	Carl Peder	50	
A50-002	Registrere multiple posisjoner for alle retninger	K05.002A	Carl Peder	30	
A50-003	Registrere om knapper blir trykket	K05.002A	Carl Peder	5	
A50-004	Integrering av enheten i systemet	K05.001A	Carl Peder	60	
			Sum	**60**	
A60 - Multi Touch Control Wheel					
Aktivitet ID	**Beskrivelse**	**Krav ID**	**Ansvarlig**	**Tidsestimat i timer**	**Avhengighet**
A60-001	Lag driver	K06.001A	Vegard	100	
A60-002	Lag en API for Qt Quick applikasjoner	K06.002A	Vegard	50	
A60-003	Registrere brukerinputs	K06.003A-06A	Vegard	50	
A60-004	Integrere i systemet		Vegard	20	
			Sum	**170**	

A70 – Kamera					
Aktivitet ID	**Beskrivelse**	**Krav ID**	**Ansvarlig**	**Tidsestimat i timer**	**Avhengighet**
A70-001	Koble opp videooverføring fra kamera	K07.007A	Svein	5	
A70-002	Koble opp kommunikasjon for kamera	K07.006A	Svein	5	
A70-003	Integrer API i Observation-applikasjon	K07.006A	Svein	10	
			Sum	**10**	
A80 - API					
Aktivitet ID	**Beskrivelse**	**Krav ID**	**Ansvarlig**	**Tidsestimat i timer**	**Avhengighet**
A80-001	Utvikle felles API til input enheter	K03.009A	Vegard	75	
A80-002	Utvikle generell kamera API	K07.006A	Alexander	75	
			Sum	**150**	
			Totalt tidsforbruk:	**2040**	**timer**

TIDLIGERE DOKUMENTHISTORIKK

Tabell 8 - Dokumenthistorikk versjon 0

Versjon	Dato	Skrevet av	Beskrivelse
0.1.0	29.01.2016	Alexander Schinnes	Første utkast
0.1.1	29.01.2016	Vegard Smedsvik	Supplert med Kap 7,8 og 9
0.1.2	30.01.2016	Alexander Schinnes	Tok ut risikovurdering og opprettet eget dokument
0.2.0	09.02.2016	Vegard Smedsvik	Oppdatert struktur
1.0	09.03.2016	Alexander Schinnes	Laget ny versjon og rettet skrivefeil. Oppdatert prosjektplanen i forhold til nye aktiviteter og tidsplan.
1.1	06.05.2016	Carl Peder Jansen	Oppdatert aktivitetslisten

REFERANSER

[1] The Unified Process Explained. Kendall Scott –utgitt år 2002
[2] PowerPoint presentasjon av oppgaven fra KDA. Fra møte hos Kongsberg 16.12.2015
[3] Tidligere bacheloroppgave Consultant Calendar Overview
Kilde - Bibliotek
[4] Tidligere bacheloroppgave Signal Communication Unit
Kilde - Bibliotek

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

BRUKERMANUAL – v 1.0

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENT 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIKK 4
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 OPPKOBLING 5
3.1 OPERATIV TILSTAND 5
3.2 OPPKOBLING MOT SERVER FOR KONTROLLERING AV KAMERA 6
3.3 STYRING AV KAMERA 7
4 STYRING MED DEVICER 8
4.1 SPACE NAVIGATOR 3D MUS 8
4.2 TOBII EYETRACKER 8
4.2 HOTKEYS 9
4.3 MULTI TOUCH CONTROL WHEEL 9
5 RECORD 10
6 SWIPE 10
7 CROSSHAIR 10
8 NIGHTVISION 11
9 INFO 11
10 DEVICE 11
11 CAMERA VENSTRE MENY 11
12 CAMERA HØYRE MENY 12
13 CONNECT 12
14 ARCHIVE 12
REFERANSER 12

1 DOKUMENT

Dette dokumentet omfatter en brukermanual for systemet Real Time Observation.

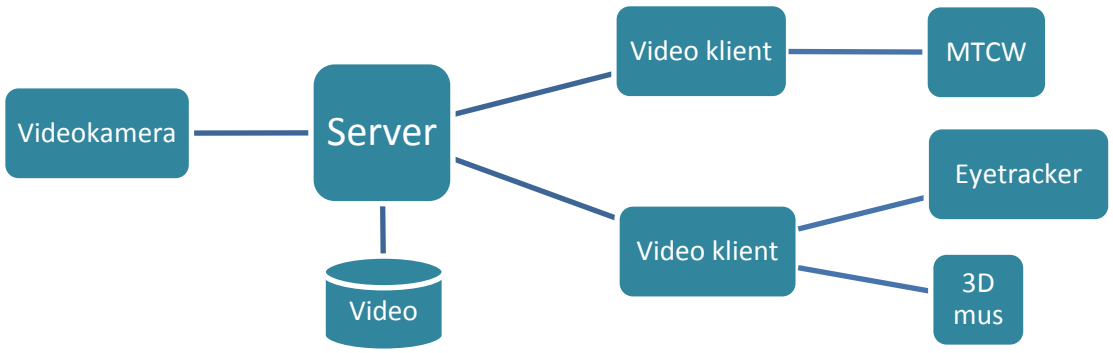
1.1 OVERSIKT

Hensikten med dette dokumentet er å gi en fullstendig brukerveiledning for brukere av systemet. Her skal operatøren få opplæring i korteste vei til operativ modus, samt forklaring for alle knapper og funksjoner man finner i Real Time Observation.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 – Systemoversikt

1.3 HISTORIKK

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
0.1	19.05	Svein	Første utkast
1.0	19.05	Vegard	Gjennomgang

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 4
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4

2.4 FIGURER OG BILDER

Figur 1 – Systemoversikt 3

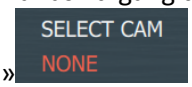
3 OPPKOBLING

3.1 OPERATIV TILSTAND

Dette avsnittet skal gi en enkel forklaring på hvordan man setter opp systemet for raskest mulig få oversikt over trusselbildet og en videofeed over hvor kameraet overvåker.

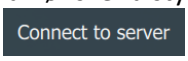
Ved å kjøre programmet vil man få opp dette bildet:

For å få opp meny trykker man på «space», eller «meny ikonet» oppe i venstre hjørne og man vil få opp menylinjer både på venstre og høyre side.

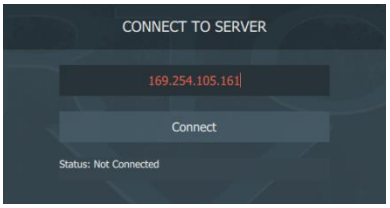
For å få rask tilgang til et kamera går man inn på menyen til venstre og trykker på «SELECT CAM» 

hvor det vil dukke opp et vindu der det velges hvilket kamera man ønsker en live video fra. Man er nå i en enkel overvåkende tilstand uten mulighet for å kontrollere kamera som innebærer zoom, tilt eller pan. For å få tilgang til dette følges videre steg 3.2.

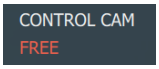
3.2 OPPKOBLING MOT SERVER FOR KONTROLLERING AV KAMERA

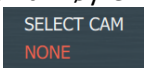
Et kriterie for å få tilgang til å styre et kamera, er at man må koble seg opp mot serveren og sjekke at kameraet man ønsker å styre er ledig. For å kunne styre et kamera klikker man på «connect to server» 

i menyen på høyre side og man får opp et vindu:

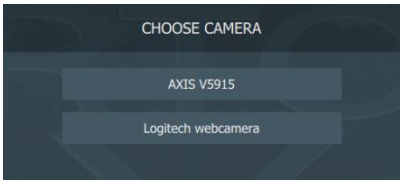


Ved å trykke på «connect» blir man tilkoblet serveren.

Så sjekker man at kameraet er ledig ved å klikke på 

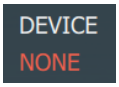
i menyen til høyre. Hvis et kamera er ledig for styring velger man type kamera i menyen til venstre: 

Fra pop-up vinduet velger man nå hvilket type kamera man ønsker å kontrollere.

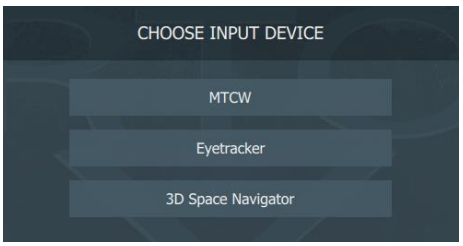


3.3 STYRING AV KAMERA

Default styring av kamera er piltastene høyre/venstre for pan og opp/ned for tilt. Kameraet vil da gå inn i en «continuous modus» hvor det fortsetter i den retningen man valgte sist helt til man avslutter det ved å trykke på nummertasten 0. Bildet vil da stå stille.

Hvis man ønsker å styre kameraet med en av devicene klikker man på 

og velger device fra pop-up vinduet.



4 STYRING MED DEVICER

Dette avsnittet vil forklare hvordan man kontrollerer kameraet ved hjelp av de forskjellige devicene.

4.1 SPACE NAVIGATOR 3D MUS

Etter at man har valgt «3D Space Navigator» fra device-menyen vil man få tilgang til 3D musen. Den styres som en enkel joystick. Hvis man ønsker å zoome kan man vri på hjulet til høyre/venstre. Knappene på siden har foreløpig ingen funksjon.

4.2 TOBII EYETRACKER

Eyetrackeren må først kalibreres for operatøren. Dette gjøres ved at man åpner Tobii eyeX applikasjonen i Windows og oppretter en ny bruker som unikt kalibreres etter operatørens øyne.

Når dette er gjort velger man «Eyetracker» fra device-menyen og kameraet styres ved at man retter øynene på de visuelle transparente rektanglene i brukergrensesnittet. Kameraet vil da gå i den retningen man ønsker i et rolig tempo.

HOTKEYS FOR EYETRACKER

Knapp	Resultat
H	Fjerner styrings rektangler
G	Viser styrings rektangler
I	Zoomer inn på punktet du ser på og sentrerer bildet
O	Zoomer ut fra punktet du ser på og sentrerer bildet
J	Sentrerer bildet uten å zoome
B	GIF animasjon vises der du ser

4.3 MULTI TOUCH CONTROL WHEEL

«MTCW» brukes på lik linje som en touch pad og velges fra device-menyen.
For å kontrollere kameraet legger man 1 finger på overflaten og kameraet vil følge fingerens mønster. Hvis man ønsker å zoome inn/ut så bruker du fingre som vist på bildet under.

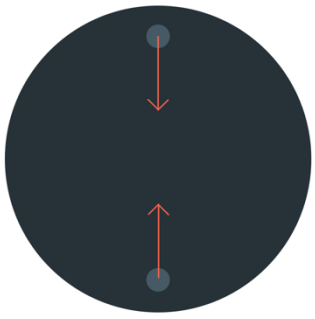


Figure 1 - Zoom ut

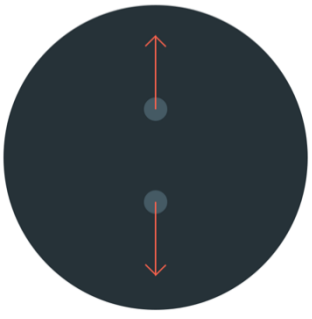
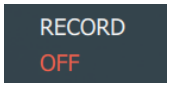


Figure 2 - Zoom inn

5 RECORD



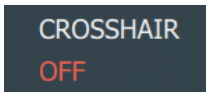
Record er en funksjon som starter lagring av video. Så lenge «RECORD» er aktiv og varsellampen lyser, vil videofeeden lagres som en fil i «ARCHIVE». Denne videofilen kan spilles av på et senere tidspunkt i videoavspilleren som åpnes fra «ARCHIVE».
Navnet på filen er gitt av datoen og tidspunktet du startet å ta opp video.

6 SWIPE

Swipe er en funksjon som lagrer et situasjonsbilde på 360°.

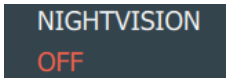
Når man klikker på «SWIPE» vil kameraet stille seg i posisjon -180 °, så aktiveres «recording» og kameraet vil foreta et sveip hvor video av området fra -180 til 180 blir lagret med tid og dato. 1 runde tar ca. 60 sekunder. Videoen lagres i et arkiv som kan hentes opp senere under «ARCHIVE».

7 CROSSHAIR



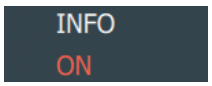
Ved å aktivere «CROSSHAIR» vil man få opp et trådkors i midten av skjermen. Dette kan flyttes på med musepekeren. Hvis man ikke ønsker å ha trådkorset tilgjengelig kan man trykke på «CROSSHAIR» igjen, og det vil forsvinne.

8 NIGHTVISION



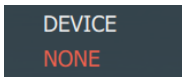
«NIGHTVISION» er en funksjon som gjør det mulig å se i mørket. Ved å aktivere denne funksjonen går man over i nightvision-modus og fargene blir svart/hvitt. For å gå tilbake til normal modus klikker man igjen på funksjonen. Denne funksjonen krever at man er koblet til et kamera med støtte for nightvision.

9 INFO



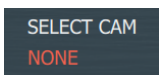
«INFO» vil gi deg informasjon oppe i høyre side av skjermen med forskjellige data som FPS, Latency og mer.

10 DEVICE



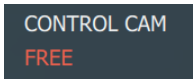
«DEVICE» gir deg mulighet for å velge hvilken device du vil kontrollere kameraet med.

11 CAMERA VENSTRE MENY



«SELECT CAM» gir deg mulighet til å velge hvilket kamera du vil ha videofeed fra.

12 CAMERA HØYRE MENY



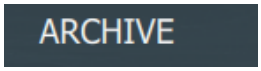
I den høyre menyen velges det om man skal kontrollere kameraet. Hvis kameraet er ledig så vil det står «FREE» oppe i venstre hjørnet av skjermen.

For å ha mulighet til å velge å styre et kamera må man være tilkoblet serveren.

13 CONNECT

For å koble til serveren klikker man på «CONNECT». Dette gir mulighet for å kontrollere et kamera.

14 ARCHIVE



«ARCHIVE» gir tilgang til tidligere lagrede videoer. Disse filene kan spilles av direkte i applikasjonens videoavspiller.

REFERANSER

Ingen

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

TEKNOLOGIDOKUMENT – v 2.0

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENT 4
1.1 OVERSIKT 4
1.2 SYSTEMOVERSIKT 4
1.3 HISTORIKK 5
2 FIGURER OG TABELLER 5
2.1 AKRONYMER OG FORKORTELSER 5
2.2 ORDBOK 5
2.3 TABELLER 6
2.4 FIGURER OG BILDER 6
3 SOFTWARE 7
3.1 UTVIKLINGSPLATTFORM 7
3.2 UTVIKLINGSMILJØ 7
3.3 REVISJONSKONTROLL 7
3.4 LISENSIERING 3. PARTS PROGRAMVARE 7
4 HARDWARE 8
4.1 AXIS V5915 VIDEOKAMERA 8
4.1.1 BESKRIVELSE 8
4.1.2 SPESIFIKASJONER 8
4.2 LOGITECH WEBCAM PRO 9000 9
4.2.1 BESKRIVELSE 9
4.2.2 SPESIFIKASJONER 9
4.3 PICOLO HD H.264 VIDEO CAPTURE BOARD 9
4.3.1 BESKRIVELSE 9
4.3.2 SPESIFIKASJONER 9
4.4 TOBII EYE X CONTROLLER 10
4.4.1 BESKRIVELSE 10
4.4.2 SPESIFIKASJONER 10
4.5 MULTI TOUCH CONTROL WHEEL 10
4.5.1 BESKRIVELSE 10
4.5.2 SPESIFIKASJONER 10
4.6 3D MOUSE SPACENAVIGATOR 11

4.6.1 BESKRIVELSE 11
4.6.2 SPESIFIKASJONER 11
5 DATABLAD 11
VEDLEGG 12
TIDLIGERE DOKUMENTHISTORIKK 12
REFERANSER 12

1 DOKUMENT

Dette dokumentet omfatter et teknologidokument for systemet Real Time Observation.

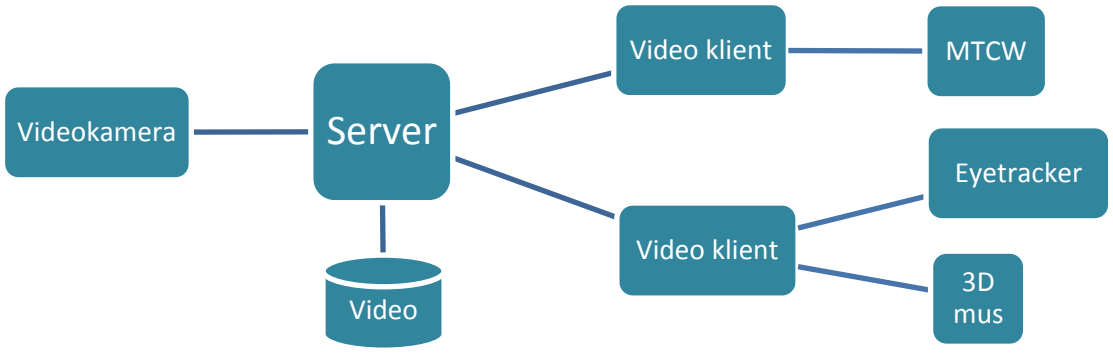
1.1 OVERSIKT

Dette dokumentet skal inneholde all informasjon om den teknologien vi har valgt å bruke, både innenfor hardware og software. Ved å lese dette dokumentet skal man sitte igjen med en oversikt over hvilke komponenter systemet inneholder, hvorfor vi har valgt akkurat disse komponentene og hvilke software verktøy og løsninger vi benytter.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 – Systemoversikt

1.3 HISTORIKK

Se vedlegg for tidligere dokumenthistorikk.

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
1.0	09.03.2016	Carl Peder Jansen	Dokumentgjennomgang før andre presentasjon
1.1	19.05.2016	Alexander Schinnes	Lagt til kap. 3.4 om lisensiering
2.0	19.05.2016	Carl Peder Jansen	Dokumentgjennomgang før innlevering

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge
MTCW	Multi Touch Control Wheel
SDK	Software Development Kit

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 5
Tabell 2 - Akronymer og forkortelser 5
Tabell 3 - Ordbok 5
Tabell 4 - AXIS V5915 videokamera, spesifikasjoner 8
Tabell 5 - Logitech Webcam Pro 9000, spesifikasjoner 9
Tabell 6 - Picolo HD H.264 spesifikasjoner 9
Tabell 7 - Tobii EyeX Controller spesifikasjoner 10
Tabell 8 - Multi Touch Control Wheel spesifikasjoner 10
Tabell 9 - 3D Mus SpaceNavigator spesifikasjoner 11
Tabell 10 - Dokumenthistorikk versjon 0 12

2.4 FIGURER OG BILDER

Figur 1 – Systemoversikt 4

3 SOFTWARE

Prosjektoppgaven er en software oppgave, derfor er det viktig å ha en god oversikt over hvilke systemer, applikasjoner og løsninger vi har valgt å bruke.

3.1 UTVIKLINGSPLATTFORM

Vi benytter Microsoft Windows 7 64bit som plattform for å utvikle all kode og funksjonalitet. Dette er en god plattform som vi kjenner godt til. Alternative valg kunne vært OS X eller Linux, men vi mener det er en fordel for oss å benytte et system vi kjenner godt fra før. Det var også dette operativsystemet som ble levert av oppdragsgiver på en HP desktop workstation PC.

3.2 UTVIKLINGSMILJØ

Det er et krav fra oppdragsgiver at Qt Quick skal benyttes for å utvikle applikasjonen. Qt Quick er et utviklingsmiljø som tillater en kombinasjon av to programmeringsspråk. Det ene er QML og det andre er C++. QML er et moderne programmeringsspråk for å kunne utvikle brukergrensesnittet i applikasjoner. Det er lett å lære, og man kan gjøre mye med forholdsvis lite kode. Qt Quick har en solid C++ kjerne i tillegg til QML for å kunne utføre all funksjonalitet som trengs for dagens software. Det finnes i tillegg flere moduler man kan benytte i både C++ og QML, som for eksempel OpenGL, filbehandling, TCP/UDP, XML og multimedia. Man har mye høynivå koding i form av QML, men også mulighet for å benytte andre språk på lavere nivå som C og C++. Kombinasjonen av disse to mulighetene gjør at Qt Quick kan brukes av både designere og «hardcore» utviklere.

3.3 REVISJONSKONTROLL

For å ha kontroll på revisjoner og versjoner av koden benytter vi TortoiseHg, som er et revisjonskontroll program. Her laster vi opp all kode som benyttes i prosjektet. Alle endringer som blir gjort underveis, store som små, blir loggført i dette programmet. Da har vi full kontroll på alt som har blitt gjort i alle deler av prosjektet til enhver tid, i tillegg til hvem som har gjort det. Det er også mulig å rulle prosjektet tilbake til en tidligere versjon hvis man oppdager feil i den nyeste utgaven. Dette gir en ekstra sikkerhet man kan støtte seg på underveis i utviklingen.

3.4 LISENSIERING 3. PARTS PROGRAMVARE

Det vi benytter av 3. parts programvare som er i bruk i systemet pr i dag er VLC og Qt. VLC har ingen restriksjoner for bruk av deres produkt. Det kan benyttes til hva som helst (innenfor landets regelverk) for personlig bruk, utdanning, research, militært, statlig eller andre profesjonelle bransjer.

Qt har to versjoner av sin programvare. Commercial og en gratis open source versjon. Vi benytter Open Source versjonen. Denne er underlagt GNU Lesser General Public License (LGPL), hvor dens regler gjelder. Ved kjøp av en commercial lisensiert versjon av Qt gjelder ikke LGPL lisensieringen med mindre man benytter seg av enten Qt WebEngine eller Qt WebKit. Disse benytter 3. parts programvare fra Qt hvor LGPL gjelder. Regler angående LGPL lisensieringen er beskrevet på Qt sine nettsider under «Frequently Asked Questions» og «Developing with the LGPL».

4 HARDWARE

Vi benytter også noe hardware i prosjektet, under følger informasjon om dette.

4.1 AXIS V5915 VIDEOKAMERA

4.1.1 BESKRIVELSE

AXIS V5915 er et nettverks kamera med innebygd motor som kan styres med jevne og fine bevegelser for pan, tilt og zoom. Det kan også ta opp høykvalitets lyd og har en kraftig 30x optisk zoom. AXIS V5915 er designet for å distribuere lyd og video til flere ulike applikasjoner. Med en enkel og åpen API, kan man lett integrere motorstyringen i andre systemer. Kameraet kan brukes til å filme og strømme konferanser, klasserom, overvåkning eller «peer-to-peer» kommunikasjon for videokonferanser.

4.1.2 SPESIFIKASJONER

Tabell 4 - AXIS V5915 videokamera, spesifikasjoner

Produktbeskrivelse	AXIS V5915 PTZ Network Camera 50Hz – nettverksovervåkningskamera
Lyd	Ja, toveismulighet
Kameramonteringstype	Veggmontering, takmontering
Dimensjoner (BxDxH)	13.6cm x 13.6cm x 18cm
Vekt	1.49kg
Farge	Svart, hvit
Konnektivitetsteknologi	Med kabler
Optisk zoom	30x
Maks. oppløsning	1920 x 1080
Bilderate	60 bilder pr. sekund
Videomoduser	1080p
Type bildebrikke	CMOS – 1/3”
Fokusjustering	Automatisk
Panoreringsområde (grader)	Fra -170 til +170
Tippeområde (grader)	-20 til +90
Grensesnitt	SDI, HDMI, Ethernet 10Base-T/100Base-TX
Spenning	DC 8 – 28V
Miljøparametere	0 C til -40 C ; 10-85 % (ikke-kondenserende) relative fuktighet
Fast IP-adresse	169.254.55.13 (satt av RTO)

4.2 LOGITECH WEBCAM PRO 9000

4.2.1 BESKRIVELSE

Logitech Webcam Pro 9000 er et høykvalitets webkamera som kan tilkobles en datamaskin via USB tilkobling. Det gir god videokvalitet, har autofokus, mikrofon og mye mer. Vi kommer til å benytte dette for demo og testing.

4.2.2 SPESIFIKASJONER

Tabell 5 - Logitech Webcam Pro 9000, spesifikasjoner

Kameratype	Farge
Maks. digitalvideooppløsning	1600 x 1200
Egenskaper	RightSound-teknologi, Skype-kompatibel, RightLight 2-teknologi, Automatic face tracking teknologi.
Grensesnitt	1 x USB 2.0 – 4 pin USB Type A
Medfølgende kabler	1 x USB-kabel 1.8m

4.3 PICOLO HD H.264 VIDEO CAPTURE BOARD

4.3.1 BESKRIVELSE

Picolo H.264 kortet er et kvalitets videokort med H.264 kompresjon som er kompatibelt med både PAL og NTSC kameraer. Det er laget for «high-end» applikasjoner innenfor videoovervåkning og tåler røffe industriforhold. Fordi kortet kan komprimere videoen selv, sparer man mye prosessorforbruk på maskinen. Den innebygde kompresjonen tilbyr høy bildekvalitet på toppen av en lav bit rate og lave lagringskrav. Hver videoinput leverer tre individuelle utgangsstrømmer. En utgang er med rå ukomprimert video, mens de to andre er med H.264 kompresjon.

4.3.2 SPESIFIKASJONER

Tabell 6 - Picolo HD H.264 spesifikasjoner

Video standarder	Farge (PAL/NTSC), sort/hvitt (CCIR/EIA)
Bildefrekvens	30fps for HD 1080p, 60fps for HD 1080i, 60fps for HD 270p
Lydkvalitet	16-bit, 32/44.1/1/48 kHz
Grensesnitt	HDMI, SDI, DVI-I og Y/Pb/Pr (komponent)
Tilkobling	PCI-Express
Vekt	115g
Størrelse (LxB)	115mm x 101mm
Utgangsstrømmer	1 x ukomprimert videostrøm + 2 x komprimert H.264 videostrøm

4.4 TOBII EYE X CONTROLLER

4.4.1 BESKRIVELSE

Dette er en MMI-enhet som kan følge hvor på skjermen en person ser ved å "tracke" øyeposisjonen. Tobii EyeX Controller er en selvstendig eyetracker som kan settes opp på monitorer, laptoper, skjermer eller andre fysiske objekter og benyttes på flere forskjellige måte. Den tillater at brukeren beveger seg under bruk samtidig som den beholder presisjonen. Det er mulig å programmere hvordan eyetrackeren skal fungere, slik at den kan benyttes til flere forskjellige formål.

4.4.2 SPESIFIKASJONER

Tabell 7 - Tobii EyeX Controller spesifikasjoner

Brukeravstand	18-40'' / 45 – 100cm
Headbox size	16 x 12'' / 40 x 30cm på 26'' / 65cm
Frekvens	60Hz
Øye til applikasjon forsinkelse	15 ms +/- 5 ms
Skjermstørrelse	Opp til 27''
Vekt	0.2 lb / 91 gram
Størrelse (BxHxL)	20 x 15 x 230mm

4.5 MULTI TOUCH CONTROL WHEEL

4.5.1 BESKRIVELSE

Multi Touch Control Wheel er en MMI-enhet som produseres av Greyhill. Det er en touch flate som kan detektere inntil fem fingre samtidig. Den har en nedfelt ring som kan benyttes som enkoder input i tillegg til fire touch piltaster som kan fungere som en joystick. En nærhetssensor kan også automatisk detektere når en bruker/finger nærmer seg enheten. Den kan brukes for å spare strøm ved at den ikke er aktiv så lenge nærhetssensoren ikke registrer noe. Enheten kan ved hjelp av touch brukes til å styre en mus, bevege objekter, zoome og rotere i tillegg til andre bevegelser.

4.5.2 SPESIFIKASJONER

Tabell 8 - Multi Touch Control Wheel spesifikasjoner

Opererende temperatur	-10C til +60C
Tetthet	IP67
Fuktighet	95% luftfuktighet på 60C i 240 timer
USB	2.0, 3.0 kompatibel, støtter full HID hastighet
Spenning	2.5 til 5.5 VDC
Strøm	50mA max ved 5V
Touchpad oppløsning	1280 x 1280 piksler

4.6 3D MOUSE SPACENAVIGATOR

4.6.1 BESKRIVELSE

Dette er en 3D mus som er laget av 3DConnexion. Den har 6 frihetsgrader med sensorer som er spesielt designet for å manipulere digitalt innhold eller kameraposisjon. Man kan dytte, løfte, pane, tilte, zoome og rotere ved hjelp av musen. Den kan beveges i alle retninger, samt en kombinasjon av flere retninger. Det følger med en SDK som gjør at softwareutviklere kan integrere den i alle applikasjoner.

4.6.2 SPESIFIKASJONER

Tabell 9 - 3D Mus SpaceNavigator spesifikasjoner

Antall frihetsgrader	6
Størrelse (LxBxH)	78mm x 78mm x 53mm
Vekt	479g
Operativsystem	Microsoft Windows, OS X, Linux
Sideveis bevegelse	55mm
Roterende bevegelse	78mm

5 DATABLAD

Se vedlagte datablad på CD for all hardware.

AXIS V5915 videokamera
[1] - Axis V5915 PTZ Network Camera
[2] - Axis V5915 User Manual

Logitech Webcam Pro 900
[3] – Logitech User Manual

Picolo HD H.264 Video Capture Board
[4] - Picolo HD Installation Guide
[5] - Picolo HD Handbook

Multi Touch Control Wheel
[6] MTCW datasheet

3D Mouse SpaceNavigator
[7] SpaceNavigator datasheet

VEDLEGG

TIDLIGERE DOKUMENTHISTORIKK

Tabell 10 - Dokumenthistorikk versjon 0

Versjon	Dato	Skrevet av	Beskrivelse
0.1	24.02.2016	Alexander Schinnes	Første utkast
1.0	09.03.2016	Carl Peder Jansen	Dokumentgjennomgang før andre presentasjon

REFERANSER

[1] Qt Quick webside: www.qt.io/qt-quick (24.02.2016)
[2] TortoiseHG webside: tortoisehg.bitbucket.org (24.02.2016)
[3] 3D mus webside: www.3dconnexion.eu/products/spacemouse/spacenavigator.html (26.02.2016)

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

DESIGNDOKUMENT – v 2.0

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENT 4
1.1 OVERSIKT 4
1.2 SYSTEMOVERSIKT 4
1.3 HISTORIKK Se vedlegg for tidligere dokumenthistorikk. 4
2 FIGURER OG TABELLER 5
2.1 AKRONYMER OG FORKORTELSER 5
2.2 ORDBOK 5
2.3 TABELLER 5
2.4 FIGURER OG BILDER 5
3 SYNOPSIS 6
4 OPPDELING 6
4.1 MODULER 6
4.1.1 KLIENTMODUL 6
4.1.2 SERVERMODUL 6
5 KLASSEDIAGRAM 7
6 SEKVENSDIAGRAM 7
6.1 VIDEOSTRØM 7
6.2 STYRING AV KAMERA VED HJELP AV MTCW 7
6.3 Lagring av video 7
7 GUI 13
8 COMPONENT DIAGRAM 14
9 USE CASE 15
9.1 SE LIVE VIDEO-STRØM 16
9.2 MANUELT OPPTAK AV VIDEO-STRØM 17
9.3 AUTOMATISK OPPTAK AV VIDEO-STRØM 18
9.4 AVSPILLING AV VIDEO FRA ARKIV 19
9.5 BYTTE VIDEOMODUS 20
9.6 MANUELL STYRING AV KAMERAPOSISJON 21
VEDLEGG 22
FORSLAG TIL GUI 1 – 25.02.2016 22
USE CASE DIAGRAMMER 23

TIDLIGERE DOKUMENTHISTORIKK..... 26
REFERANSER..... 26

1 DOKUMENT

Dette dokumentet omfatter en designdokumentasjon for systemet Real Time Observation.

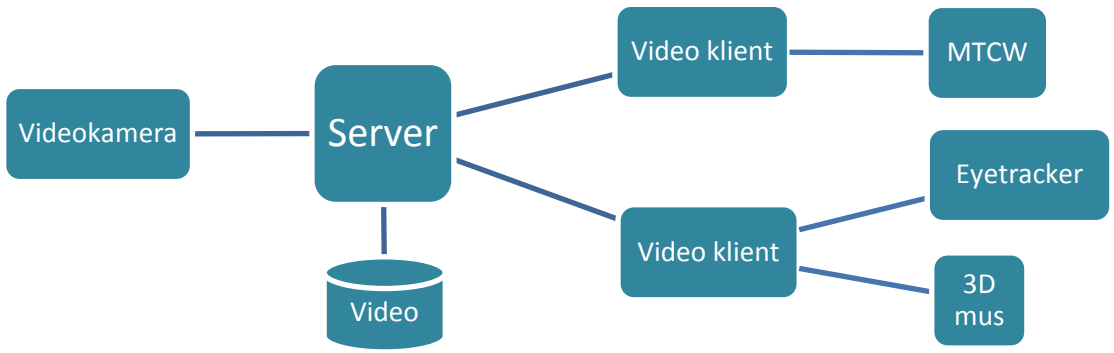
1.1 OVERSIKT

Dette dokumentet inneholder beskrivelse av design og arkitekturen for systemet.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 – Systemoversikt

1.3 HISTORIKK

Se vedlegg for tidligere dokumenthistorikk.

Tabell 1 - Dokumenthistorikk

Versjon	Dato	Skrevet av	Beskrivelse
2.0	20.05.2016	Vegard Smedsvik	Gjennomgang av dokumentet, oppdatert

Se dokumenthistorikk for tidligere versjoner vedlagt.

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorikk 4
Tabell 2 - Akronymer og forkortelser 5
Tabell 3 - Ordbok 5
Tabell 4 - Tidligere dokumenthistorikk 26

2.4 FIGURER OG BILDER

Figur 1 – Systemoversikt 4
Figur 2 - Klassediagram for systemet 8
Figur 3 - Sekvensdiagram for videostrøm 10
Figur 4 - Styring av kamera 11
Figur 5 – Lagring av video 12
Figur 6 - Use case top level 15
Figur 7 - Første utkast til GUI 22
Figur 8 - Andre utkast til GUI hovedskjerm 13
Figur 9 - Andre utkast til GUI connect 13
Figur 10 - Use case live stream 23
Figur 11 - Use case manuell opptak 23
Figur 12 - Use case automatisk opptak 24
Figur 13 - Use case avspilling av video 24
Figur 14 - Use case bytte videomodus 25
Figur 15 - Use case manuell styring 25

3 SYNOPSIS

Arkitekturen og designet har bakgrunn fra use casene og kravene vi har laget til nå. Det er kravene og use casene med høyest prioritet (A), som definerer designet og arkitekturen.

Hovedkravet vårt er å få mindre enn 200 ms forsinkelse fra nettverkskameraet til klienten. Dette krever en del analyse. Vi skal lage et fullverdig system ved hjelp av «open-source» biblioteker og software. Når systemet er oppe skal vi finne ut hvor mye forsinkelse de forskjellige delene skaper. Det vil også være fokus på å implementere forskjellige MMI-devicer til å styre nettverkskameraet.

4 OPPDELING

Systemet som er utviklet består av 2 moduler som skal kommunisere med hverandre via nettverket.

4.1 MODULER

4.1.1 KLIENTMODUL

Klientmodulen er en modul som skal kunne kobles til en servermodul. Det vil bli opprettet tre slike moduler hvor samtlige skal kommunisere med servermodulen. Klienten skal også kunne ta imot videostrøm som ligger på nettverket. Hver klient vil ha en MMI-Device koblet til, hvor hovedhensikten til gitt MMI-Device er å styre kameraet.

4.1.2 SERVERMODUL

Servermodulen er en modul som består av et Picolo HD videokort som skal ta imot en videostrøm fra et nettverkskamera, AXIS V5915 PTZ. Serveren sin oppgave er å ta imot video fra kameraet, prosessere den og sende den ut på nettverket.

Serveren skal holde oversikt over hvor mange klienter som er koblet til og hvilken klient som styrer kameraet. Når en klient har valgt å styre kameraet, så blir den samme funksjonen låst hos de andre klientene.

5 KLASSEDIAGRAM

Oppgaven vår går ut på å lage to systemer som skal kunne kommunisere med hverandre. Klasse diagrammet under viser hvilke klasser som skal ha nettverks kommunikasjon. Vi har valgt å ha TCP kommunikasjon for å sende system informasjon og UDP for å sende videostrøm. TCP har en ekstra sikkerhet når det gjelder å motta og sende informasjon, hvor hver pakke som blir sendt får respons av mottaker om at de har mottatt pakken. Denne sikkerheten skaper mer forsinkelse. UDP har ikke denne sikkerheten og er derfor mer egnet for å sende videostrøm (i vårt tilfelle).

Diagrammet under viser også klassenes hierarki.

Se figur 2 – Klassediagram for systemet

6 SEKVENSDIAGRAM

6.1 VIDEOSTRØM

Sekvensdiagrammet under viser hvordan videostrømmen vil oppføre seg fra en bruker starter systemet, til man ser video i GUI på klienten.
Det første som skjer, er at brukeren starter serveren. Vi tar utgangspunkt i at klienten er koblet på. Deretter sjekker serveren hvilket kamera som er koblet på og skaffer rett videostrøm. Serveren sjekker så om den har UDP kommunikasjon med klienten og sender videostrømmen over. Videostrømmen blir så vist i GUI på klienten.

Se figur 3 – Sekvensdiagram for videostrøm

6.2 STYRING AV KAMERA VED HJELP AV MTCW

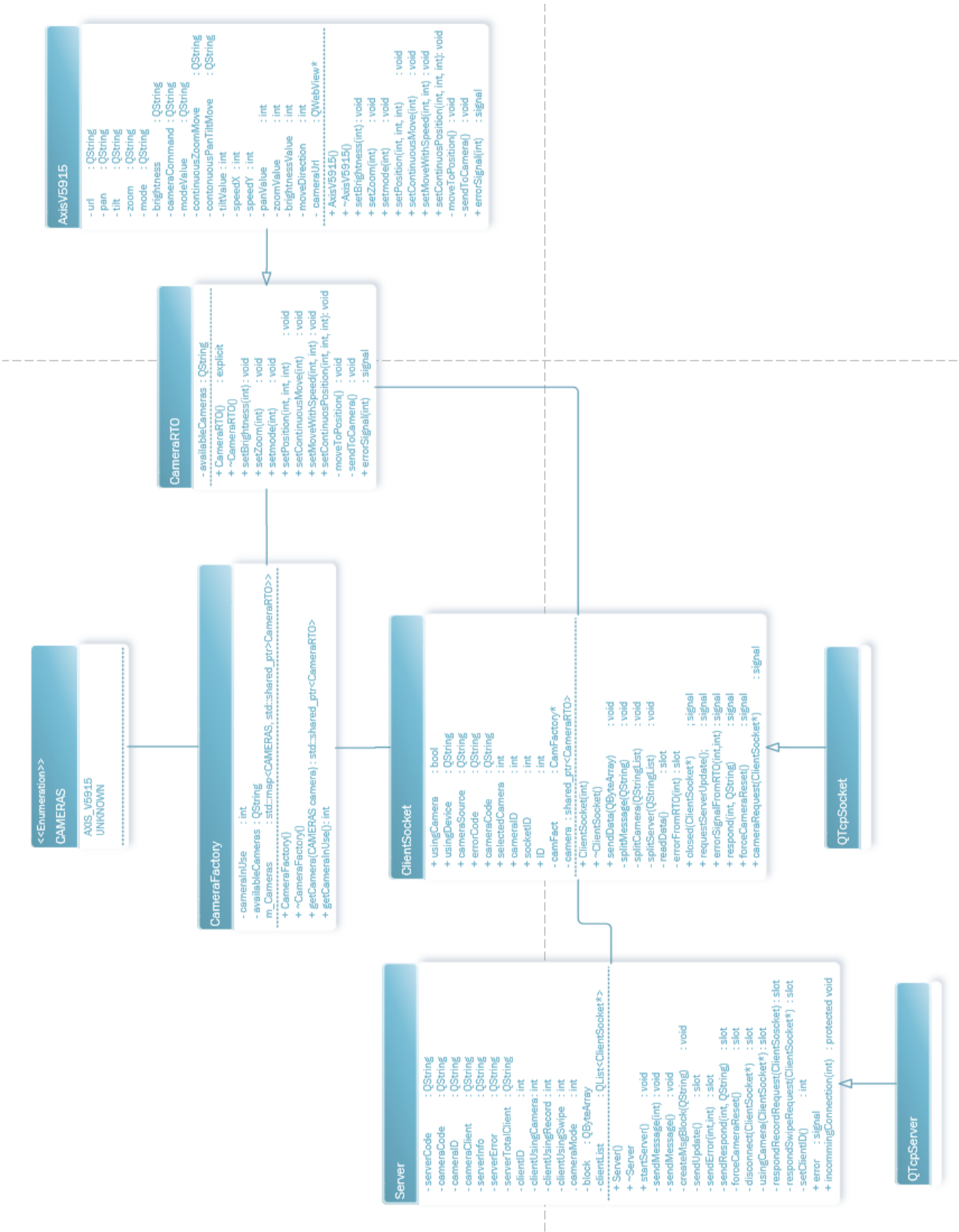
Dette er et sekvensdiagram som viser kommunikasjonen mellom Multi Touch Control Wheel og nettverkskameraet.

Se figur 4 – Styring av kamera

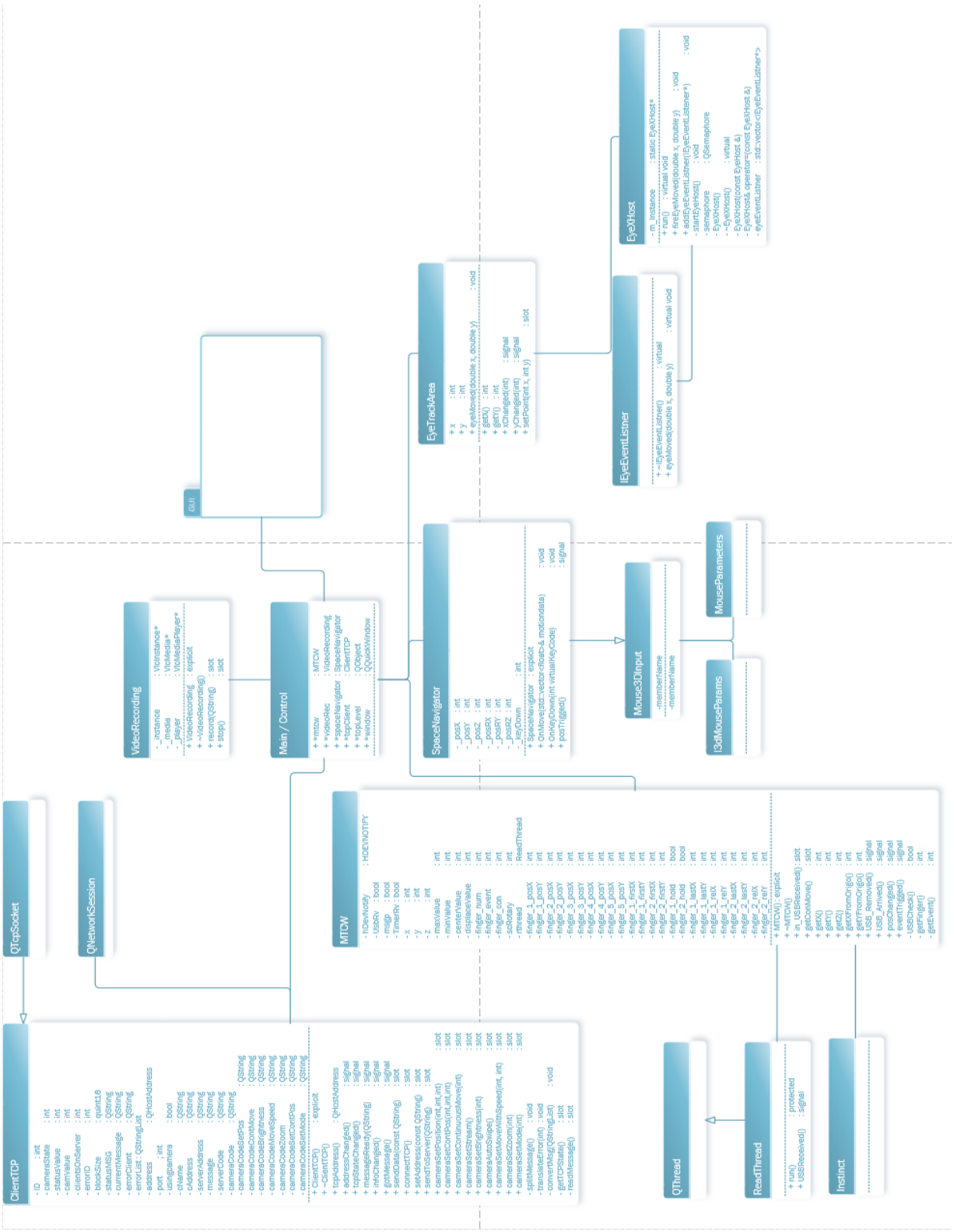
6.3 Lagring av video

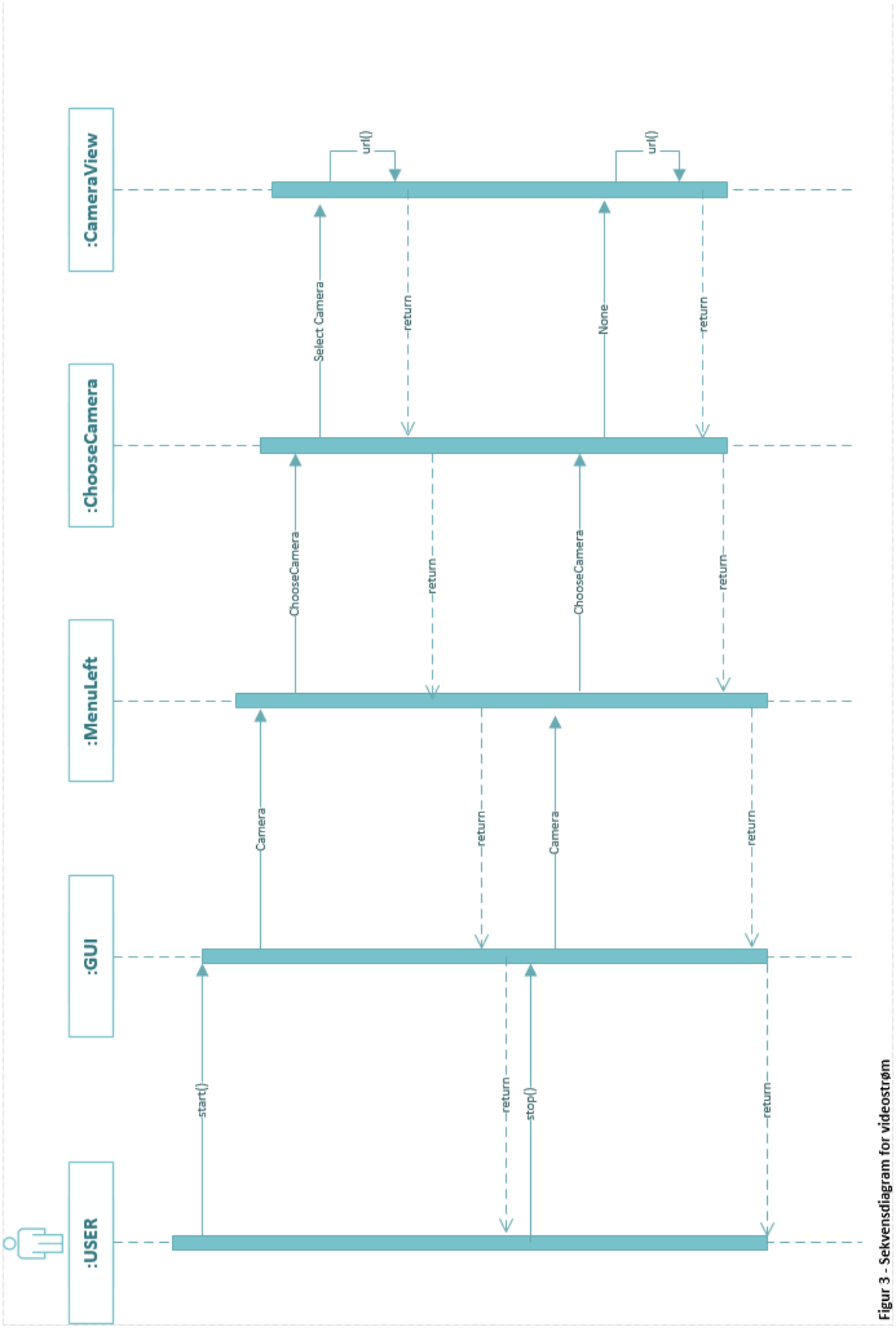
Dette er et sekvensdiagram som viser hvordan man starter og stopper opptak av video.

Se figur 5 – Lagring av video

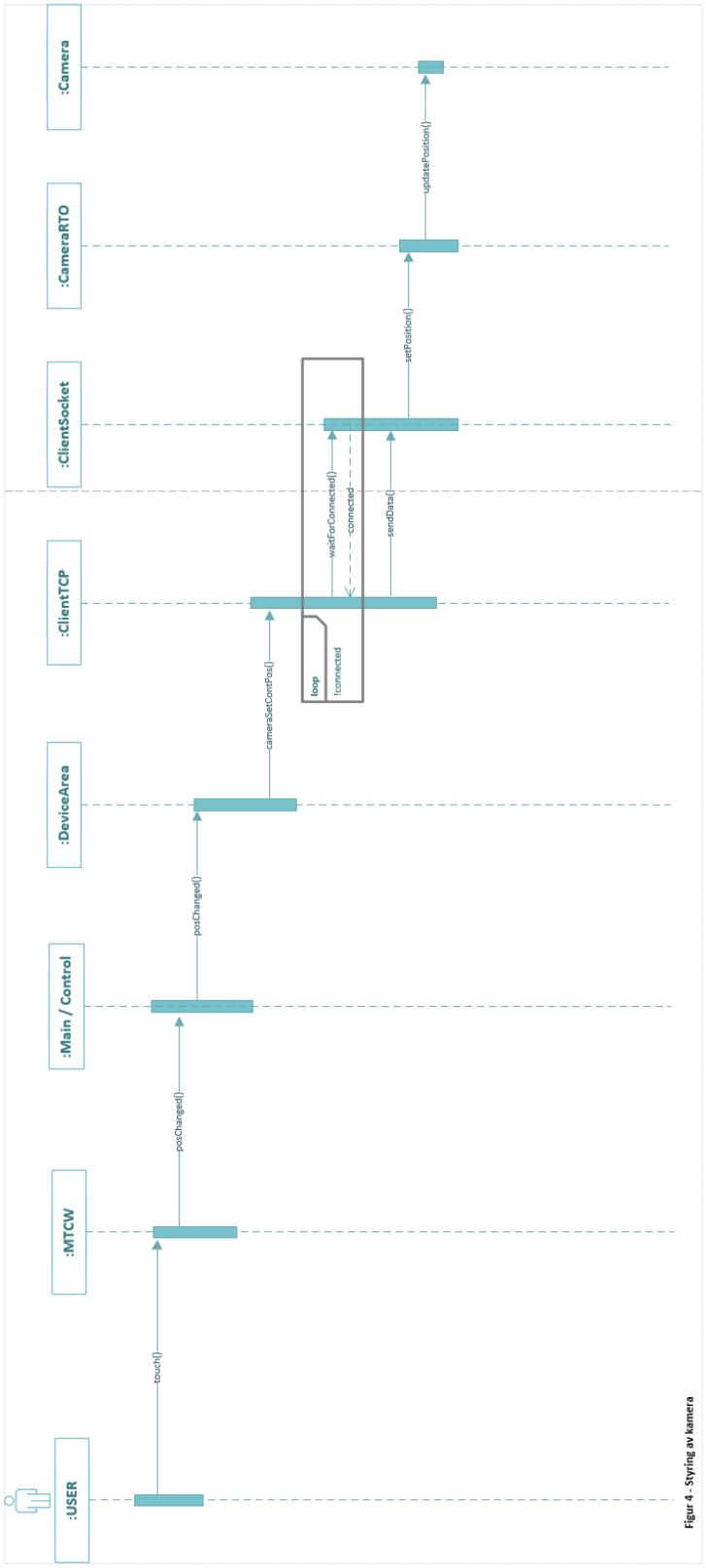


Figur 2 - Klassediagram for systemet

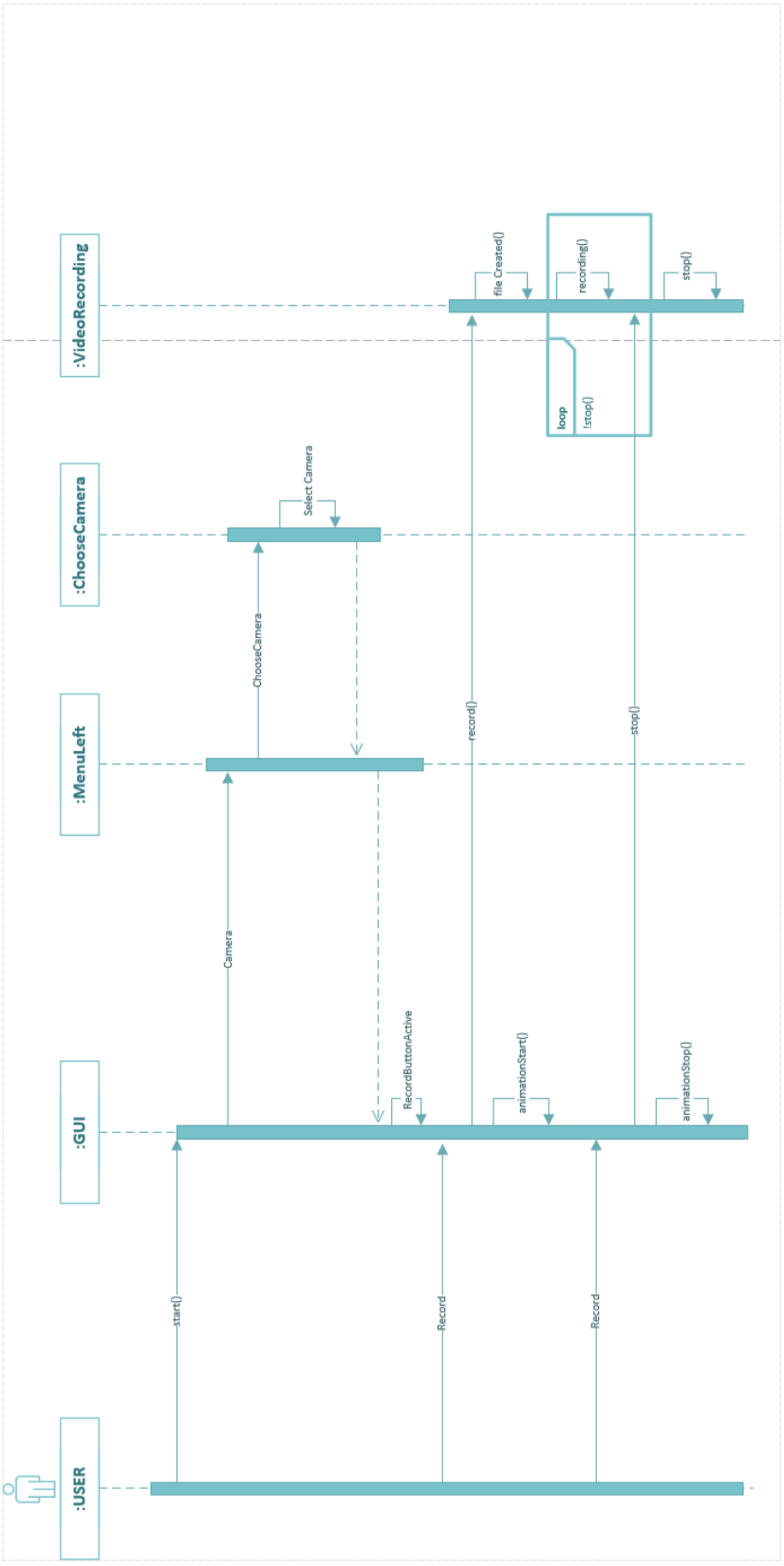

QTcpSocket
QNetworkSession
ClientTCP
VideoRecording
Main / Control
GUI
MTCW
SpaceNavigator
EyeTrackArea
IEyeEventListner
EyeXHost
Mouse3DInput
MouseParameters
I3dMouseParams
QThread
ReadThread
Instinct



Figur 3 - Sekvensdiagram for videostrøm



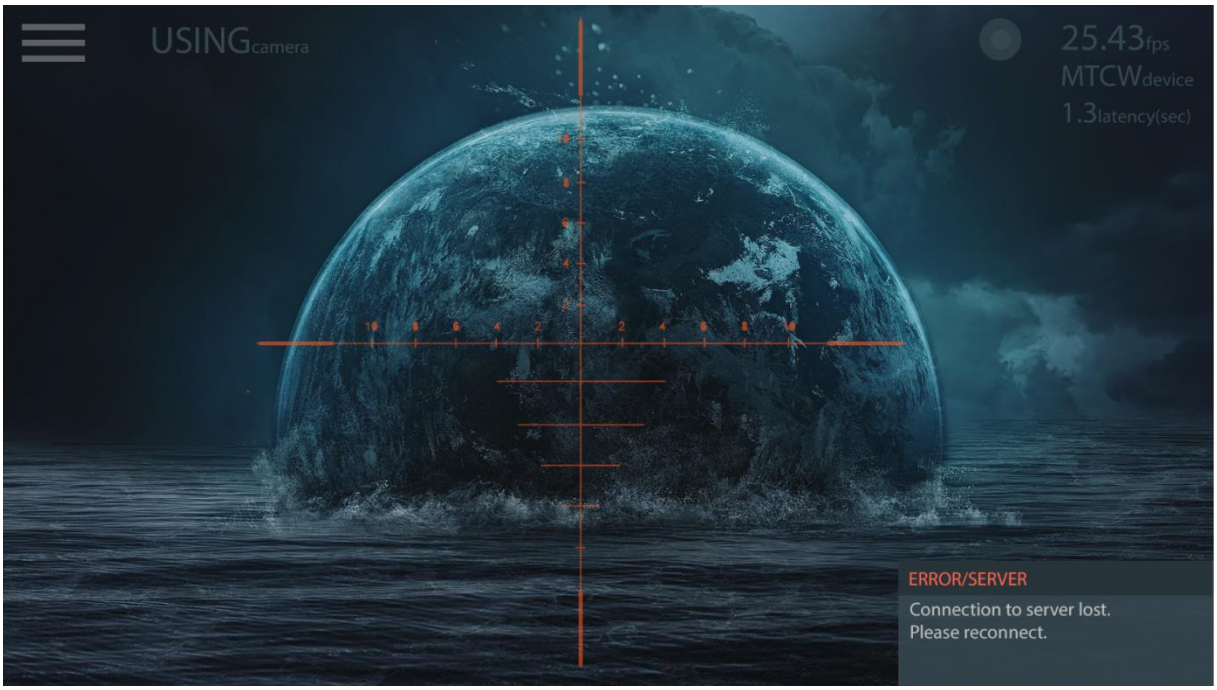
Figur 4 - Styring av kamera



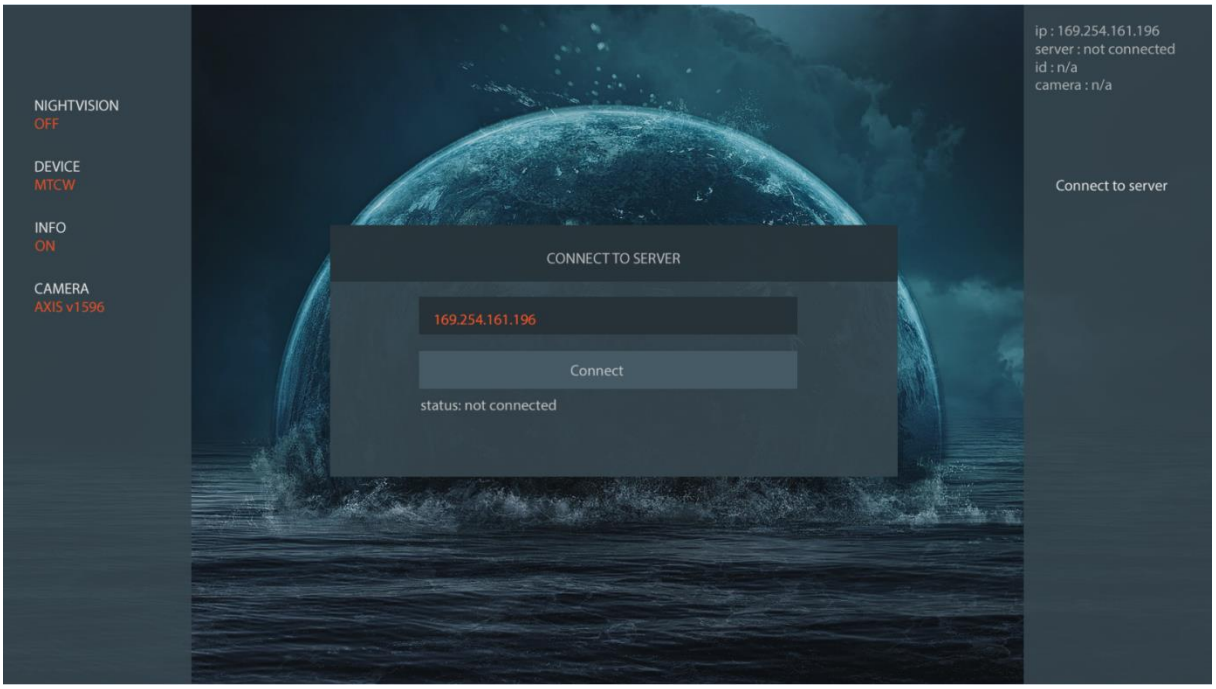
Figur 5 – Lagring av video

7 GUI

Første utkastet av GUI, se vedlegg, er blitt videreutviklet. All funksjonalitet fungerer på samme måte, bortsett fra hvilken informasjon man ser oppe i høyre hjørne. Enten så viser vi all tilgjengelig informasjon, eller ingen. All funksjonalitet er lagt til venstre meny, mens høre meny er beregnet for å komme inn i arkiv, koble seg til server og spørre om å ta i bruk kamera. Om man velger et kamera, så vil videostrømmen erstatte bakgrunnsbildet.



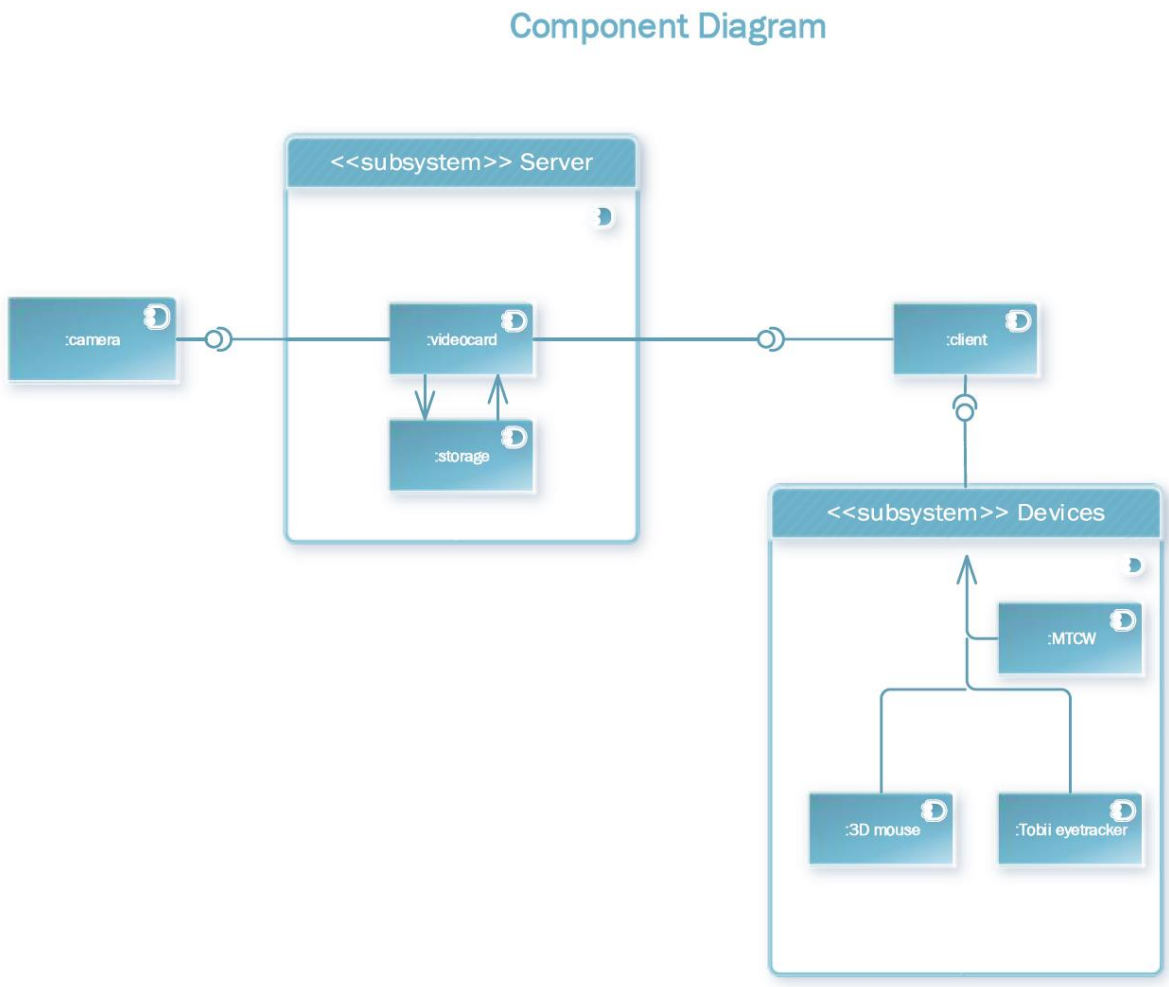
Figur 6 - Andre utkast til GUI hovedskjerm



Figur 7 - Andre utkast til GUI connect

8 COMPONENT DIAGRAM

Vi har laget komponent diagram. Dette diagrammet viser hvordan komponentene i systemet er koblet sammen.



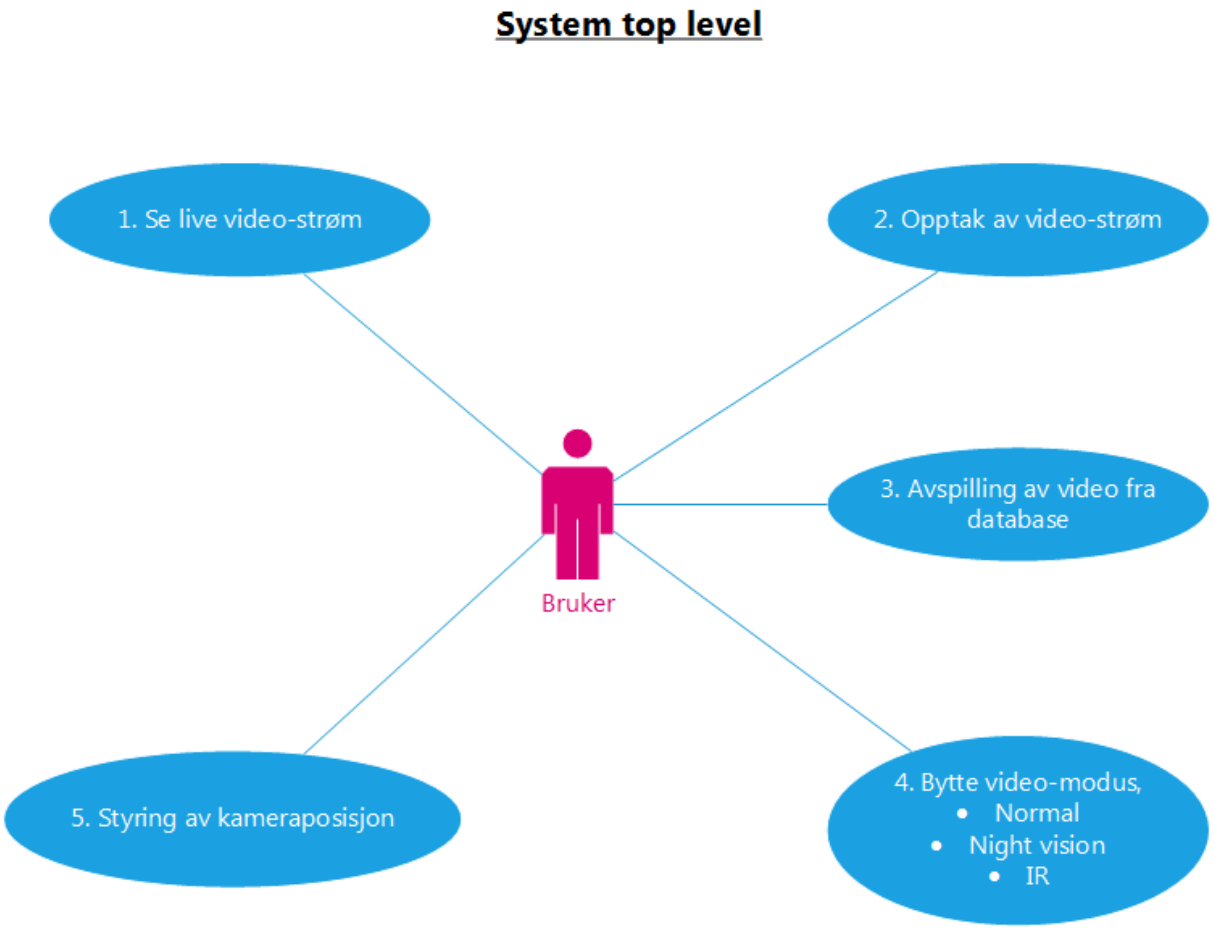
Figur 8 - Komponent diagram

9 USE CASE

Use case kategorier:

- **UC01** - Live video-strøm
- **UC02** - Opptak av video
- **UC03** - Avspilling av video
- **UC04** - Bytte av videomodus
- **UC05** - Styring av kameraposisjon

Vi har laget en toppnivå use case som beskriver hele systemet som vist under. I tillegg har vi laget en under-use case for disse fem hovedområdene. Disse er lagt med som vedlegg. Beskrivelser for hver use case følger på de neste sidene.



Figur 9 - Use case top level

9.1 SE LIVE VIDEO-STRØM

Use Case Beskrivelse	
Navn	Se live video-strøm
Use Case ID	UC01.001
Krav ID	K01.001A, K01.002C, K01.003A, K02.001A, K03.001A, K03.005A, K03.009A, K03.006C, K03.007C, K04.001A, K05.002A, K06.001A, K06.007A, K07.001A, K07.002A, K07.007A
Beskrivelse	Se live video-strøm inkluderer: • Bruker skal kunne slå av og på video-strøm. • Bruker skal kunne se video-strøm fra kameraet på sin skjerm. • Bruker skal få teknisk informasjon samtidig som video vises gjennom overlays.
Forutsetninger	• Kamera er aktivt • Videoserver er aktiv
Antagelser	• Bruker/operatør har opplæring i bruk av systemet
Hoved sekvens	1. Bruker aktiverer «Live» knapp ved bruk av input enhetene. 2. Sjekker betingelser for bruk av kamera. 3. Hvis betingelser ikke er OK, gå til alternativ sekvens A. 4. Kamera aktiv OK. 5. Videoserver mottar signal fra kamera 6. Videoserver sender videosignalet videre til brukers skjerm. 7. Brukers skjerm viser video-strøm. 8. Use case slutt.
Alternativ sekvens A	Betingelser for bruk av kamera er ikke oppfylt: 1. Feilmelding vises på skjerm 2. Lukk feilmelding
Aktører/Brukere	Alle brukere

9.2 MANUELT OPPTAK AV VIDEO-STRØM

Use Case Beskrivelse	
Navn	Manuelt opptak av video-strøm
Use Case ID	UC02.001
Krav ID	K01.001A, K01.002C, K01.003A, K02.001A, K03.001A, K03.009A, K04.001A, K05.002A, K06.001A, K06.007A, K07.001A, K07.002A, K07.007A
Beskrivelse	Opptak av video-strøm inkluderer: • Bruker skal ha mulighet til å ta opp video. • Bruker skal ha mulighet til å starte og stoppe videoopptak. • Video skal lagres. • Bruker skal kunne opprette tags (merkelapper) i video-strøm. • Bruker skal kunne ta skjermdumper av video-strøm. • Lagre tags og skjermdumper som blir opprettet av bruker.
Forutsetninger	• Kamera er aktivt • Videoserver er aktiv
Antagelser	• Bruker/operatør har opplæring i bruk av systemet
Hoved sekvens	1. Bruker aktiverer «Record» knapp ved bruk av input enhetene. 2. Sjekker betingelser for bruk av kamera. 3. Hvis betingelser ikke er OK, gå til alternativ sekvens A. 4. Kamera aktiv OK. 5. Starter opptak av video 6. Videoopptaket blir lagret i mappe med dato, og filnavnet med et logisk navn, f.eks. dato-tid. 7. Bruker aktiverer «Stop Record» knapp ved bruk av input enhetene. 8. Stopper opptak av video. 9. Use case slutt.
Alternativ sekvens A	Betingelser for bruk av kamera er ikke oppfylt: 1. Feilmelding vises på skjerm 2. Lukk feilmelding
Aktører/Brukere	Alle brukere

9.3 AUTOMATISK OPPTAK AV VIDEO-STRØM

Use Case Beskrivelse	
Navn	Automatisk opptak av video-strøm
Use Case ID	UC02.002
Krav ID	K01.001A, K01.002C, K01.003A, K02.001A, K03.001A, K03.009A, K04.001A, K05.002A, K06.001A, K06.007A, K07.005A, K07.006A, K07.007A,
Beskrivelse	Automatisk styring av kameraposisjon inkluderer: • Automatisk kjøring av kameraposisjon som kjører frem og tilbake. • Bruker skal ha muligheten til og manuelt starte og avslutte automatisk modus.
Forutsetninger	• Kamera er aktivt • Videoserver er aktiv
Antagelser	• Kamera er klar til bruk. • Bruker/operatør har opplæring i bruk av systemet
Hoved sekvens	1. Bruker aktiverer «Swipe» knapp ved bruk av input enhetene. 2. Sjekker betingelser for bruk av kamera. 3. Hvis betingelser ikke er OK, gå til alternativ sekvens A. 4. Kamera aktiv OK. 5. Kjører funksjon «Swipe», denne starter opptak av video for deretter å kjøre kameraet 340 grader rundt, når runden er gjort avsluttes videoopptaket og kameraet blir kjørt tilbake til utgangsposisjon. 6. Går tilbake til live video modus. 7. Use case slutt.
Alternativ sekvens A	Betingelser for bruk av kamera er ikke oppfylt: 1. Feilmelding vises på skjerm 2. Lukk feilmelding
Aktører/Brukere	Alle brukere

9.4 AVSPILLING AV VIDEO FRA ARKIV

Use Case Beskrivelse	
Navn	Avspilling av video fra arkiv
Use Case ID	UC03.001
Krav ID	K01.002C, K01.003A, K01.005A, K02.001A, K03.001A, K03.002B, K03.003B, K03.004C, K03.005A, K03.006C, K03.007C, K03.009A, K04.001A, K05.002A, K06.001A, K06.007A
Beskrivelse	Avspilling av video fra arkiv inkluderer: • Bruker skal kunne hente opp og spille av videoopptak fra et videoarkiv. • Bruker skal ha mulighet for enkle avspillingsvalg slik som start, stop, pause, hoppe og spole. • Bruker skal ha mulighet for å analysere video ved hjelp av funksjoner som tags og skjermdumper.
Forutsetninger	• Videoserver er aktiv • Bruker har autorisert tilgang til systemet.
Antagelser	• Bruker har opplæring i bruk av systemet
Hoved sekvens	1. Bruker aktiverer «Archives» knapp. 2. Arkiv meny som består av mapper som er navngitt etter dato kommer opp. 3. Bruker velger mappe 4. Bruker velger video fil 5. Videofil sendes fra videoserver til bruker. 6. Avspillingsvindu for video åpnes 7. Bruker aktiverer «Play» knapp eller andre avspillings alternativer (spole, pause og hoppe). 8. Video blir spilt av hos bruker. 9. Use case slutt
Aktører/Brukere	Alle brukere

9.5 BYTTE VIDEOMODUS

Use Case Beskrivelse	
Navn	Bytte videomodus
Use Case ID	UC04.001
Krav ID	K02.001A, K03.001A, K03.009A, K04.001A, K05.002A, K06.001A, K06.007A, K07.001A, K07.007A
Beskrivelse	Bytte videomodus inkluderer: • Bruker skal kunne bytte mellom 3 forskjellige video modes: 1. Normal 2. Night vision 3. Infrared(IR)
Forutsetninger	• Kamera er aktivt • Videoserver er aktiv
Antagelser	• Bruker har opplæring i bruk av systemet
Hoved sekvens	1. For at disse alternativene skal være tilgjengelig må betingelsen «Kamera aktivt» være oppfylt. Når dette er oppfylt vil tre knapper være tilgjengelig for bruker: A. N (Normal) B. NV (Night vision) C. IR (Infrared) 2. Hvis betingelser ikke er OK, gå til alternativ sekvens A. 3. Betingelser OK. 4. Bruker aktiverer en av de tre knappene. 5. Sjekker nåværende videomodus. 6. Hvis nåværende modus er den samme som valgte modus, gå til alternativ sekvens B. 7. Hvis nåværende modus er IKKE den samme som valgt modus. 8. Kameraet skifter til f.eks. «Normal» modus. 9. Use case slutt
Alternativ sekvens A	Betingelser for bruk av kamera er ikke oppfylt: 1. Feilmelding vises på skjerm 2. Lukk feilmelding
Alternativ sekvens B	Nåværende modus ER den samme valgt modus: 1. Feilmelding «Mode already selected» 2. Bruker fjerner feilmelding ved hjelp av input devices.
Aktører/Brukere	Alle brukere

9.6 MANUELL STYRING AV KAMERAPOSISJON

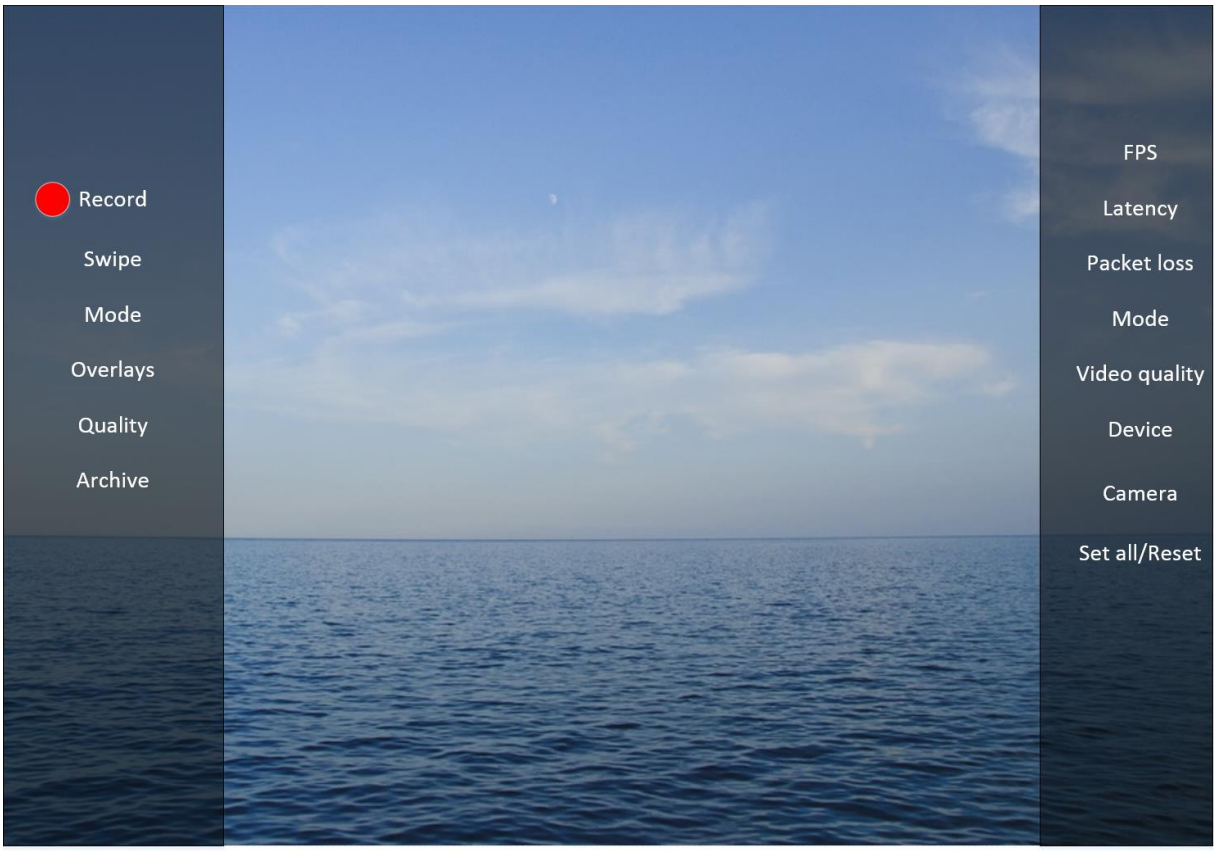
Use Case Beskrivelse	
Navn	Manuell styring av kameraposisjon
Use Case ID	UC05.001
Krav ID	K02.001A, K03.001A, K03.009A, K04.001A, K05.002A, K06.001A, K06.007A, K07.005A, K07.006A, K07.007A
Beskrivelse	Manuell styring av kameraposisjon inkluderer: • Bruker skal kunne styre kameraets posisjon ved hjelp av input enheter • Styring av kamerafunksjoner som optisk zoom og fokus.
Forutsetninger	• Kamera er aktivt • Videoserver er aktiv
Antagelser	• Bruker har opplæring i bruk av systemet
Hoved sekvens	1. Sjekker betingelser for bruk av kamera. 2. Hvis betingelser ikke er OK, gå til alternativ sekvens A. 3. Kamera aktiv OK. 4. Bruker styrer kameraet x og y akse samt funksjoner som zoom og fokus ved bruk a input enhetene. 5. Use case slutt.
Alternativ sekvens A	Betingelser for bruk av kamera er ikke oppfylt: 3. Feilmelding vises på skjerm 4. Lukk feilmelding
Aktører/Brukere	Alle brukere

VEDLEGG

Under utviklingen av systemet har vi gjort oss noen tanker om hvordan GUI til klienten skal se ut.

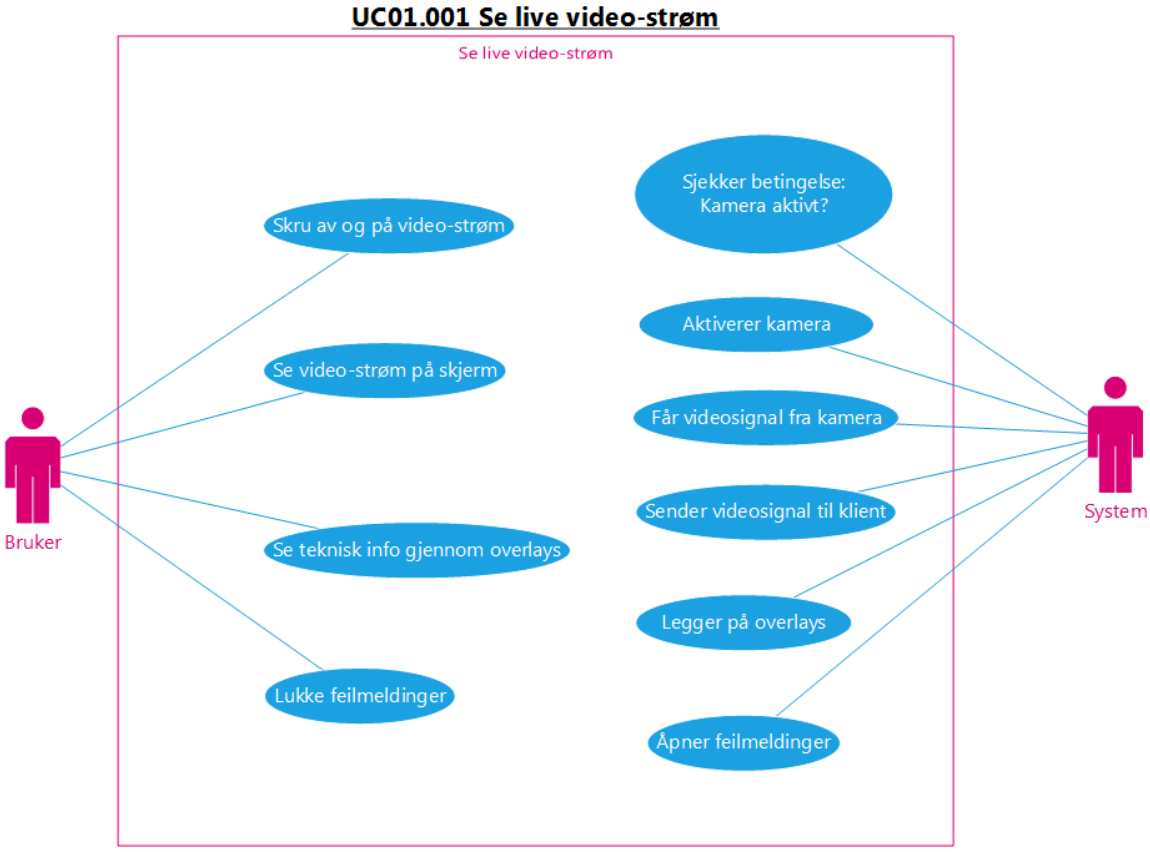
FORSLAG TIL GUI 1 – 25.02.2016

Dette er første utkastet vi kom opp med. Hver av «gråsonene» kommer fram ved å trykke på en «meny»-knapp. Venstre «meny» består av det tekniske man kan gjøre. Mens høyre «meny» skal man kunne «aktivere» elementene. Når man lukker «menyen» så skal de aktiverte elementene være synlig å gi informasjon knyttet til navnet. Det vi si om man «aktiverer FPS» så vil man få oppgitt denne informasjonen når «menyen» er lukket.

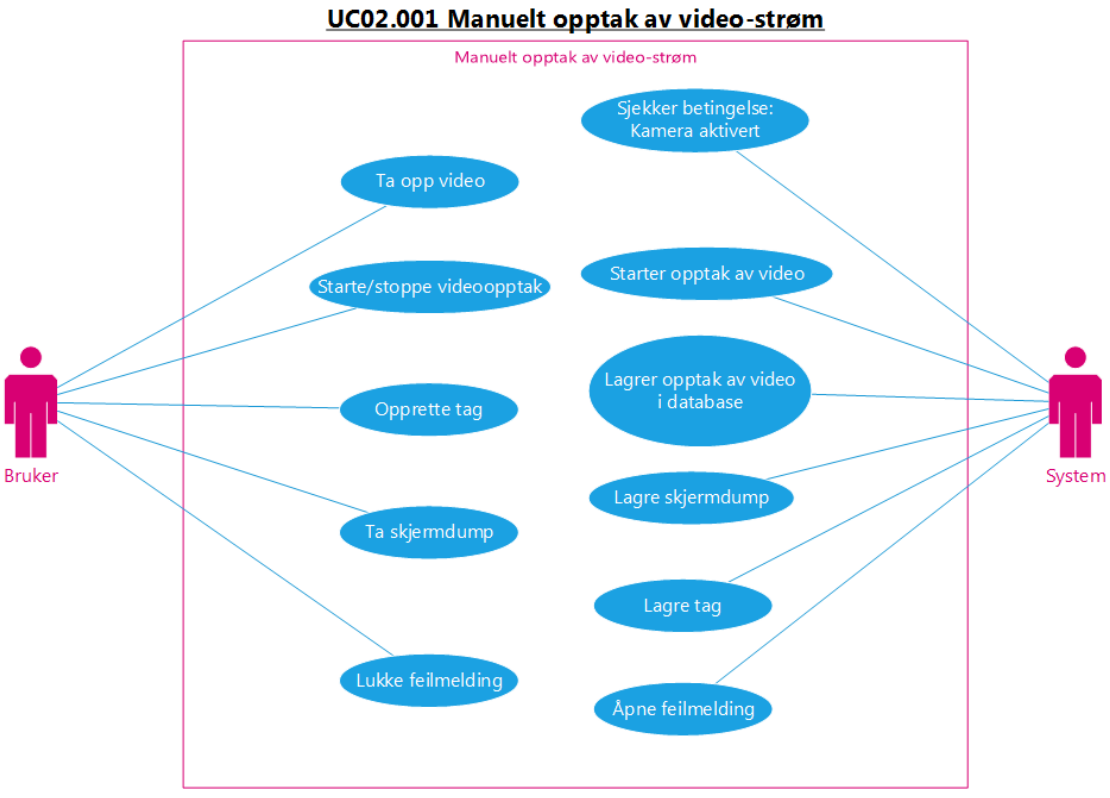


Figur 10 - Første utkast til GUI

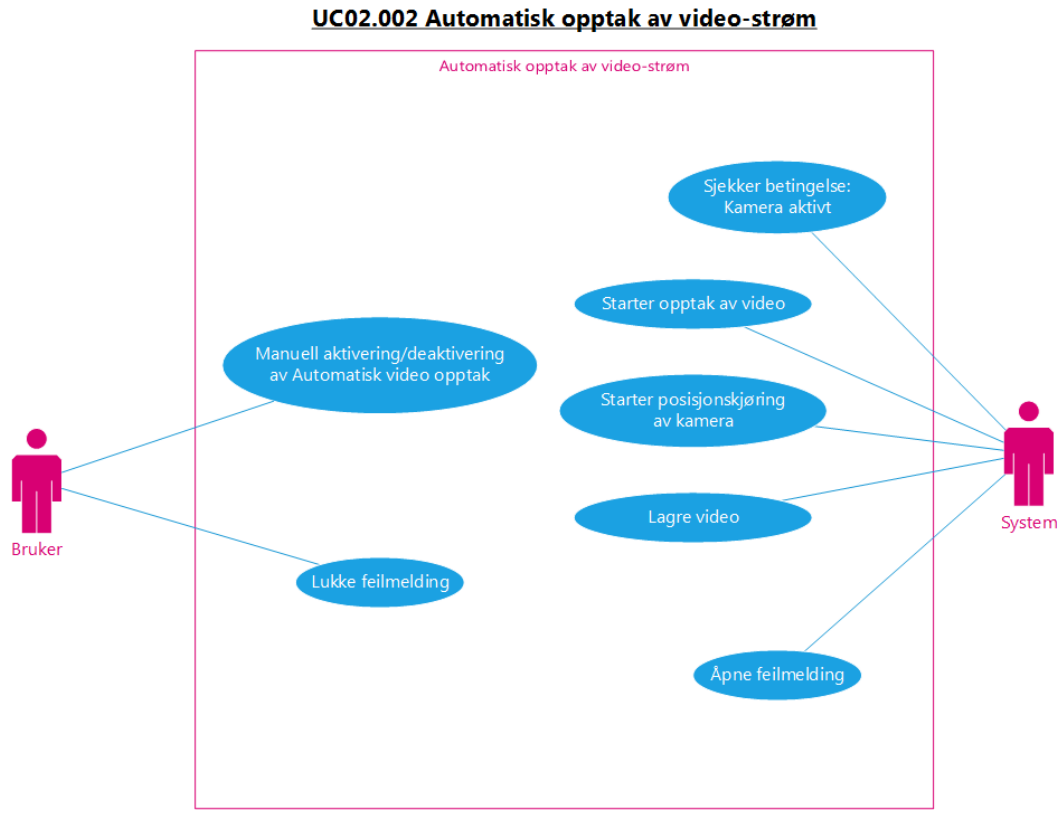
USE CASE DIAGRAMMER



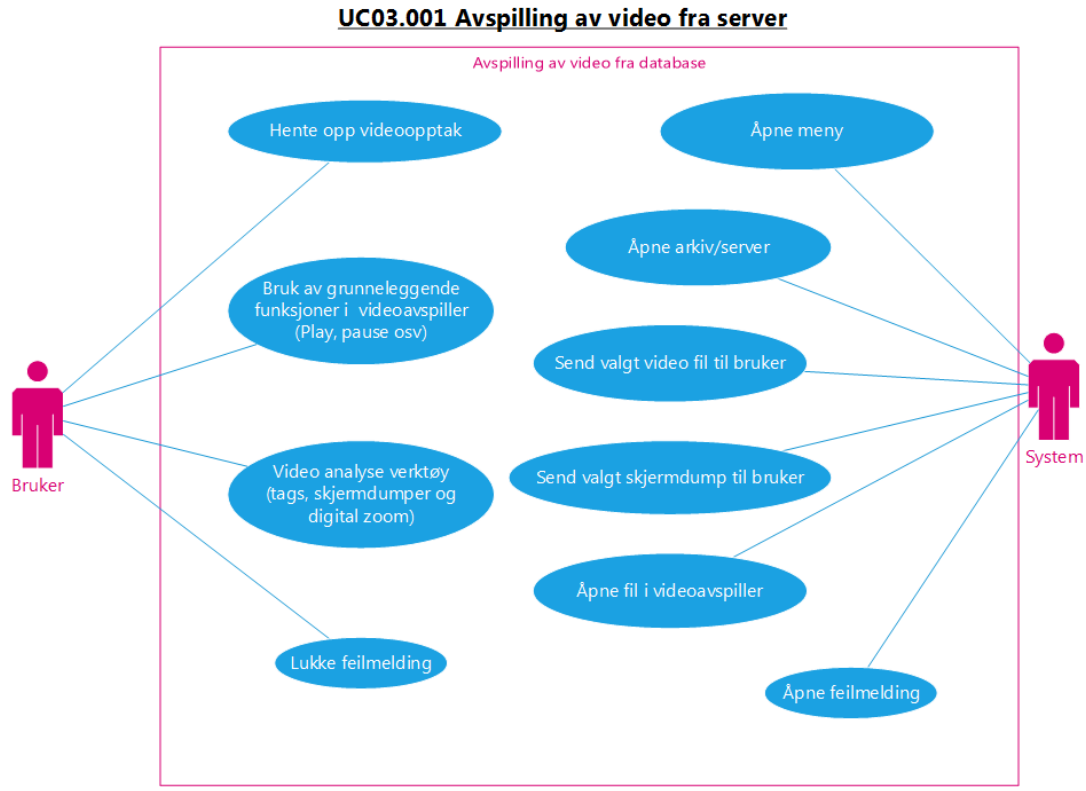
Figur 11 - Use case live stream



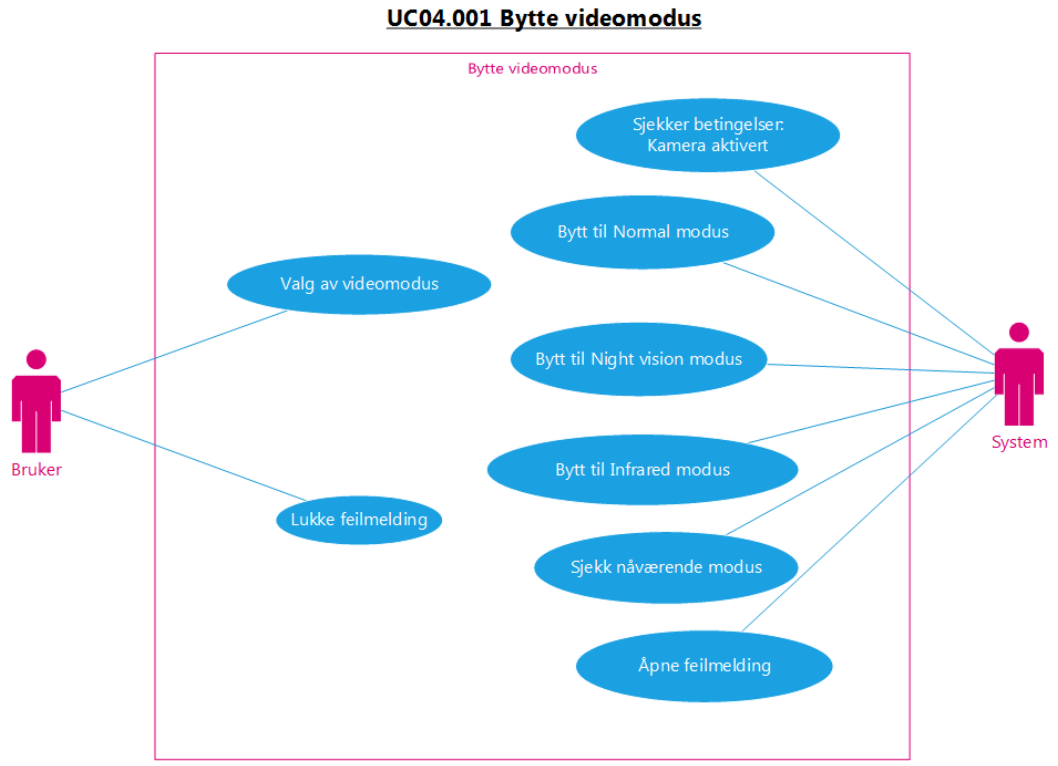
Figur 12 - Use case manuell opptak



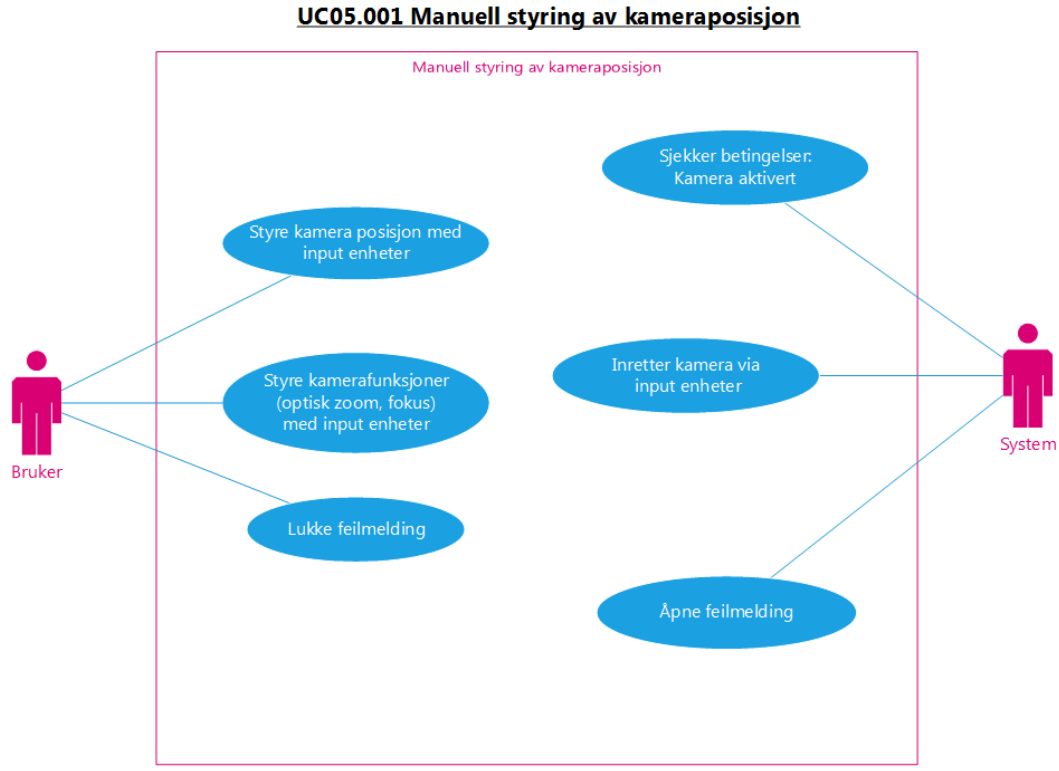
Figur 13 - Use case automatisk opptak



Figur 14 - Use case avspilling av video



Figur 15 - Use case bytte videomodus



Figur 16 - Use case manuell styring

TIDLIGERE DOKUMENTHISTORIKK

Tabell 4 - Tidligere dokumenthistorikk

Versjon	Dato	Skrevet av	Beskrivelse
0.1	04.03.2016	Vegard Smedsvik	Første utkast
1.0	09.03.2016	Vegard Smedsvik	Gjennomgang av dokumentet og rettet skrivefeil

REFERANSER

[1] – Resultatet av design: http://www.ntnu.no/lie/fag/maler-standarder/UP/resultat-av-design.html (04.03.2016)
[2] – Designdokument: Bacheloroppgave. Consulant Calendar Overview (04.03.2016)
[3] – UML IN A NUTSHELL: Sinan Si Albir (03.03.2016)
[4] – UML Distilled Third Edition: Martin Fowler (03.03.2016)
[5] – Schaum's Outline's UML: Simon Bennett, John Skelton (17.02.2016)

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

ANALYSEDOKUMENT – v 2.0

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENT 4

1.1 OVERSIKT 4

1.2 SYSTEMOVERSIKT 4

1.3 HISTORIKK 5

2 REFERANSER, FIGURER OG TABELLER 5

2.1 AKRONYMER OG FORKORTELSER 5

2.2 ORDBOK 5

2.3 TABELLER 6

2.4 FIGURER OG BILDER 6

3 VIDEOSTRØM 6

3.1 FORSINKELSE 6

3.1.1 FORSINKELSE FRA KAMERA OG SERVER 8

3.1.2 FORSINKELSE FRA NETTVERK 10

3.1.3 FORSINKELSE HOS KLIENT 11

3.2 HARDWARE 11

3.2.1 AXIS V5915 VIDEOKAMERA 11

3.2.2 PICOLO HD H.264 12

3.3 HARDWARE - ALTERNATIVE LØSNINGER 12

3.3.1 PICOLO.NET HD4 ENCODER 12

3.4 SOFTWARE 13

3.4.1 QT QUICK 13

3.4.2 FFMPEG 13

3.4.3 VLCLIB 13

3.4.4 UNREAL MEDIA SERVER 14

3.4.5 PROCESS EXPLORER 14

3.4.6 OPEN BROADCASTER SOFTWARE 14

3.5 VIDEO FORMATER 14

3.5.1 H.264 14

3.6 PIXEL FORMAT 15

3.6.1 YUV 15

3.6.2 RGB 15

4 KOMMUNIKASJON 15
4.1 PROTOKOLLER - INTERN DATAKOMMUNIKASJON 15
4.1.1 INTERCOM DDS 15
4.1.2 UDP 15
4.1.3 SCTP 15
4.1.4 TCP 16
4.2 PROTOKOLLER - VIDEOSTRØM 16
4.2.1 REAL-TIME TRANSPORT PROTOCOL (RTP) 16
4.2.2 REAL-TIME STREAMING PROTOCOL (RTSP) 16
4.2.3 USER DATAGRAM PROTOCOL (UDP) 16
4.3 TCP VIRKEMÅTE 16
5 KAMERA 17
5.1 VALG AV KAMERA 17
6 EVALUERING OG RESULTATER 18
6.1 VIDEOSTRØM - FORSINKELSE 18
6.1.1 H.264 NETTVERKSSTRØM FRA KAMERA – VLC KLIENT 18
6.1.2 VLC – UNREAL MEDIA SERVER – VLC KLIENT 18
6.1.3 UNREAL LIVE SERVER – UNREAL MEDIA SERVER – VLC KLIENT 19
6.1.4 FFMPEG – UNREAL MEDIA SERVER – VLC KLIENT 19
6.1.5 OBS – UNREAL MEDIA SERVER – VLC KLIENT 21
6.2 MMI-ENHETER 21
6.2.1 TOBII EYEX EYETRACKER 22
6.2.2 3D MUS SPACE NAVIGATOR 22
6.2.3 MULTI TOUCH CONTROL WHEEL 23
VEDLEGG 25
HISTORIKK 25
REFERANSER 25

1 DOKUMENT

Dette dokumentet omfatter et analysedokument for systemet Real Time Observation.

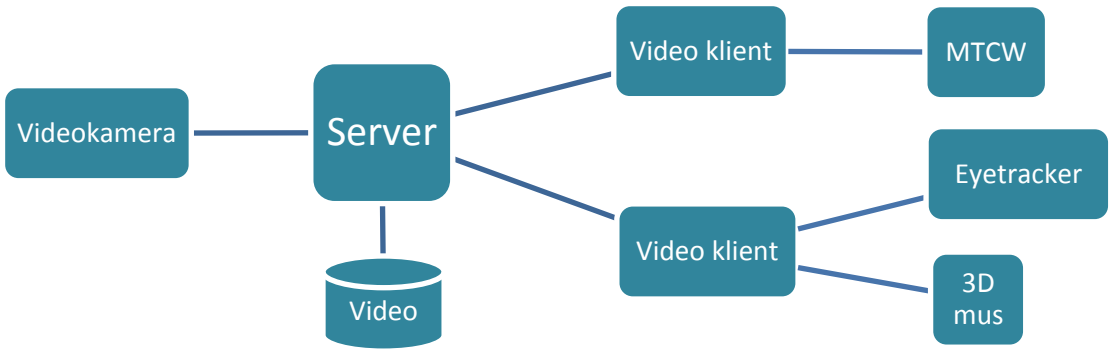
1.1 OVERSIKT

Mye av oppgaven går ut på å utføre analyse av video og løsninger, derfor har vi valgt å samle all analyse i dette dokumentet. Vi vil også dokumentere de ulike løsningene vi har prøvd ut før vi har funnet endelig og fungerende løsning. Fremgangsmåte for analysering av video vil også fremkomme i dette dokumentet.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 – Systemoversikt

1.3 HISTORIKK

Se vedlegg for tidligere dokumenthistorikk.

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
1.0	09.03.2016	Svein Bekkevold	Gjennomgang av skrivefeil og oppdatering til v1.0
1.1	09.05.2016	Alexander Schinnes	Lagt til en del rundt videostrøm, analyse og forsinkelse.
2.0	19.05.2016	Carl Peder Jansen	Fylt inn info på 3D space navigator, deretter gjennomgang av dokumentet.

2 REFERANSER, FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
GUI	Graphical User Interface
IP	Internet Protocol
UI	User Interface
FPS	Frames per Second

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 5
Tabell 2 - Akronymer og forkortelser 5
Tabell 3 - Ordbok 5
Tabell 4 – Historikk 25

2.4 FIGURER OG BILDER

Figur 1 – Systemoversikt 4
Figur 2 - Videostrøm prosess 7
Figur 3 - Hvordan måle forsinkelse ([2] - Bilde fra Axis dokumentasjon) 8
Figur 4 - Nettverksstrøm ([2] - Bilde fra Axis dokumentasjon) 10
Figur 5 - Picolo.net HD4 forsinkelse ([5] – utklipp fra datablad) 12
Figur 6 – Axis og Canon 17
Figur 7 – Venstre /høyre styring 22
Figur 8 – Opp/ned styring 22
Figur 9 – Zoom inn/ut 22
Figur 10 - Continuous move for MTCW 23
Figur 11 - Absolutt styring 24
Figur 12 - Relativ styring 24
Figur 13 - Zoom in, MTCW 24
Figur 14 - Zoom ut, MTCW 24

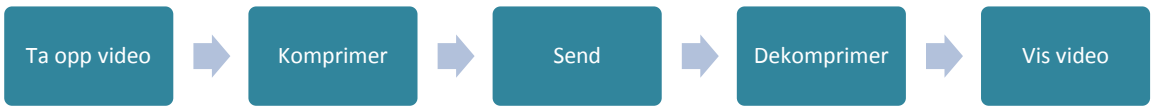
3 VIDEOSTRØM

Vi skal motta et videobilde i nåtid fra et kamera inn til en server. Fra denne serveren skal videobildet kunne streames ut til flere klienter over lokalnettverket. På klientene skal videoen kunne vises i en Qt Quick applikasjon. I tillegg skal man kunne lagre video fra kameraet på server gjennom klienten. For å finne best mulig løsning på dette problemet kreves det mye analyse av hardware og software løsninger.

3.1 FORSINKELSE

Fordi forsinkelsen på videostrømmen er veldig viktig for vårt produkt, vil vi gi noe grunnleggende info om dette. Forsinkelse kan defineres som tiden det tar fra noe blir sendt fra en kilde og til den samme informasjonen blir mottatt i siste ledd av kjeden. For oss blir dette fra noe skjer i virkeligheten som kameraet oppfatter, og til det vises i videoavspilleren hos klienten. Mer spesifikt tiden fra et frame sendes fra kamera og til den samme framen blir vist på skjermen. Dette kan også kalles «sensor-to-screen» forsinkelse.

Hele sekvensen fra en video tas opp til den vises hos klienten kan kort forklares med følgende prosess:



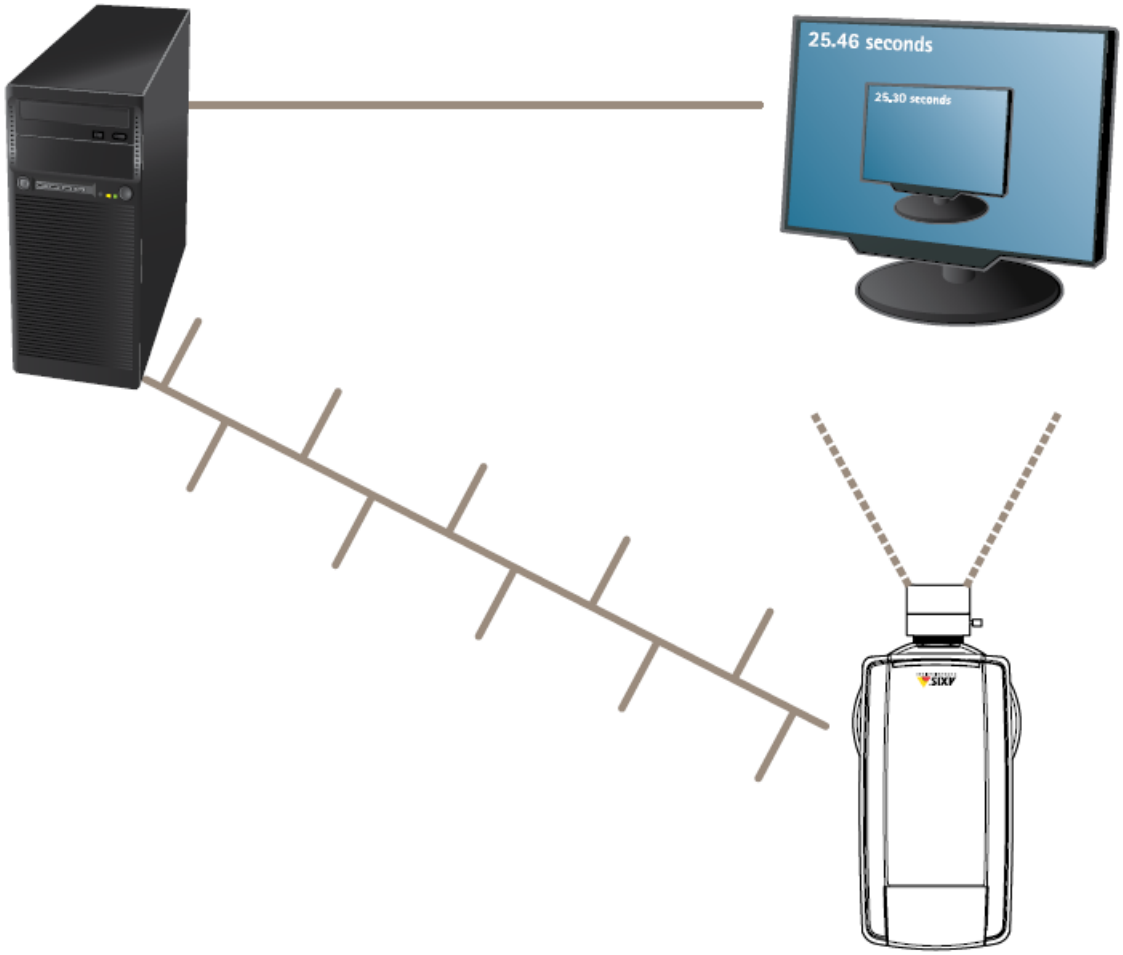
Figur 2 - Videostrøm prosess

Hvert av disse trinnene vil produsere eller etterlate seg en forsinkelse. Hvis vi summerer alle disse forsinkelsene vil vi få en total forsinkelse. Det er denne vi vil holde så lav som mulig, helst under 200ms. I denne prosessen kan vi dele forsinkelsen inn i tre deler:

1. Forsinkelse forårsaket av kameraet og videokort (bilde prosessering / enkoding forsinkelse)
2. Forsinkelse forårsaket av nettverket (forsinkelse ved overføring)
3. Forsinkelse forårsaket av mottakeren (klient buffer, dekoding og forsinkelse ved visning)

Vi må analysere alle delene av prosessen for å skape et best mulig system med minst mulig forsinkelse. I tillegg vil selvfølgelig bildekvalitet og oppløsning på videoen sendt fra kameraet ha innflytelse på forsinkelsen.

For å kunne vite konkret hvor mye forsinkelsen er på, så må denne måles. Forsinkelse blir normalt oppgitt i sekunder eller millisekunder. En enkel metode å måle forsinkelse på er å benytte en tidtaker som teller visuelt på skjermen, for så å sette videokameraet til å ta opp innholdet på skjermen og vise kamerabildet på samme skjerm. Da vil disse to bildene begge inneholde telleren. Hvis man tar skjermdump av skjermbildet kan man da sammenligne klokken på disse to bildene. Dette er forklart visuelt i figur 3 på neste side. På denne figuren kan vi se at tidsdifferansen er 460ms – 300ms, noe som gir oss en forsinkelse på 160ms.



Figur 3 - Hvordan måle forsinkelse ([2] - Bilde fra Axis dokumentasjon)

Denne løsningen vil imidlertid fungere dårlig der det benyttes video med lav bildefrekvens. Da vil det kunne være en potensielt stor feilmargin fordi videoen oppdateres veldig tregt.

3.1.1 FORSINKELSE FRA KAMERA OG SERVER

Forsinkelsen i en videostrøm kan komme fra flere steder. Både hardware og software vil påvirke hvor stor den totale forsinkelsen vil bli. I dette delkapittelet vil vi gå nærmere inn på den delen av forsinkelsen som oppstår på server og kamerasiden.

Bildefrekvens

Det opptaket eller den videostrømmen som et kamera gir fra seg inneholder en bildefrekvens, heretter kalt fps («frames per second»). En video består av flere bilder i sekundet som bildebrikken på kameraet tar opp. Det vil være forskjellig forsinkelse ut ifra hvor mange bilder i sekundet man velger. I første omgang kommer vi til å fokusere på å benytte 25fps og 50fps. Med 25fps vil bildesensoren ta et bilde/frame hvert 1/25 del av et sekund, noe som vil gi en opptaksforsinkelse på 40ms. Med 50fps vil sensoren ta et bilde hvert 1/50 del av et sekund, noe som vil redusere opptaksforsinkelsen til 20ms. I utgangspunktet vil vi da ende opp med en mindre forsinkelse ved å

øke bildefrekvensen, men samtidig vil videoen inneholde dobbelt så mye bildeinformasjon per sekund. Dette vil da igjen ta lengre tid å behandle/enkode og tar dermed lengre tid å sende over nettverket. Den forsinkelsen som blir spart ved økt bildefrekvens må veies opp mot den økte forsinkelsen det tar å behandle, sende og vise videoen.

Forsinkelse grunnet bildeforbedring

Hver eneste frame som blir tatt opp av kameraet vil gå gjennom en forbedringsprosess i kameraet. Dette kan være for eksempel de-interlacing, skalering, oppløsning, rotasjon og flere utgangsstrømmer. Vi kommer ikke til å rotere bildet fra kameraet fordi den kommer til å stå riktig vei i forhold til omgivelsene, dermed trenger ikke kameraet å bruke tid på å rotere bildet før det sendes.

Oppløsningen vi velger vil imidlertid ha påvirkning på forsinkelsen. En høy oppløsning vil føre til flere piksler som prosessoren må enkode, i tillegg vil det føre til en større mengde datapakker som må sendes over nettverket. Hvis det er begrenset båndbredde på nettverket vil dette føre til en økt forsinkelse.

Kameraet har flere utgangsstrømmer og kan sende en video over HDMI, SDI og nettverkskabel. Alle disse strømmene blir enkodet med samme prosessor. Dermed vil det føre til en økt forsinkelse hvis det benyttes flere utgangsstrømmer fordi alle da må enkodes på samme tid i samme prosessor.

Forsinkelse grunnet komprimering

Etter at et bilde har blitt prosessert vil det bli enkodet for å komprimere den mengden data det trenger å overføre. Selve komprimeringen inneholder ofte flere matematiske algoritmer som fjerner noe bildeinformasjon. Dette vil ta tid avhengig av hvor mye data som trengs å prosesseres.

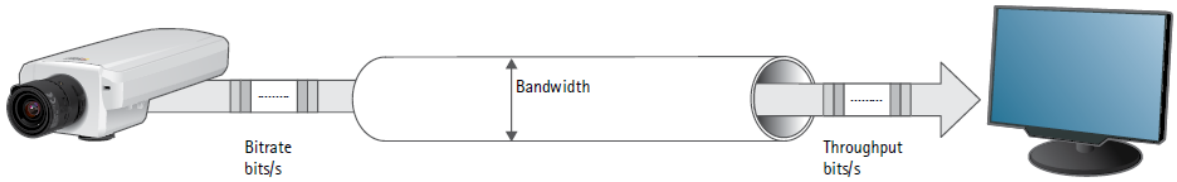
Under komprimeringen, også kalt enkoding, vil det være mye som kan påvirke forsinkelsen, blant annet hvilket format det skal komprimeres til. Vi kommer til å benytte H.264 kompresjon. Dette er en relativt avansert kompresjon sett i forhold til for eksempel MJPEG. Det er bare snakk om noen mikrosekunders forsinkelse i forskjell som det tar å pakke ned videostrømmen for de to formatene. Den største forskjellen ligger ofte i utpakkingen/dekodingen hos klient. Den H.264 datastrømmen som Axis kameraene produserer krever at dekoderen bufrer minst en frame, mens MJPEG dekoding ikke krever noe buffer.

På den annen side så er H.264 komprimeringen ofte veldig god. Den er veldig effektiv og vil produsere et mindre antall datapakker å sende over nettverket. Dette vil da selvfølgelig ha en positiv effekt på den totale forsinkelsen.

Det er viktig å tenke på den totale forsinkelsen når man velger innstillinger for komprimering. På den ene siden kan man si at en avansert komprimeringsalgoritme vil ta lenger tid å enkode og dekode, mens på den andre siden vil den reduserte datamengden som blir sendt over nettverket kanskje veie opp for den ekstra tiden det tar å enkode/dekode denne. Dette er vurderinger som må testes og analyseres nøyere. Vi kommer ikke til å benytte lyd fra kameraet, dermed kan vi eliminere den prosesseringstiden som dette ville tatt.

3.1.2 FORSINKELSE FRA NETTVERK

Fordi videoen som blir tatt opp av kameraet skal distribueres videre til flere forskjellige klienter over nettverket, må denne tiden også tas med i beregningen av den totale forsinkelsen. Båndbredden på nettverket bestemmer hvor mye informasjon som kan sendes på samme tid. Jo større båndbredde, desto mer videoinformasjon kan du sende over på samme tid uten forsinkelse. Båndbredden avgjøres av alle komponenter data som sendes gjennom nettverket. Dette kan være svitsjer, rutere, kabler, proxier, protokoller etc. Fordi vi kun kommer til å sende data over det lokale nettverket, vil denne forsinkelsen være noen få ms, altså tilnærmet ubetydelig for vår del.



Figur 4 - Nettverksstrøm ([2] - Bilde fra Axis dokumentasjon)

Den totale forsinkelsen som er forårsaket av nettverket er i hovedsak grunnet tre ting. Komponentene som videoen må sendes gjennom, altså den tilgjengelige båndbredden på nettverket, størrelsen på datapakken som sendes i bitrate og hvilken overføringsprotokoll som benyttes.

Vi har gjort en del research på hvilken protokoll som egner seg best for å overføre real-time video. Vi endte til slutt opp med TCP og UDP som aktuelle protokoller. TCP er en sikrere protokoll for overføring, men den vil også føre til noe mer forsinkelse. TCP krever et «handshake» ved opprettelse av connection og for hver eneste datapakke som skal sendes og mottas. Hvis en datapakke som blir sendt fra for eksempel server til klient ikke mottar en bekreftelse på at den er mottatt, vil denne datapakken forsøkes å sendes igjen inntil den får en bekreftelse. Dette gjør TCP til en svært sikker og pålitelig overføringsprotokoll, men den er også noe tregere enn UDP.

UDP krever ingen bekreftelse på at en datapakke er mottatt før den sender neste. Den vil kun sende alt den får beskjed om, men bryr seg ikke om datapakkene mottas i riktig rekkefølge eller om de mottas i det hele tatt. Dette gjør at UDP er en rask overføringsprotokoll, men langt ifra like pålitelig som TCP. For de fleste videoer og bilder kan det mistes noen datapakker nå og da uten at dette gjør et stort utslag på resultatet. Det vil bli noe dårligere bildekvalitet, men det skal mistes en del datapakker før dette blir svært synlig på videoen.

I første omgang kommer vi nå til å gå for UDP som vår overføringsprotokoll. Den kan deles videre inn i unicast og multicast avhengig av antall klienter og lignende. Vi kommer til å forsøke og streame video til en UDP multicast adresse som alle klientene i vårt system skal motta video på.

3.1.3 FORSINKELSE HOS KLIENT

Klienten skal motta videostrømmen fra nettverket, pakke den ut, dekode og rendre denne i videoavspilleren. Dette vil også føre med seg en forsinkelse på videostrømmen. Her vil også CPU kapasiteten, operativsystemet, nettverkskortet og grafikkortet på klientmaskinen spille en rolle.

Normalt vil det også ligge noe informasjon i bufferen på klientmaskinen når det streames fra nettverket for at ikke dekoderen skulle måtte vente på at datapakker blir mottatt fra nettverket. Dette kalles ofte en avspillingsbuffer. Alle videoer som skal spilles av fra nettverket trenger dette bufferet. Det vil si at videoen lastes ned i et buffringsminne før det blir spilt av. Dette for å unngå at videoen skal hakke ved små forsinkelser eller frafall av nettverk i korte perioder på noen millisekunder. Størrelsen på denne bufferen er forskjellig fra program til program. Vi kommer muligens til å benytte et VLC-bibliotek som integreres i Qt for å spille av video. VLC har en standard avspillingsbuffer som medfører en forsinkelse på 20ms, mens Quicktime har en avspillingsbuffer som kun medfører 5ms forsinkelse. Disse verdiene gjelder riktignok for avspillingen av video fra fil. For nettverksavspilling gjelder andre verdier. Mer detaljert info om dette ligger under kapittel 3.3. Størrelsen på denne bufferen kan i de fleste tilfeller justeres, men ved å velge en for liten avspillingsbuffer vil dette øke jitter. Jitter er en betegnelse på signalets ustabilitet, at pulsens flanker ikke kommer helt regelmessig. Dette måles i prosent og beskriver avviket i tid relativt til tiden for en hel periode. Hvis det er mye jitter så får vi en reduksjon i prosessens kvalitet. Så det må finnes en balanse mellom jitter og akseptabel forsinkelse.

Forsinkelsen som går med til dekodingen av videoen, vil være noe avhengig av hvor kraftig grafikkortet er. Det er normalt raskere å dekode ved hjelp av hardware istedenfor software, men vi har ingen annen hardware å dekode med enn det medfølgende grafikkortet på klientmaskinene. Den H.264 datastrømmen som produseres av Axis krever at dekoderen bufrer minst en frame før dekodingen kan starte.

Enhver skjerm har også en bestemt tid for hvor ofte bildet på skjermen skal oppdateres. De fleste skjermer har en oppdateringsfrekvens på rundt 14-15 ms, mens skjermer som er beregnet for krevende bildebehandling kan ha en lavere oppdateringsfrekvens på rundt 4-5 ms.

3.2 HARDWARE

Vi har fått utlevert hardware utstyr fra oppdragsgiver slik at vi har et utgangspunkt å begynne med. Dette er et Picolo HD H.264 videokort i tillegg til et videokamera som har blitt bestilt i etterkant av prosjektstart. Utstyret er nærmere beskrevet i eget teknologidokument.

3.2.1 AXIS V5915 VIDEOKAMERA

Streamen kameraet sender over nettverket har et delay på 30-50 ms fra kameraet sin side. Dette er opplyst i Axis sin dokumentasjon på kameraet. Denne forsinkelsen har noe påvirkning på totalforsinkelsen, men er generelt ikke den største bidragsyteren. Vi vil vurdere forsinkelsen på nettverksstrømmen opp mot strømmen fra HDMI for å se hva som vil være den beste løsningen for oss.

3.2.2 PICOLO HD H.264

Dette er videokortet vi har mottatt fra oppdragsgiver. All video fra HDMI-kabelen går gjennom dette kortet. En begrensning vi har funnet med dette kortet er at det kun støtter 30fps for HD 1080p, mens den støtter 60fps for HD 1080i. I tillegg har det en relativt høy forsinkelse på videostrømmen som legges igjen i komprimeringen. Vi har vært i dialog over mail med personell fra Euresys, som produserer og leverer Picolo-kortene. Der har vi fått oppgitt informasjon angående forsinkelse i kortet. Det er omtrentlig 80ms forsinkelse på raw strømmen fra kortet, mens det er omtrentlig 400ms forsinkelse på den hardware komprimerte H.264 strømmen ut fra kortet. Grunnen til at det er så stor forsinkelse på den komprimerte strømmen er at H.264 enkoderen trenger flere bilder/frames før den kan starte komprimeringen. Enkoderen er rask nok til å komprimere videobildet, men siden bildets «pipeline» trenger å «fylles» før den kan komprimere vil det alltid ligge en forsinkelse på strømmen som går ut av kortet sett i forhold til den strømmen som kommer inn.

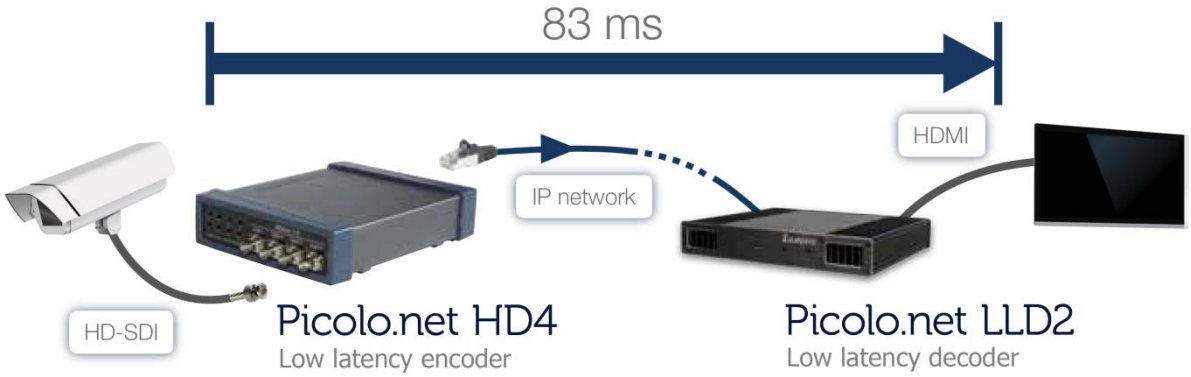
3.3 HARDWARE - ALTERNATIVE LØSNINGER

Fordi Picolo HD H.264 videokortet ikke klarer å håndtere/enkode videoen raskt nok for oss, så har vi valgt å se på alternativer som vil løse de problemene vi har.

3.3.1 PICOLO.NET HD4 ENCODER

Det og enkode HD video fører ofte til relativt store forsinkelsesutfordringer med tanke på real-time systemer. For å løse dette problemet har Euresys utviklet en enkoder og dekoder for å minimere forsinkelsen av H.264 enkoding og dekoding ved å gå direkte via nettverk uten et videokort.

Picolo.net HD4 har fire HD-SDI innganger med enkoding som kan streame video direkte over et IP nettverk med støtte for full HD (1080p30). For å motta denne videoen igjen kan det kobles til en Picolo.net LLD2 dekoder.



Figur 5 - Picolo.net HD4 forsinkelse ([5] – utklipp fra datablad)

Fra figur 5 kan vi se at total forsinkelse fra kamera til avspilling er oppgitt til å være 83 ms ved hjelp av denne løsning til Euresys. Det er i tillegg støtte for å streame på nettverket med flere ulike protokoller. Blant annet UDP unicast og multicast.

3.4 SOFTWARE

Vi har prøvd ut flere forskjellige løsninger for software som skal sende og motta videostrømmen. Serveren som skal sende video trenger ikke å kunne integreres i Qt Quick etter avtale med oppdragsgiver, så her kan vi også velge ferdig utviklet software hvis dette gir det beste resultatet.

3.4.1 QT QUICK

Vi skal spille av videoen fra en nettverksadresse. Dette har vi ikke funnet noen løsninger på for å gjøre i Qt uten bruk av en tredjeparts software. Vi har derfor forsøkt ulike biblioteker som kan integreres i Qt. Vi prøvde også å streame videoen direkte fra Qt, men har heller valgt å la dette gå gjennom annen software.

3.4.2 FFMPEG

Ffmpeg [11] er et open source software for behandling og avspilling av multimedia, i tillegg til at det inneholder flere sett med ulike biblioteker som libavcodec og libavformat. Ffmpeg kan bli brukt til å innhente informasjon som f.eks «Frames Per Second (FPS)», konvertere video til andre formater, spille av video filer og nettverksstrømmer og til å dekode data. Ffmpeg inneholder mer enn 100 forskjellige kodeker, og blir brukt av flere andre applikasjoner som VLC, HandBrake og Blender. Vi har forsøkt å benytte ffmpeg for streaming av video. Den kan integreres i ulike applikasjoner, eller benyttes som en «stand alone» applikasjon som kan benyttes gjennom console vindu. Vi benyttet ffmpeg biblioteket alene uten påvirkning fra noen andre applikasjoner for å se hvilken ytelse dette gav.

3.4.3 VLCLIB

Streamingen fra VLC har vi funnet ut at går noe tregt, mens avspillingen så langt har fungert bra. Derfor har vi integrert VLClib i klientapplikasjonen for avspillingen av lokal video og real-time video fra nettverket. Avspillingen og videokvaliteten er god, men VLC fører meg seg et delay på klientsiden. VLC har et default avspillingsbuffer på 1000ms. Det vil si at den laster inn 1000ms av videoen før den begynner å spille av innholdet. Dette gir en meget god flyt i videostrømmen uten hakking, men det er dessverre dårlig egnet for real time strømmen med ønske om lavest mulig forsinkelse. I den visuelle versjonen av VLC er det mulig å endre størrelsen/tiden på dette bufferet, og derfor måtte det også finnes en mulighet for å kunne endre størrelsen på dette bufferet i vlclib. Etter en del research fant vi ut hvordan dette kunne gjøres, og vi har nå fått ned tiden til 300ms. Går vi under 300ms blir det fort problemer med «jitter» og videobildet hakker.

3.4.4 UNREAL MEDIA SERVER

[12] Dette er et ferdig utviklet software som kun er beregnet på å streame ulike medier ut på nettverket. Det kan velges om en vil benytte rå strøm eller komprimert H.264 strøm fra Picolo kortet.

3.4.5 PROCESS EXPLORER

For å gjøre analyse av hardwareforbruk som CPU, RAM, nettverkstrafikk og lignende benytter vi Microsoft Process Explorer. Dette er en noe mer avansert versjon av den klassiske oppgavebehandleren i Windows. Det er også mulig å finne ut hvilke dll filer som blir benyttet av de forskjellige applikasjonene. Dette er noe vi kan bruke til feilsøking blant annet.

3.4.6 OPEN BROADCASTER SOFTWARE

Open Broadcaster Software (OBS) er et gratis open source produkt for videoopptak og live streaming. Den enkoder videoen med H264 (x264) og AAC for lyd. Det er skrevet i C og C++ og har støtte for real-time kilder, videokort, scene komposisjon, enkoding, opptak og kringkasting. Overføringen av data gjøres med Real Time Messaging Protocol, og kan bli sendt til en hvilken som helst RTMP destinasjon.

3.5 VIDEO FORMATER

For å definere et videoformat kan vi si at det er en kombinasjon av en beholder eller kontainer for medieinnhold og de nødvendige kodekene som trengs for å tolke innholdet. Mediekontaineren beskriver for eksempel hvor de forskjellige dataene er lagret, filsystemet og hvilke kodeker som trengs. Dataen som er lagret inne i kontaineren har blitt enkodet av kodeker. For å kunne spille av dette innholdet trengs det kodeker som kan dekode innholdet igjen. Det er kodekene som utfører komprimeringen av videoene.

3.5.1 H.264

H.264 er det videoformatet som er ønsket å bruke av oppdragsgiver, derfor vil vi legge størst vekt på denne komprimeringen. H.264 utfører en meget god videokomprimering. Den reduserer videostørrelsen meget bra uten å gå på stor bekostning av videokvaliteten. H.264 er derfor et meget populært format fordi man trenger mindre båndbredde for overføring og mindre lagringsplass. Det er mulig å bruke H.264 for tapsfri komprimering, men det benyttes vanligvis komprimering med noe tap av data. Det er tre forskjellige typer frames som kan benyttes innenfor H.264, I, P og B. I er individuelle frames som inneholder all informasjon som trengs i forbindelse med dekodingen. P har referanse til tidligere enten P eller I frames slik at ikke alt innholdet trenger å dekodes. Mens B frames har referanse til både tidligere og senere frames. Det negative med denne framen er at den ikke inneholder all tilgjengelig informasjon selv. Dermed vil en liten forstyrrelse på nettverket føre til feil i enkodingen.

3.6 PIXEL FORMAT

3.6.1 YUV

YUV er et format for og enkode farger ved hjelp av de tre verdiene Y, U og V. Y representerer lysstyrken til fargen, også kalt luma. U og V komponentene kalles «chroma» eller fargeforskjell og representerer fargeinformasjonen. Ved svart/hvitt video vil det kun være Y komponenten som benyttes, fordi U og V inneholder fargeverdier.

3.6.2 RGB

RGB står for Rød, Grønn og Blå. Dette er primærfargene i et fargespekter. Ved hjelp av å endre mengden av rød, grønn og blå kan stort sett alle farger «simuleres». Dette er en additiv fargemodell, noe som vil si at de tre verdiene til fargene blir lagt sammen og resultatet blir én farge.

4 KOMMUNIKASJON

4.1 PROTOKOLLER - INTERN DATAKOMMUNIKASJON

Intern datakommunikasjon handler om kommunikasjonen mellom server og klienter på alt som ikke handler om videostrømmen. For kommunikasjon angående videostrøm, se info under protokoller for videostrøm. Vi har gjort en liten analyse av hvilken type protokoll vi skal bruke til kommunikasjon internt i systemet. Fra starten hadde vi flere alternativer som: InterCOM DDS, UDP, SCTP og TCP.

4.1.1 INTERCOM DDS

InterCOM DDS (Intern Communication Data Design System) er et system som er utviklet av Kongsberg Gruppen og det ble vurdert om vi skulle bruke dette fordi det er implementert i de fleste av arbeidsgiverens systemer. Etter råd fra ekstern veileder om at det er et komplekst system å sette seg inn i for en hovedoppgave, valgte vi å ikke gå for den løsningen.

4.1.2 UDP

UDP (User Datagram Protocol) er en minimal meldingsorientert nettverksprotokoll for forbindelsesløs overføring av informasjon, og opererer i transportlaget i OSI-modellen for datanett. Dette er en løsning som ikke gir noen garanti for at pakker som sendes i en ende blir mottatt i den andre enden. I forhold til kommunikasjon internt kan det da bli problematisk om man skulle sende en kritisk melding om error som ikke går igjennom. Vi har derfor valgt å ikke ta i bruk UDP fordi det er usikkert.

4.1.3 SCTP

SCTP (Stream Control Transmission Protocol) er en avansert protokoll for transportlaget i OSI-modellen, som kanskje vil avløse TCP en gang. Grunnen til at vi har valgt bort SCTP er fordi det er

såpass nytt i forhold til TCP og fordi det er generelt mer info og kunnskap om TCP som gjør det lettere for oss i forhold til research.

4.1.4 TCP

TCP (Transmission Control Protocol) er en nettverksprotokoll som gir pålitelig overføring av informasjon. TCP passer på at ingen pakker som blir sendt forsvinner, men forsikrer seg om at pakkene kommer fram, selv om det tar lenger tid enn f.eks. UDP. TCP-modulen i mottakerenden sender en kvittering om at alle pakkene er mottatt. Hvis kvitteringen ikke er mottatt innen et visst tidspunkt, vil et tidsavbrudd oppstå. Da vil sender anta at pakken er tapt og pakken må sendes på nytt. TCP sjekker også at datastrømmen ikke er skadd ved å bruke en sjekksum. Sjekksummen blir beregnet av senderen og kontrollert hos mottaker for hver pakke.

4.2 PROTOKOLLER - VIDEOSTRØM

4.2.1 REAL-TIME TRANSPORT PROTOCOL (RTP)

Real-Time Transport Protocol (RTP) er en protokoll som tilbyr levering av “end-to-end” real time data, som video og lyd. RTP er vanligvis lagt over UDP, selv om den kan bruke andre protokoller også. Når RTP legges over en annen protokoll vil man kunne bruke multicast. Altså at datapakker sendes til flere forskjellige destinasjoner. RTP brukt alene vil ikke kunne sørge for at datapakkene blir relevert til riktig tid, eller at de i det hele tatt leveres. Det overlater den til transport protokollen underlagt. Hver pakke som blir sendt er uansett nummerert slik at klienten kan pakke de ut i riktig rekkefølge.

4.2.2 REAL-TIME STREAMING PROTOCOL (RTSP)

Denne protokollen gjør det mulig å etablere en kontroll over media strømmen som sendes. RTSP har ingenting å gjøre med transporteringen av selve pakkene, men den kan sammenlignes med en fjernkontroll for TVen. Man kan bruke RTSP til å stoppe, spille av, spole osv. Fordi RTSP ikke håndterer transporteringen kan det brukes flere forskjellige andre protokoller for dette.

4.2.3 USER DATAGRAM PROTOCOL (UDP)

Som nevnt tidligere ønsker vi å bruke UDP til å transportere videoen fra server til klient. Dette vil gi den raskeste overføringen for real time video strømming. Mer rundt UDP er gitt tidligere i dokumentet.

4.3 TCP VIRKEMÅTE

TCP forbindelser har tre faser: opprettelse av en forbindelse, dataoverføring og avslutningen av forbindelsen.

Opprettelse av en forbindelse. En tretrinns handshake blir brukt til å opprette forbindelsen. Typisk skjer dette ved at den ene enden åpner en socket og venter passivt på at noen skal koble seg til. Dette er typisk for server-enden av en forbindelse. Klientsiden utfører en aktiv åpning ved å sende et TCP segment med SYN flagget satt til serveren. Serveren skal da besvare et gyldig SYN(Synchronize)-forespørsel med SYN/ACK. Og til slutt skal klienten igjen svare med et ACK(Acknowledge) som fullfører håndtrykket og oppretter en forbindelse.

Dataoverføringen. I hovedsak så brukes et sekvensnummer for å ordne segmentene i riktig rekkefølge og sjekksummer for å detektere feil i segmentene. Dette bidrar til at vi sikkert får overført alle de pakkene som vi vil ha fra en server til klient.

Avslutning av forbindelsen. I denne fasen benyttes et 4 way handshake. Hver ende av forbindelsen avslutter uavhengig av hverandre. For hvert endepunkt som lukker forbindelsen kreves et par med FIN(Finish) og ACK segmenter.

5 KAMERA

5.1 VALG AV KAMERA

Vi har i denne prosessen sett på en del ulike kameraer som vi har vurdert opp imot ytelse og pris. I starten fikk vi et budsjett på kr 10 000,- å bruke på innkjøp av kamera. Etter en rask research fant vi ut at om vi skulle møte kravene som var satt til kamera skulle oppfylles måtte budsjettet mer enn dobles. HDMI utgang er en stor faktor som drar prislappen opp.

Figur 6 – Axis og Canon

Kameraene som ble vurdert og sammenlignet var Axis V5914/15 og Canon VB M40. Det ble skrevet en kort rapport som sammenlignet de forskjellige kameraene og deres kvaliteter som ble sendt til oppdragsgiver. Oppdragsgiveren valgte å gå for det dyreste og mest avanserte kameraet: Axis V5915 til kr 23 000,-

6 EVALUERING OG RESULTATER

Her vil vi gi en oppsummering av forskjellige løsninger og de resultatene vi har funnet fram til.

6.1 VIDEOSTRØM - FORSINKELSE

Fordi vi har prøvd ut mange forskjellige metoder for å kunne streame en video fra kamera til server og klient vil vi raskt gå gjennom de viktigste alternativene her. Alle forsinkelser som er oppgitt er målt fra end-to-end. Det vil si tiden det tar fra noe skjer i virkeligheten til det blir vist på klientskjermen. Som en sammenligning for å forstå disse verdiene kan vi nevne at Axis har en total forsinkelse på 450ms når videoen blir vist gjennom deres software som er web basert. Alle beregninger er gjort med full oppløsning på 1920x1080 og 30FPS hvis ikke annet er nevnt.

6.1.1 H.264 NETTVERKSSTRØM FRA KAMERA – VLC KLIENT

Nettverkskameraet Axis V5915 kan levere ut en H.264 strøm over RTSP protokollen som kan hentes ut og spilles av direkte i VLClib på klienten. Her skjer all enkodingen og distribueringen av videostrømmen på selve kameraet. Dette er den mest effektive måten å distribuere ut videoen på som vi har funnet så langt og er den metoden vi endte opp med å bruke. Her har vi en total forsinkelse på 550ms. På klientsiden har vi en forsinkelse på 300ms i VLClib. Det vil dermed si at enkodingen og distribueringen av videoen som blir gjort på kameraet tar 250ms. Det er i tillegg utført tester for å se hvordan oppløsning og FPS påvirker forsinkelsen. Resultatene ble som følger:

1080p – 25 FPS gir forsinkelse = 500ms
1080p – 50 FPS gir forsinkelse = 550ms

720p – 25 FPS gir forsinkelse = 450ms
720p – 50 FPS gir forsinkelse = 500ms

480x270 – 25 FPS gir forsinkelse = 500ms
480x270 – 50 FPS gir forsinkelse = 500ms

Forsinkelse: 550ms
CPU forbruk VLC Klient (gj.snitt): 37,2%

6.1.2 VLC – UNREAL MEDIA SERVER – VLC KLIENT

Denne metoden innebærer at VLC plukker opp videoen fra Picolo kortet, enkoder denne og sender den videre til Unreal Media Server som distribuerer strømmen ut på nettverket. Ved avspillingen av videobildet direkte i VLC sin UI med 0ms «live-caching» er det en forsinkelse på 450ms. Det vil si den tiden som Picolo kortet bruker på og enkode videoen uten at VLC transkoder den. For at videostrømmen skal kunne spilles av på nettverket må VLC transkode strømmen selv etter at den er kommet fra videokortet, ellers fungerer det ikke. Når det skal ut på nettverket har vi valgt å benytte UDP fra VLC til Unreal Media Server. Fra Unreal Media Server blir strømmen transportert videre ut på nettverket via RTP. Denne strømmen kan spilles av i VLClib på klient. Den totale forsinkelsen er målt

til 3200ms med dette oppsettet. Her er det da altså dobbel enkoding, fordi VLC må enkode i tillegg til Picolo for at det skal fungere.

Forsinkelse: 3200ms
CPU forbruk Server VLC (gj.snitt): 61,3%
CPU forbruk VLC Klient (gj.snitt): 18,2%

6.1.3 UNREAL LIVE SERVER – UNREAL MEDIA SERVER – VLC KLIENT

Unreal Live Server kan hente opp videoen fra Picolo kortet for så å videresende dette til Unreal Media Server for distribuering til nettverket. I Unreal Live Server kan det velges om det skal benyttes komprimert eller ukomprimert strøm ut fra videokortet, i tillegg til at det kan velges om det skal tilføres software komprimering eller ikke før bildet sendes videre. Her viser det seg i midlertid at videostrømmen ikke kan spilles av på noen måte hvis det velges å hente ukomprimert strøm fra videokortet. Det må velges komprimert strøm. I tillegg må det velges å legge på software enkoding for at strømmen skal kunne avspilles. Hva dette kommer av har vi ikke hatt tid til å se nærmere på. Dermed ender denne løsning opp med dobbel enkoding, noe som gir en total forsinkelse på 2150ms. I tillegg er det små tegn til hakking og noe dårligere kvalitet ved store bevegelser i bildet. Det krever en større enkodings prosess og enkode video hvor det er store forandringer i bildet fra frame til frame.

Forsinkelse: 2150ms
CPU forbruk Server ULS (gj.snitt): 30,6%
CPU forbruk VLC Klient (gj.snitt): 16,1%

6.1.4 FFMPEG – UNREAL MEDIA SERVER – VLC KLIENT

Med FFMPEG har vi full kontroll på hvilken videostrøm vi henter ut fra videokortet. Det er også enkelt å hente ut informasjon om hvilke komprimeringer som er tilgjengelige med forskjellig videostørrelse og FPS. Fordi FFMPEG er en noe mer avansert form å hente ut videoen på enn de øvrige eksemplene velger vi å gå litt mer i detalj for å forklare dette. Under følger en kort oppsummering av hvordan vi har gått frem for og streame med ffmpeg:

Terminalen åpnes og man setter «directory» til hvor man har valgt å plassere ffmpeg/bin, slik at man kan benytte seg av applikasjonen.

For å finne Picolo kortet:

```
$ ffmpeg -list_devices true -f dshow -i dummy
```

Her får man oversikt over alle inputs som er tilgjengelige via Direct Show. I alle kommandoer senere må ID til Picolo benyttes. Det vil ikke fungere og benytte seg av selve navnet. Device ID må brukes. I vårt tilfelle består ID nummeret til Picolo kortet på over 150 tegn, så vi vil heretter referere til ID som «picolo». Den virkelige ID som benyttes står under «alternative name» ved å kjøre liste over tilgjengelige source enheter.

Liste over hvilke innstillinger og strømmer som er tilgjengelige fra Picolo:

```
$ ffmpeg -f dshow -list_options true -i video=”picolo”
```

Resultatet viser at Ffmpeg kan hente ut både raw video i formatet «yuv420p» og komprimert video i H.264 format. Under er de mest aktuelle formatene for oss å hente ut listet opp.

```
[dshow @ 0000000000832940] DirectShow video device options (from video devices)
[dshow @ 0000000000832940]  Pin "Video Capture" (alternative pin name "0")
```

Raw

```
[dshow @ 0000000000832940]   pixel_format=yuv420p  min s=1920x1080 fps=25 max
s=1920x1080 fps=50
[dshow @ 0000000000832940]   pixel_format=yuv420p  min s=1920x1080 fps=29.97 max
s=1920x1080 fps=59.9402
 [dshow @ 0000000000832940]   pixel_format=yuv420p  min s=640x480 fps=29.97 max
s=640x480 fps=59.9402
[dshow @ 0000000000832940]   pixel_format=yuv420p  min s=640x480 fps=25 max
s=640x480 fps=50
 [dshow @ 0000000000832940]   pixel_format=yuv420p  min s=1280x720 fps=29.97 max
s=1280x720 fps=59.9402
[dshow @ 0000000000832940]   pixel_format=yuv420p  min s=1280x720 fps=25 max
s=1280x720 fps=50
 [dshow @ 0000000000832940]   pixel_format=yuv420p  min s=1920x1080 fps=29.97 max
s=1920x1080 fps=59.9402
[dshow @ 0000000000832940]   pixel_format=yuv420p  min s=1920x1080 fps=25 max
s=1920x1080 fps=50
 [dshow @ 0000000000832940]   pixel_format=yuv420p  min s=1920x1080 fps=29.97 max
s=1920x1080 fps=59.9402
[dshow @ 0000000000832940]   pixel_format=yuv420p  min s=1920x1080 fps=25 max
s=1920x1080 fps=50
```

Compressed H.264

```
 [dshow @ 0000000000832940]   vcodec=h264  min s=640x480 fps=29.97 max s=640x480
fps=59.9402
[dshow @ 0000000000832940]   vcodec=h264  min s=640x480 fps=25 max s=640x480 fps=50
 [dshow @ 0000000000832940]   vcodec=h264  min s=1280x720 fps=29.97 max s=1280x720
fps=29.97
[dshow @ 0000000000832940]   vcodec=h264  min s=1280x720 fps=25 max s=1280x720
fps=25
 [dshow @ 0000000000832940]   vcodec=h264  min s=1920x1080 fps=29.97 max
s=1920x1080 fps=29.97
[dshow @ 0000000000832940]   vcodec=h264  min s=1920x1080 fps=25 max s=1920x1080
fps=25
 [dshow @ 0000000000832940]   vcodec=h264  min s=1920x1080 fps=29.97 max
s=1920x1080 fps=59.9402
[dshow @ 0000000000832940]   vcodec=h264  min s=1920x1080 fps=25 max s=1920x1080
fps=50
 [dshow @ 0000000000832940]   vcodec=h264  min s=640x480 fps=29.97 max s=640x480
fps=59.9402
[dshow @ 0000000000832940]   vcodec=h264  min s=640x480 fps=25 max s=640x480 fps=50
```

Det finnes også langt flere enn disse formatene, men det er hovedsakelig valg mellom enten «raw yuv420p», komprimert H.264 og valg mellom flere forskjellige størrelser og FPS.

For å strømme fra videokortet til udp:

```
$ ffmpeg  -f dshow -pix_fmt yuv420p -re -i video="picolo" -an -r 120 -s 1920x1080 -
preset veryfast -tune zerolatency -f mpegts udp://169.254.55.99:5130
```

Det denne kommandoen forteller er at Picolo kortet skal benyttes som source, det skal hentes ut ukomprimert strøm, lyden skal fjernes fra video for mindre forsinkelse, størrelse settes i tillegg til andre innstillinger for videoen. Utgangen skal streames til ip-adressen hvor Unreal Media Server [12] ligger og den porten UMS benytter for video inn. Det har blitt prøvd mange forskjellige innstillinger og kodeker for å finne løsningen som gir minst delay. Det beste alternativet med tanke på forsinkelse her er å ta inn raw videostrøm i yuv420p format for så å gjøre dette om til mpegts format for streaming over UDP. Dette går noe utover bildekvaliteten, men vi streamer fremdeles med full oppløsning og 30FPS.

Forsinkelse: 1000ms
CPU forbruk Server FFMPEG (gj.snitt): 4,3%
CPU forbruk VLC Klient (gj.snitt): 16,2%

6.1.5 OBS – UNREAL MEDIA SERVER – VLC KLIENT

OBS står for Open Broadcaster Software og benyttes for å hente ut bilde fra videokortet. Dette er en software som er tilpasset gaming, men det er fullt mulig å benytte seg av for oss. Her går kommunikasjonen over RTMP, som hentes opp i Unreal Media Server og streames videre over RTMP til VLC hos klient. Begrensningen i OBS er at det ikke er noe alternativ å kunne velge mellom komprimert og ukomprimert strøm fra Picolo. Det er rundt 300-400ms forsinkelse på videoen inn til OBS, noe som vil si at det mest sannsynlig hentes ut en ferdig komprimert strøm fra videokortet. Det foregår da en dobbel enkoding som vil gi en forsinkelse på 1050ms. Her har ikke oppløsningen noe å si på forsinkelsen, men FPS har stor betydning. Her vil en FPS på 30 gi en forsinkelse på 2800ms, mens en FPS på 120 gir en forsinkelse på 1100ms. Dette med en bitrate på 5000 kb/s og aktivert CBR og CBR padding som gir raskest respons. Denne forsinkelsen er ikke konstant, men kan variere noe ut i fra hvor mye bevegelse det er i bildet. I tillegg har vi opplevd at et raskere intervall mellom hver I/Key frame vil gi en mindre forsinkelse. En I frame inneholder all informasjon og er ikke avhengig av andre frames for å kunne dekodes. Vi bruker denne løsningen for webkameraet hvor vi benytte et intervall på 1 sekund.

Forsinkelse: 1100ms
CPU forbruk Server OBS (gj.snitt): 43,5%
CPU forbruk VLC Klient (gj.snitt): 33%

6.2 MMI-ENHETER

Vi har integrert flere «man-machine-interface» enheter i RTO applikasjonen. Under følger en kort oppsummering av den oppfatningen vi har av hvor godt egnet disse enhetene er til å styre kameraet.

6.2.1 TOBII EYEX EYETRACKER

Vi har utviklet i hovedsak to forskjellige metoder for å kunne styre kameraposisjonen ved hjelp av eyetrackeren.

Metode 1: Det er laget rektangler med langs alle kanter og hjørner av bildet. Disse har en gjennomsiktighet på ca. 50 % slik at man skal kunne se videobildet i bakgrunnen. Når man ser på disse avgrensede områdene vil kameraet flytte seg med en konstant bevegelse i den valgte retningen inntil blikket flyttes ut av rektangelet. Ser man for eksempel i rektangelet langs øvre kant, vil kameraet bevege seg med positiv tilt opp. Det er også mulig å kombinere retninger ved å se i hjørnene. Kameraet vil da bevege seg skrått.

Metode 2: Her benyttes ikke rektangler, men piltaster for å utføre en posisjonsendring på kameraet i forhold til hvor brukeren ser. Her er det funksjonalitet for å sentrere bildet ut i fra det punktet brukeren ser på når tasten J trykkes. Man kan zoome inn og sentrere bildet på det punktet brukeren ser når tasten I (in) trykkes. Det er også mulig å zoome ut og sentrere bildet fra det punktet brukeren ser når tasten O (out) trykkes.

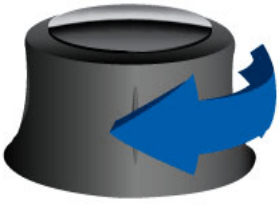
Det er også mulig å kombinere metode 1 og 2. Etter vår mening er metode 1 lite egnet til å styre kameraet fordi man er avhengig av å holde blikket fast innenfor kontrollområdet så lenge kameraet skal være i bevegelse. Dermed blir det vanskelig å følge med på selve bildet. Metode 2 fungerer i midlertid langt bedre. Dette er en relativt rask, behagelig og mer naturlig måte å styre kameraet på. Det negative med denne metoden er at det kan ta noe lang tid å flytte kameraet med store bevegelser, da man aldri kan posisjonere kameraet lenger enn til hjørnet av bildet som vises. Vi mener det ligger et potensiale i forhold til å styre kameraet med metode 2, men da helst i kombinasjon med en annen styringsenhet i tillegg.

6.2.2 3D MUS SPACE NAVIGATOR

Det er utviklet en metode for å styre kamera ved hjelp av 3D musen. Vi har valgt å bruke kun tre av de seks mulige parameterne til å styre kamera. Det ble testet litt med de forskjellige parameterne, men vi endte opp med å bruke kun tre av disse da 3D musen er veldig sensitiv og det fort kan bli vanskelig å styre den hvis alle parameterne er aktive samtidig. Hvis parameterne som ikke er tatt i bruk skulle bli interessante på et senere tidspunkt er disse lett tilgjengelig i koden.

Figur 7 – Venstre /høyre styring

Figur 8 – Opp/ned styring



Figur 9 – Zoom inn/ut

For å styre kameraets posisjon har vi valgt å bruke rotasjonsparameterne i x og y retning, disse er visualisert i figur 7 og 8. Når man da beveger musen i noen av disse retningene vil da kameraets posisjon endres i den aktuelle retningen. Hastigheten på kameraet blir styrt av hvor mye trykk man tilfører musens rotasjonsparametere hvor maks hastighet er når musen er i endeposisjon.

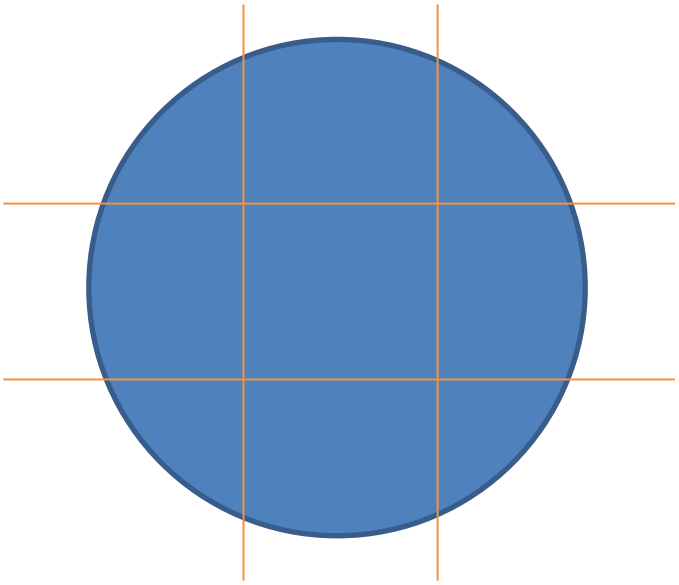
Kameraets zoom funksjon styrer vi ved å rotere musen som vist i figur 9. Her blir det zoomet inn hvis man roterer med klokken og zoomet ut hvis man roterer mot klokken. Her er også hastigheten på zoomingen regulert etter hvor mye trykk som blir tilført musen.

Vi er fornøyde med denne metoden og mener den gir oss en enkel og god styring av kameraet. Med en mindre videoforsinkelse vil også styringen oppleves som mye bedre.

6.2.3 MULTI TOUCH CONTROL WHEEL

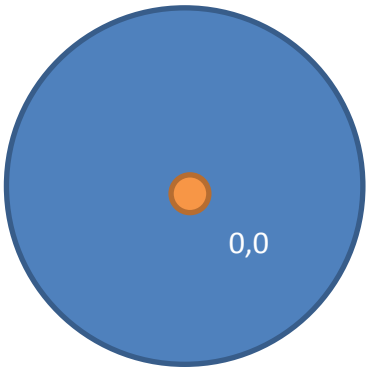
Det er blitt utviklet flere måter å styre kamera på ved hjelp av MTCW. Hvor en kombinasjon av metode 3 og zoom 3 er det som er implementert i det ferdige systemet. Det kan være noe uvant å bruke enheten til å begynne med men etter litt bruk blir man vant til den.

Metode 1: Denne metoden går ut på å dele MTCW inn i et rutenett på 3x3. Slik som det er vist på bildet under. Ved å trykke på et av feltene vist under, så kamera gå i den retningen ved en bestemt fart. Ved å trykke ruten i midten så vil kamera stoppe.



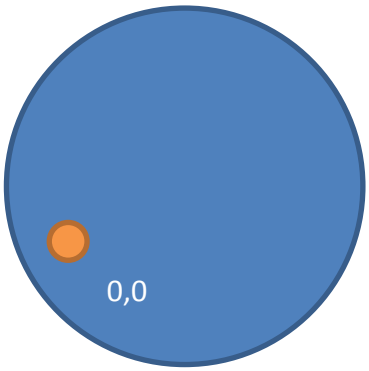
Figur 10 - Continuous move for MTCW

Metode 2: Det er også blitt utviklet en funksjon som tar utgangspunkt i origo. Ved å bruke en finger å styre den rundt på flaten, så vil kamera prøve å følge samme bevegelse. Ved å bruke denne metoden kan du også styre farten på kamera, ved å flytte fingeren rask/sakte.



Figur 11 - Absolutt styring

Metode 3: Denne metoden tar utgangspunkt i hvor du trykker. Deretter så fungerer den på samme måte som metode 2. Det er denne metoden vi har gått for i systemet.

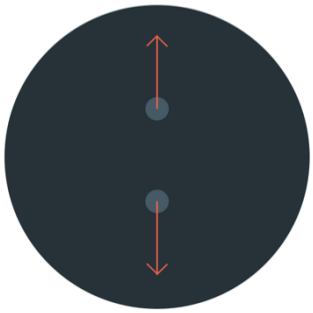


Figur 12 - Relativ styring

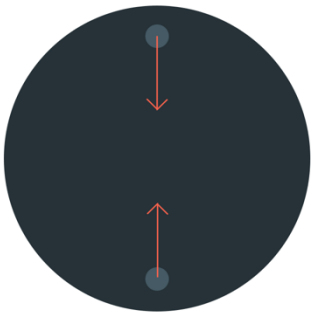
Zoom 1: Vi har sett på muligheten ved å bruke enkoder ringen som zoom. Dette fungerer veldig bra, så lenge styringen blir deaktivert. Problemet som oppstod da vi kombinerte denne løsningen med en av de andre metodene, var at de overlappet de hverandre og resultatet endte som oftest med at vi styrte kamera. Selv om vi prøvde å zoome.

Zoom 2: Vi har sett også sett på en løsning hvor man brukte 3 fingre for å aktivere/deaktivere zoom funksjonalitet. Men det var ikke alltid de tre fingrene ble registrert av MTCW.

Zoom 3: Den siste metoden vi har brukt for å zoome er å bruke to fingre. Denne metoden har sin opprinnelse fra mobilen.



Figur 13 - Zoom in, MTCW



Figur 14 - Zoom ut, MTCW

VEDLEGG

HISTORIKK

Her er en oversikt over historikk som er gjort før versjon 1.0

Tabell 4 – Historikk

Versjon	Dato	Skrevet av	Beskrivelse
0.1	26.02.2016	Alexander Schinnes	Første utkast
0.2	01.03.2016	Svein Bekkevold	Lagt til kap 4
0.3	05.03.2016	Svein Bekkevold	Lagt til kamera

REFERANSER

[1] - Delays in Axis IP Surveillance Cameras:
http://lup.lub.lu.se/luur/download?func=downloadFile&recordOId=3971650&fileOId=3973011 (26.02.2016)
[2] - Latency in live network video surveillance:
http://www.axis.com/files/whitepaper/wp_latency_live_netvid_63380_external_en_1504_lo.pdf (26.02.2016)
[3] – Master's Thesis. Cross-Platform Video Management Solutions:
http://lup.lub.lu.se/luur/download?func=downloadFile&recordOId=7373677&fileOId=7373681 (26.02.2016)
[4] – PC1842 – Picolo HD H.264 Handbook v. 3.2
[5] – Picolet.net HD4 Encoder datasheet
[6] – Picolo.net HD4 Encoder handbook
[7] – Info om InterCOM DDS:
http://www.kongsberggeospatial.com
(01.03.2016)
[8] – Info om Jitter:
http://www.wikipedia.org/wiki/Klokke_(elektronikk)
(09.03.2016)
[9] – TCP, UDP og SCTP
http://www.wikipedia.org/wiki/UDP
(01.03.2016)
[10] – Qt
https://www.qt.io (03.05.2016)
[11] – FFMPEG
https://ffmpeg.org (03.05.2016)
[12] – Unreal Media Server
http://www.umediaserver.net (03.05.2016)

Høgskolen i Sørøst-Norge

KONGSBERG

GRUPPE 2

RISIKODOKUMENT – v 2.0

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENT 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIKK 4
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 5
2.4 FIGURER OG BILDER 5
3 BAKGRUNN 5
4 RISIKOVURDERING 6
5 RISIKOANALYSE 7
5.1 PROSJEKTRISIKO 7
5.2 SYSTEMRISIKO 9
6 RISIKO SOM HAR INTRUFFET 10
6.1 PROSJEKTRISIKO 10
6.2 SYSTEMRISIKO 11
7 IKKE-GJELDENDE RISIKO 11
7.1 PROSJEKTRISIKO 11
7.2 SYSTEMRISIKO 12
8 KVALITETSSIKRING 13
8.1 ORGANISERING AV KVALITETSSIKRING 13
VEDLEGG 13
TIDLIGERE DOKUMENTHISTORIKK 13
REFERANSER 13

1 DOKUMENT

Dette dokumentet omfatter en risikovurdering for hele prosjektet. All risikoanalyse som er utført i oppstartsfasen vil være dokumentert her, i tillegg til at dokumentet vil oppdateres jevnlig med tiltak som blir utført i tilfelle eventuelle risikoer skulle slå til.

1.1 OVERSIKT

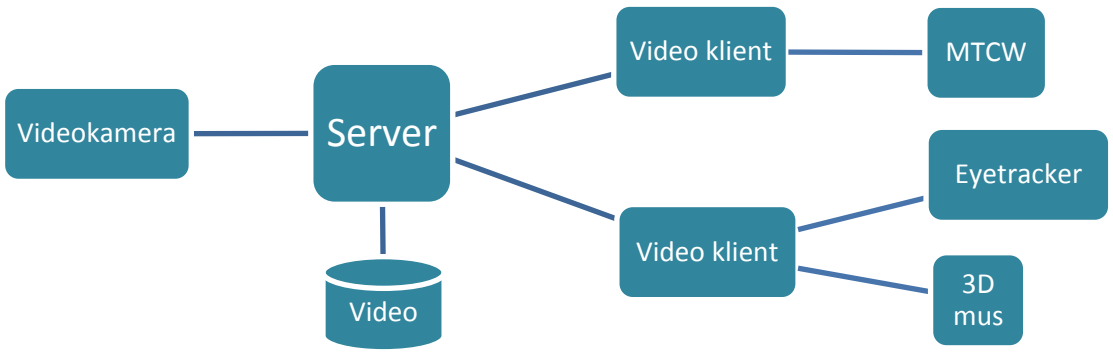
Hensikten med dette dokumentet er å kartlegge alt det som kan gå galt og hvordan man kan forebygge det under hele prosjektperioden.

Hvis man får kartlagt det som kan gå galt, kan man lage en plan for å redusere risikoen, samt forebygge risikoen og få løst problemet så fort som mulig. Et godt laget risikodokument kan forhindre uheldige situasjoner og negative ting som kan oppstå og vil hjelpe oss med å levere et bedre produkt.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 – Systemoversikt

1.3 HISTORIKK

Se vedlegg for tidligere dokumenthistorikk.

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
2.0	19.05	Vegard	Gjennomgang, oppdatert «ikke gjeldene-risiko»

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
MTCW	Multi Touch Control Wheel
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.
Greyhill	Firma som produserer og driver support på MTCW.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 4
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4
Tabell 4 - Risikopåvirkning 6
Tabell 5 - Risikomatrise 6
Tabell 6 - Risikofaktor 7
Tabell 7 - Prosjektrisiko 7
Tabell 8 - Systemrisiko 9
Tabell 9 - Inntruffede prosjektrisiko 10
Tabell 10 - Inntruffede systemrisiko 11
Tabell 11 - Ikke-gjeldende prosjektrisiko 11
Tabell 12 - Ikke-gjeldende systemrisiko 12
Tabell 13 - Dokumenthistorikk versjon 0 13

2.4 FIGURER OG BILDER

Figur 1 – Systemoversikt 3
Figur 2 - Risikoplot 10

3 BAKGRUNN

Det vil alltid oppstå uforutsette problemer, hvor det er vanskelig å vite hva den beste beslutningen for prosjektet er. Ved å definere mål og målsetninger tidlig i prosjektet, vil det hjelpe oss å redusere risikoen for å ta feil beslutninger. Da vi startet dette prosjektet diskuterte vi hvilke mål vi måtte oppnå, for å kunne si at systemet er en suksess. Dermed kan vi legge en plan for hvordan vi skal handle, når enkelte faremomenter skulle oppstå. Om prosjektet blir en suksess eller ikke kan i noen tilfeller være helt avhengig av hvordan vi har planlagt for både forutsatte og uforutsatte problemer og situasjoner. Ved å legge ned en god innsats i risikoanalysen og tenke nøye igjennom ulike situasjoner som kan oppstå, kan vi spare mye tid senere i prosjektet. Da er vi godt forberedt i tilfelle disse situasjonene skulle oppstå. Tiltakene som da skal utføres er godt gjennomtenkt, slik at de kan handles etter og gi en god pekepinn på hvilke konsekvenser den eventuelle feilen får.

4 RISIKOVURDERING

Vi har valgt å presentere risikoanalysen i tabeller og matriser, grunnet bedre oversiktlighet. Denne type analyse hjelper oss å forutse hendelser som kan oppstå under prosjektet. Risiko består av to hovedfaktorer.

1. Det negative utfallet av en prosjektaktivitet.
2. Sannsynligheten for at utfallet skal forekomme.

Dette kan skrives som en matematisk formel:

$$Risiko \ = \ innvirkning \ x \ sannsynlighet$$

Videre blir det drøftet eksempler.

Tabell 4 - Risikopåvirkning

	Påvirkning på prosjektet	
	Sannsynlighet	**Innvirkning**
Svært liten	< enn en hendelse pr. 1000 time	Prosjektet vil kunne gå videre uten vansker.
Liten	Ca. en hendelse pr. 1000 time	Problematisk, løses kontrollert av gruppen.
Middels	Ca. en hendelse pr. 100 time	Prosjektet stagnerer, tiltak må vurderes.
Stor	Ca. en hendelse pr. 10 time	Prosjektet stopper opp, tiltak er nødvendig.
Svært stor	> enn en hendelse pr. 10 time	Kritisk. Handling må skje umiddelbart.

Tabell 5 - Risikomatrise

		Innvirkning				
		1 – Svært liten	2 – Liten	3 – Middels	4 - Stor	5 -Svært Stor
Sannsynlighet	1 – Svært liten	1	2	3	4	5
	2 – Liten	2	4	6	8	10
	3 – Middels	3	6	9	12	15
	4 – Stor	4	8	12	16	20
	5 – Svært stor	5	10	15	20	25

Tabellen over viser risikoen i forhold til sannsynlighet og innvirkning. Denne brukes til å hjelpe oss til å beregne på visse risikoscenarioer, ved å bruke formelen som er gitt over. Ved å regne på de forskjellige scenariene kan vi plassere dem i en risikokategori. Igjen vil dette hjelpe oss i å handle riktig når situasjonen oppstår.

Tabell 6 - Risikofaktor

Risikokategori	Rang nr.	Rang	Forklaring
15-25	1	Kritisk	Øyeblikkelig tiltak.
10-12	2	Høy	Uakseptabel risiko. Tiltak nødvendig.
5-9	3	Medium	Tiltak vurderes
2-4	4	Lav	Akseptabel risiko. Ingen tiltak.
1	5	Veldig lav	Risiko minimal. Ingen tiltak.

En risiko kan reduseres ved å:

- Redusere sannsynligheten: - Kartlegge mulige risikoer, analysere disse og deretter lage en forebyggingsplan.
- Redusere konsekvensen: - Lage en plan som inneholder tiltak for å redusere faren til en risiko.

5 RISIKOANALYSE

I dette prosjektet har vi valgt å dele opp risikoene i to forskjellige kategorier. Disse har vi kalt prosjektrisiko og systemrisiko. Ved å skille disse to ulike risikotypene vil vi ha en god oversikt over risikoene som gjelder det administrative rundt prosjektet og risikoene som gjelder systemet eller produktet som vil følge det gjennom hele dets levetid. Dette er viktig for kunden å ha, slik at de vet om eventuelle risikoer som kan oppstå, og hvordan disse risikoene skal håndteres.

5.1 PROSJEKTRISIKO

I tabell 0-5 er det utført en analyse av alle risikoer som kan oppstå gjennom prosjektfasen. Ved å ta i bruk de forskjellige diagrammene som vi har nevnt i kapittel 3, og se for oss ulike scenarioer, så kan vi se hvordan dette vil ramme gruppens samarbeid eller prosjektet. Her har vi brukt formelen gitt i kapittel 3. Hvor S = sannsynlighet, I = innvirkning og R = Risiko.

Tabell 7 - Prosjektrisiko

ID nr.	Scenario	Årsak	S	I	R	Tiltak
1.01	Tap av data	Pc-svikt, menneskelig feil.	1	3	3	Sikkerhetskopi: lokalt, ekstern harddisk og onlinelagring på Google Drive. Skal utføres ukentlig backup.
1.02	Lett sykdom	Flere	3	2	6	Melder om fravær og tar vitaminpiller. Øvrige medlemmer utfører den sykes arbeid etter behov.
1.03	Alvorlig sykdom eller død	Flere	1	4	4	Revurder prosjektplan og målsetninger.

ID nr.	Scenario	Årsak	S	I	R	Tiltak
1.04	Uenigheter	Dårlig kommunikasjon, stress, eksterne faktorer.	4	2	8	Forbedre kommunikasjon. Ta ting opp, avstemning i demokrati.
1.05	Uløste utfordringer	Innaktivitet, manglende kunnskap	3	3	9	Scrum møter, hjelp av gruppen.
1.06	Ligge bak tidsplanen	Arbeid avviker fra prosjektplanen.	2	3	6	Analyse av situasjon, oppdatere/gjennomgang av prosjektplan.
1.07	Inaktivet	Manglende motivasjon, forsinkelser, latskap.	1	4	4	Dialog med vedkommende. Fokus på godt samarbeid i gruppen.
1.08	Trekk av oppdragsgiver	Brudd på kontrakt	1	5	5	Jevnlige møter med oppdragsgiver. Innretting etter beskjeder.
1.09	Personlig forsinkelser	Transportproblemer, dårlig vær, forsovelser	3	1	3	Ta alltid hensyn til uforutsette problemer ved å planlegge god tid.
1.10	Står fast i oppgaven	Manglende kunnskap	3	3	9	Benytt eksterne og interne ressurser for å få innspill.
1.11	Manglende ekstern oppfølgning	Nedprioritering av studentoppgaven	2	3	6	Snakk med oppdragsgiver. Følg oppgaven så langt den er beskrevet og forstått.
1.12	Manglende intern oppfølgning	Sykdom, dårlig kommunikasjon	2	2	4	Forbedre kommunikasjon, benytt evt. annen intern kompetanse.
1.13	Fravær	Arbeid, jobbintervju, dårlig prioritering	4	2	8	Gi oppgaver som kan jobbes med slik at tapt tid hentes inn utenom skoletid.
1.14	Intern sabotasje	Medlemmer motarbeider gruppen	1	5	5	Diskuteres innad i gruppen med intern veileder til stede, nødvendige tiltak utføres.
1.15	Fysiske katastrofer	Naturkatastrofer, vannlekkasjer, brann	1	4	4	Utstyr må erstattes slik at arbeidet kan fortsette. Avtal med oppdragsgiver.

I tabellen over så ser vi at ingen av de drøftede scenarioene legger seg i den «Høy» og «Kritiske» kategorien. Det betyr imidlertid ikke at slike scenarioer ikke kan oppstå, så det er viktig at alle gruppemedlemmer er lett tilgjengelig, slik at eventuelle uforutsette situasjoner blir diskutert og tiltak blir utført deretter.

5.2 SYSTEMRISIKO

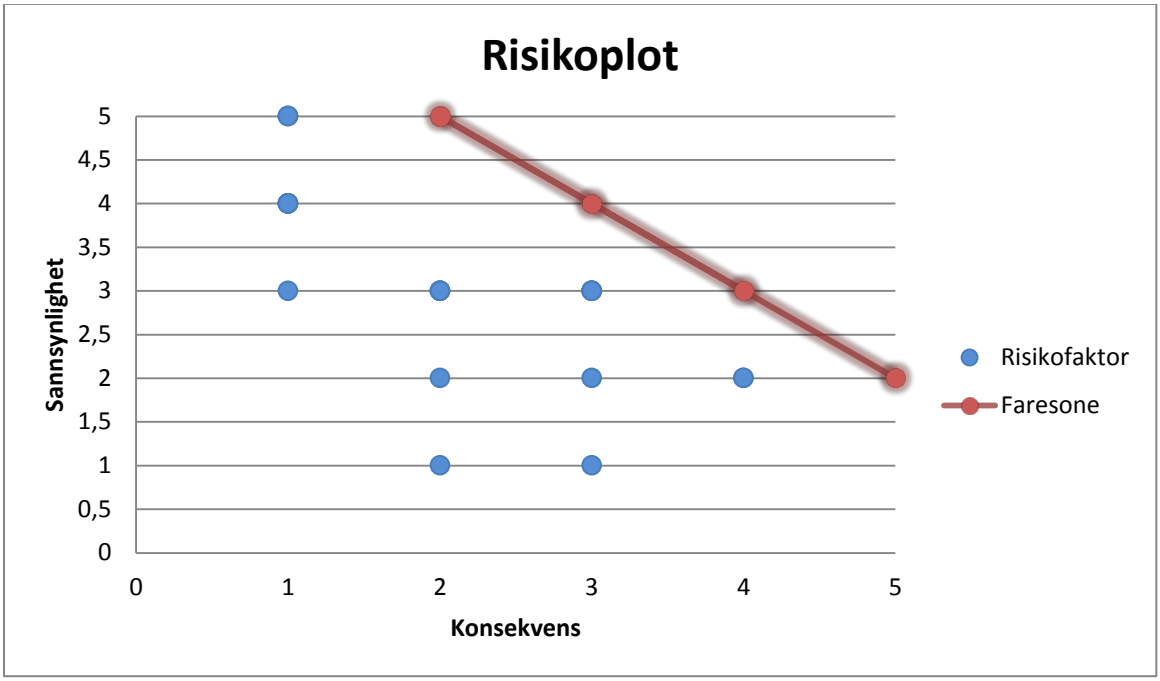
Dette er risikokategorien som omhandler situasjoner som kan oppstå i forbindelse med system og produkt. For at vi skal kunne produsere det best tenkelige systemet for oppdragsgiver, så er det viktig å se på noen tekniske scenarioer. Her er kommunikasjon mellom MMI – enhetene og videokameraet, «Software brukervennlighet»(API til hver av enhetene), driver til MTCW og en liten database er faktorer som er betydningsfulle.

Tabell 8 - Systemrisiko

ID nr.	Scenario	Årsak	S	I	R	Tiltak
2.01	Begrenset utviklingsverktøy	Qt Quick støtter ikke den funksjonaliteten vi trenger og ønsker.	2	3	6	Må avtales med oppdragsgiver om funksjonalitet skal fjernes, eller om alternativt utviklingsverktøy skal tas i bruk.
2.02	Begrenset språk for programmering	C++ og QML fungerer ikke som hovedspråk for våre behov.	1	4	4	Må gjøres research på andre programmeringsspråk og avtales med oppdragsgiver ved eventuell overgang til annet språk.
2.03	Stor forsinkelse på bildestrøm	For treg maskinvare eller ikke-optimalisert software.	3	3	9	Kode bør optimaliseres og feilkilder for software og hardware må dokumenteres og begrunnes.
2.04	Manglende kommunikasjon mellom klient og server	Feil eller manglende støtte i software eller begrenset kompetanse.	1	5	5	Gjør research på kommunikasjon og hør med tilgjengelig teknisk personell fra oppdragsgiver for hjelp.
2.05	Utstyr blir ødelagt	Dårlig kvalitet, ulykker eller menneskelig feil.	1	4	4	Hør med oppdragsgiver om utstyr kan erstattes, hvis ikke må årsak, konsekvens og tiltak dokumenteres.
2.06	Lite lagringskapasitet for HD-video	For lite plass på tilgjengelige harddisker.	2	1	2	Videolengden må forkortes, lagret video må slettes etter opptak. Evt. be oppdragsgiver om større harddisk ved stort behov.
2.07	Kamera kan ikke styres	Begrenset API eller manglende støtte for noe hardware eller software.	2	3	6	Funksjonalitet for systemet må endres og dokumenteres. Evt. avtal med oppdragsgiver om støtte for alternativ styring.
2.08	MTCW kan ikke brukes	Manglende driver	3	3	9	Få tak i driver fra Greyhill som kan brukes som utgangspunkt for å lage egen driver.

I tabellen over så ser vi at ingen av de drøftede scenarioene legger seg i den «Høy» og «Kritiske» kategorien. Det betyr imidlertid ikke at slike scenarioer ikke kan oppstå, så det er viktig at alle gruppemedlemmer er lett tilgjengelig, slik at eventuelle uforutsette situasjoner blir diskutert og tiltak blir utført deretter.

Vi har også valgt å lage et Scatter-diagram som viser risikoene plottet grafisk. Dette gjør vi for å finne ut hvor sannsynlig det er at problemer oppstår og hvor farlig det er for prosjektarbeidet. Risikoplottet er satt opp med sannsynlighet på x-aksen og konsekvens på y-aksen. Risikopunkter er merket med blå punkter. Den røde linjen, og alt ovenfor denne, viser faresonen. Hvis en risikofaktor havner på, eller over denne linjen, må det gjøres tiltak. Da er risikofaktoren 10 eller høyere. Men her finnes det ingen blå punkter på farelinjen eller i sonen over. Dermed krever ingen risikoer øyeblikkelige tiltak, men alle vil selvfølgelig vurderes.



Figur 2 - Risikoplot

6 RISIKO SOM HAR INTRUFFET

6.1 PROSJEKTRISIKO

Følgende risikoer som er listet i tabell 9 har intruffet en eller flere ganger i løpet av prosjektet.

Tabell 9 - Inntruffede prosjektrisiko

ID nr.	Scenario	Kommentar
1.13	Fravær	Medlemmer har vært på privat arbeid i skoletiden noen enkeltdager samt noen jobbintervju. Oppgaver blir delt ut slik at tiden hentes inn igjen utenom skoletid. Tidsplan overholdes.

1.02	Lett sykdom	I forbindelse med 2. presentasjon ble det fravær på grunn av sykdom. Carl Peder var syk og hadde mistet stemmen. Etter avtale med gruppen benyttet han derfor dagen hos legen istedenfor på skolen. Øvrige gruppemedlemmer overtok hans del og løste utfordringen på en god måte.

6.2 SYSTEMRISIKO

Tabell 10 - Inntruffede systemrisiko

ID nr.	Scenario	Kommentar
2.03	Stor forsinkelse på bildestrøm	Vi ser allerede at vi har stor forsinkelse på bildestrøm. Det er mye tid som går med til å komprimere strømmen, det går også mye tid i software. Sammen med kunde har vi konkludert med at nåværende hardware løsning ikke er den mest gunstige. Vi fortsetter likevel å bruke denne da det er kundens ønske.

7 IKKE-GJELDENDE RISIKO

7.1 PROSJEKTRISIKO

Tabell 11 gir en oversikt over risikoer som har blitt analysert, men ikke inntruffet. De er derfor oppsummert ved prosjektslutt som ikke-gjeldende.

Tabell 11 - Ikke-gjeldende prosjektrisiko

ID nr.	Scenario	Kommentar
1.01	Tap av data	Det ble tatt jevnlig backup av alle dokumenter og systemet.
1.03	Alvorlig sykdom eller død	Ingen døde eller fikk alvorlig sykdom
1.04	Uenigheter	Det ble diskutert og løst på en diplomatisk måte
1.05	Uløste utfordringer	Det ble holdt Scrum møter hver dag, samt gruppen hjalp hverandre
1.06	Ligge bak tidsplanen	Det var god oppfølging innad i gruppen
1.07	Inaktivet	Motivasjonen er alltid på topp
1.08	Trekk av oppdragsgiver	Hadde jevnlige møter
1.09	Personlig forsinkelser	Alle jobbet/var tilgjengelig fra 09:00 – 15:00.
1.10	Står fast i oppgaven	Benyttet ekstern veileder for å komme videre
1.11	Manglende ekstern oppfølgning	De møtte alltid opp til avtalt tid/møter
1.12	Manglende intern oppfølgning	Var tilgjengelig på kontoret når vi trengte det. Møtte opp til avtalt tid møter
1.14	Intern sabotasje	Ingen medlemmer motarbeidet gruppen
1.15	Fysiske katastrofer	Ingen fysiske katastrofer har oppstått.

7.2 SYSTEMRISIKO

Tabell 12 gir en oversikt over risikoer som har blitt analysert, men ikke inntruffet. De er derfor oppsummert ved prosjektslutt som ikke-gjeldende.

Tabell 12 - Ikke-gjeldende systemrisiko

ID nr.	Scenario	Kommentar
2.01	Begrenset utviklingsverktøy	Brukte 3'parts programmer / open-source biblioteker for å gi ønsket funksjonalitet
2.02	Begrenset språk for programmering	C++ og QML hadde alt vi trengte av funksjonalitet
2.04	Manglende kommunikasjon mellom klient og server	Laget egen server og klient, slik at vi har full kontroll.
2.05	Utstyr blir ødelagt	Vi har tatt vare på utstyr vi har fått utlevert
2.06	Lite lagringskapasitet for HD-video	Vi har masse plass for lagring
2.07	Kamera kan ikke styres	Det ble kjøpt inn et Axis V5915 som hadde innebygget motor, samt en API.
2.08	MTCW kan ikke brukes	Det ble laget en driver

8 KVALITETSSIKRING

Å tenke gjennom de forskjellige scenarioene gjør at vi kan unngå de mest kritiske, og fortsatt ha et system som oppfyller kravene fra oppdragsgiver slik at systemet kan bli en suksess.

8.1 ORGANISERING AV KVALITETSSIKRING

Kvalitetssikring foregår internt i gruppen. Alle krav har tester som må være godkjent før vi kan konkludere med at oppdragsgivers krav til systemet er oppfylt. I denne prosessen er det viktig å sørge for god kommunikasjon med ekstern veileder slik at oppdragsgivers og prosjektgruppens visjon av systemet er samstemte.

Ansvarlig for kvalitetssikring: Carl Peder Jansen

Deltakere: Vegard Smedsvik, Alexander Schinnes, Svein Bekkevold og Carl Peder Jansen.

Ekstern veileder: Arnt-Levi Alstad, Kongsberg Defence & Aerospace

VEDLEGG

TIDLIGERE DOKUMENTHISTORIKK

Tabell 13 - Dokumenthistorikk versjon 0

Versjon	Dato	Skrevet av	Beskrivelse
0.1	31.01.2016	Alexander Schinnes	Første utkast
0.2	01.02.2016	Alexander Schinnes	Oppdatert kapittel 5, risikoer som har intruffet. ID. nr 1.13
1.0	09.03.2016	Carl Peder Jansen	Dokumentgjennomgang før andre presentasjon
1.1	06.04.2016	Alexander Schinnes	Oppdatert intruffet risiko 1.02

REFERANSER

[1] J.Blaalid, Risikohåndtering av IT-prosjekter, versjon 1998.
[2] T.Strøm*, O.H. Graven*, Prosjekthåndbok, Høgskolen i Buskerud, avdeling for Teknologi, versjon 2016.
[3] Prosjektplan – HBV – Kongsberg Maritime, Tidligere bacheloroppgave

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

KRAVSPESIFIKASJON – v 2.0

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENT 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIKK 4
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 OPPSETT 5
3.1 KRAVKATEGORIER 5
3.2 PRIORITET 5
3.3 EKSEMPEL PÅ KRAV ID 5
4 OVERSIKT OVER KRAV 6
5 KRAV 8
5.1 K01 – SYSTEM OG SERVER 8
5.2 K02 - KLIENT 9
5.3 K03 – GUI 10
5.5 K05 – 3D MUS SPACE NAVIGATOR 13
5.6 K06 – MULTI TOUCH CONTROL WHEEL 14
5.7 K07 - VIDEOKAMERA 14
VEDLEGG 17
TIDLIGERE DOKUMENTHISTORIKK 17
REFERANSER 19

1 DOKUMENT

Dette dokumentet omfatter en kravspesifikasjon for systemet Real Time Observation.

1.1 OVERSIKT

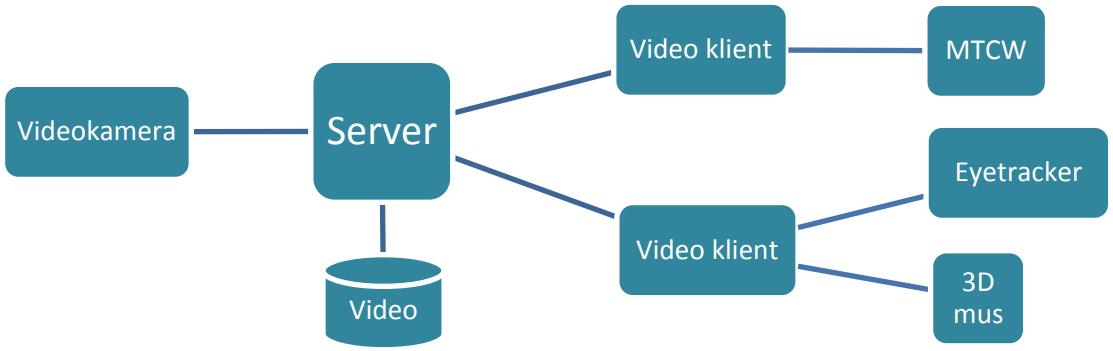
Dette dokumentet inneholder alle krav som er gitt av oppdragsgiver, hvilken prioritering hvert enkelt krav og dets underkrav har, samt en mer detaljert beskrivelse av kravene.

Kravspesifikasjonen er en beskrivelse av hvilke brukerfunksjoner som skal være i systemet. Denne blir utarbeidet før utstyr kjøpes inn og systemet utvikles for å sikre at kundens behov blir dekket når det gjelder funksjonalitet, ytelse og brukervennlighet. Kravspesifikasjonen skal beskrive hva vi skal ha med i systemet, ikke hvordan dette skal løses. Kravene danner grunnlaget for videre utvikling av systemet.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 – Systemoversikt

1.3 HISTORIKK

Se vedlegg for tidligere detaljert dokumenthistorikk.

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Dato
2.0	Korrekturlesning og retting. Fjernet statusfelt.	Alexander	19.05.2016

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 4
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4
Tabell 4 - Dokumenthistorikk versjon 0-1.1.1 17

2.4 FIGURER OG BILDER

Figur 1 – Systemoversikt 3

3 OPPSETT

3.1 KRAVKATEGORIER

Vi har delt inn kravene i følgende kategorier:

K01 - System og server (med videokort)
K02 - Klient
K03 - GUI
K04 - Eyetracker
K05 - 3D Mus Space Navigator
K06 - Multi Touch Control Wheel
K07 - Videokamera

3.2 PRIORITET

Kravene er delt inn i prioritet, slik at vi alltid vet hvilke krav som er viktigst å utføre underveis i prosjektet. Dette vil også gi kunden info om hvilke funksjoner og deler av systemet som vi arbeider med først. Vi har delt inn prioritetene i A, B og C krav.

A krav er svært viktig
B krav er viktig
C krav er ønskelig

3.3 EKSEMPEL PÅ KRAV ID

Under er et eksempel på hvordan kravene er satt opp. De er nummerert i forhold til hovedområde, krav og eventuelle underkrav.

Eksempel: K01.001A.01A

K01 betyr at dette er et krav innenfor område «system og server».
001A er hovedkravnummer med prioritet innenfor valgt område.
01A er underkrav med prioritet.

4 OVERSIKT OVER KRAV

Krav oversikt				
Krav ID	Underkrav	Navn	Use Case ID	Test ID
K01.001A	01A 02A 03A 04A 05C	Video strøm	UC01.001 UC02.001 UC02.002	T01.001
K01.002A	01A 02C 03A 04C 05C	Lagring	UC01.001 UC02.001 UC02.002 UC03.001	T01.002
K01.003A		Videokort	UC01.001 UC02.001 UC02.002 UC03.001	T01.003
K01.004A	01C	Server	UC01.001 UC02.001 UC02.002 UC03.001	T01.004
K01.005A		Analyse av videostrøm	UC03.001	T01.005
K02.001A	01A 02A 03A 04A 05B 06A	Kommunikasjon	UC01.001 UC02.001 UC02.002 UC03.001 UC04.001 UC05.001	T02.001
K03.001A	01A 02A 03C 04C 05A 08C 07C 10A 09A 12A 11A 13B	Meny	UC01.001 UC02.001 UC02.002 UC03.001 UC04.001 UC05.001	T03.001
K03.002B	01A 02A	Play/Pause	UC03.001	T03.002
K03.003B	01B 02C	Tidslinje	UC03.001	T03.003
K03.004C		Spole	UC03.001	T03.004
K03.005A		Live	UC01.001 UC03.001	T03.005
K03.006C	01A 02B 03A	Tags	UC01.001 UC03.001	T03.006
K03.007C	01A 02A 03B	Skjermdump	UC01.001 UC03.001	T03.007
K03.010A		Integrering	UC01.001 UC02.001 UC02.002 UC03.001 UC04.001 UC05.001	
K04.001A		Deteksjon	UC01.001 UC02.001	T04.001

			UC02.002 UC03.001 UC04.001 UC05.001	
K05.002A		Posisjon	UC01.001 UC02.001 UC02.002 UC03.001 UC04.001 UC05.001	T05.002
K06.001A		Driver	UC01.001 UC02.001 UC02.002 UC03.001 UC04.001 UC05.001	T06.001
K06.007A	01A 02A 03A 04A	Grensesnitt	UC01.001 UC02.001 UC02.002 UC03.001 UC04.001 UC05.001	
K07.001A		Oppløsning	UC01.001 UC02.001 UC04.001	T07.001
K07.002A	01A 02A	Bildefrekvens	UC01.001 UC02.001	T07.002
K07.003B		Nattmodus	UC02.002 UC04.001	T07.003
K07.004C		Infrarød modus	UC04.001	T07.004
K07.005A	01A 02A	Motorisert	UC02.002 UC05.001	T07.005
K07.006A		API	UC02.002 UC05.001	T07.006
K07.007A		Utganger	UC01.001 UC02.001 UC02.002 UC04.001 UC05.001	T07.007

5 KRAV

5.1 K01 – SYSTEM OG SERVER

K01 - System og server			
Krav ID	**K01.001A**		
Navn	Video strøm		
Beskrivelse	Skal kunne motta video fra videokort og videresende strømmen til klient.		
Underkrav	Krav	Beskrivelse	Test ID
	01A	Skal ha maksimal forsinkelse på 200 ms fra handling oppstår til den vises i klientens videospiller.	T01.001.01
	02A	Skal kunne sende video til 3 klienter.	T01.001.02
	03A	Skal sende komprimert H.264 strøm til lagring, se krav K01.002C.	T01.001.03
	04A	Skal kunne gjøre krav 02A og 03A samtidig.	T01.001.04
	05C	Skal kunne stå på i 24 timer i døgnet i 7 dager i strekk.	T01.004.01
Kryssreferanser			
Use Case ID	UC01.001, UC02.001, UC02.002		
Test ID	T01.001		
Informasjon			
Opphav til krav	Presentasjon bacheloroppgave og oppgavetekst.		
Kommentar			

K01 - System og server			
Krav ID	**K01.002A**		
Navn	Lagring		
Beskrivelse	Skal kunne lagre/hente opp alle videoene, tags (ref. K03.006C), og skjermdumper (ref. K03.007C)		
Underkrav	Krav	Beskrivelse	Test ID
	01A	Skal lagre all video som blir tatt opp.	T01.002.01
	02C	Skal lagre alle tags som blir opprettet.	T01.002.02
	03A	Skal kunne hente opp video.	T01.002.03
	04C	Skal kunne hente opp tags.	T01.002.04
	05C	Skal kunne hente opp skjermdumper.	T01.002.05
Kryssreferanser			
Use Case ID	UC01.001, UC02.001, UC02.002, UC03.001		
Test ID	T01.002		
Informasjon			
Opphav til krav	Presentasjon bacheloroppgave og oppgavetekst, hvorav tags og skjermdumper er et forslag fra studentgruppen.		
Kommentar			

K01 - System og server	
Krav ID	**K01.003A**
Navn	Videokort
Beskrivelse	Skal kunne motta video fra videokamera på server.
Kryssreferanser	
Use Case ID	UC01.001, UC02.001, UC02.002, UC03.001
Test ID	T01.003
Informasjon	
Opphav til krav	Presentasjon bacheloroppgave og oppgavetekst.
Kommentar	

K01 - System og server	
Krav ID	**K01.005A**
Navn	Analyse av videostrøm
Beskrivelse	Det skal utføres analyse av hvordan forandring av FPS og oppløsning påvirker videostrømmens forsinkelse.
Kryssreferanser	
Use Case ID	UC03.001
Test ID	T01.005
Informasjon	
Opphav til krav	Møte med Arnt-Levi (04.02.2016, Krona)
Kommentar	

5.2 K02 - KLIENT

K02 - Klient			
Krav ID	**K02.001A**		
Navn	Kommunikasjon		
Beskrivelse	Alle klienter skal ha kommunikasjon og dataoverføring fra server.		
Underkrav	Krav	Beskrivelse	Test ID
	01A	Skal kunne koble seg opp mot server.	T02.001.01
	02A	Skal kunne motta video fra server.	T02.001.02
	03A	Skal kunne videresende video til videoavspiller.	T02.001.03
	04A	Skal kunne styre kameraposisjon.	T02.001.04
	05B	Skal kunne velge videostrøm fra server.	T02.001.05
	06A	Skal kunne velge å ta opptak av videostrøm.	T02.001.06
Kryssreferanser			
Use Case ID	UC01.001, UC02.001, UC02.002, UC03.001, UC04.001, UC05.001		
Test ID	T02.001		
Informasjon			
Opphav til krav	Presentasjon bacheloroppgave og oppgavetekst.		
Kommentar			

5.3 K03 – GUI

K03 - GUI			
Krav ID	**K03.001A**		
Navn	Meny		
Beskrivelse	Skal kunne gi brukeren valgmuligheter som oppgitt i underkrav. Qt Quick skal brukes til all GUI for applikasjonen.		
Underkrav	Krav	Beskrivelse	Test ID
	01A	Skal kunne velge å se forsinkelse.	T03.001.01
	02A	Skal kunne velge å se bilder i sekundet (FPS).	T03.001.02
	03C	Skal kunne velge å vise «packet loss».	T03.001.03
	04C	Skal kunne velge å se datatrafikk.	T03.001.04
	05B	Skal kunne velge å se video i H.264 format.	T03.001.05
	07C	Skal kunne velge å se CPU forbruk.	T03.001.07
	08C	Skal kunne velge å se RAM forbruk.	T03.001.08
	09A	Skal kunne velge å vise video fra kamera uten effekter.	T03.001.09
	10C	Skal kunne velge å vise video fra kamera med IR.	T03.001.10
	11A	Skal kunne velge å vise video fra kamera med nattmodus	T03.001.11
	12A	Skal kunne velge å vise et «overlay» med all tilgjengelig informasjon.	T03.001.12
	13B	Skal kunne velge FPS på videostrømmen.	T03.001.13
Kryssreferanser			
Use Case ID	UC01.001, UC02.001, UC02.002, UC03.001, UC04.001, UC05.001		
Test ID	T03.001		
Informasjon			
Opphav til krav	Presentasjon bacheloroppgave og oppgavetekst.		
Kommentar			

K03 - GUI			
Krav ID	**K03.002B**		
Navn	Play/Pause		
Beskrivelse	Skal kunne pause og spille av video. Qt Quick skal brukes til all GUI for applikasjonen.		
Underkrav	Krav	Beskrivelse	Test ID
	01A	Skal kunne spille av video fra gitt punkt.	T03.002.01
	02A	Skal kunne pause video.	T03.002.02
Kryssreferanser			
Use Case ID	UC03.001		
Test ID	T03.002		
Informasjon			
Opphav til krav	Presentasjon bacheloroppgave og oppgavetekst.		
Kommentar			

K03 - GUI			
Krav ID	**K03.003B**		
Navn	Tidslinje		
Beskrivelse	Skal kunne vise hvor lenge du har sett på en video på en tidslinje, hvor 0 er starten på videoen og slutten viser lengden på videoen. Qt Quick skal brukes til all GUI for applikasjonen.		
Underkrav	Krav	Beskrivelse	Test ID
	01B	Skal kunne trykke på tidslinjen og hoppe til gitt tidspunkt.	T03.003.01
	02C	Skal kunne vise «Tags» som er blitt opprettet for gitt video på gitt tidspunkt.	T03.003.02
Kryssreferanser			
Use Case ID	UC03.001		
Test ID	T03.003		
Informasjon			
Opphav til krav	Presentasjon bacheloroppgave og oppgavetekst.		
Kommentar			

K03 - GUI	
Krav ID	**K03.004C**
Navn	Spole
Beskrivelse	Skal kunne spole video framover og bakover. Qt Quick skal brukes til all GUI for applikasjonen.
Kryssreferanser	
Use Case ID	UC03.001
Test ID	T03.004
Informasjon	
Opphav til krav	Forslag til krav fra studentgruppen.
Kommentar	

K03 - GUI	
Krav ID	**K03.005A**
Navn	Live
Beskrivelse	Skal kunne hoppe direkte til «live stream» ved ett klikk. Qt Quick skal brukes til all GUI for applikasjonen.
Kryssreferanser	
Use Case ID	UC01.001, UC03.001
Test ID	T03.005
Informasjon	
Opphav til krav	Presentasjon bacheloroppgave og oppgavetekst.
Kommentar	

K03 - GUI			
Krav ID	**K03.006C**		
Navn	Tags		
Beskrivelse	Skal opprette en indikasjon på tidslinjen, som kalles tags. Tags er et lite bokmerke som plasseres på valgte tidspunkter i en video. Her skal det være mulig å legge til kommentarer slik at man raskt finner fram til spesielle hendelser på en video. Qt Quick skal brukes til all GUI for applikasjonen.		
Underkrav	Krav	Beskrivelse	Test ID
	01A	Indikasjonen skal være plassert på nåværende tidspunkt i videoen.	T03.006.01
	02B	Skal kunne «hovre» over indikasjonen for å få mer informasjon.	T03.006.02
	03A	Skal kunne ha mulighet til å skrive inn en kommentar.	T03.006.03
Kryssreferanser			
Use Case ID	UC01.001, UC03.001		
Test ID	T03.006		
Informasjon			
Opphav til krav	Forslag til krav fra studentgruppen.		
Kommentar			

K03 - GUI			
Krav ID	**K03.007C**		
Navn	Skjermdump		
Beskrivelse	Skal kunne ta et bilde av videoavspilleren. Qt Quick skal brukes til all GUI for applikasjonen.		
Underkrav	Krav	Beskrivelse	Test ID
	01A	Skal lagres med informasjon om tidspunkt.	T03.007.01
	02A	Skal lagres med informasjon om hvilken video den er hentet fra.	T03.007.02
	03B	Skal lagres med en kommentar gitt av brukeren.	T03.007.03
Kryssreferanser			
Use Case ID	UC01.001, UC03.001		
Test ID	T03.007		
Informasjon			
Opphav til krav	Forslag til krav fra studentgruppen.		
Kommentar			

K03 - GUI			
Krav ID	**K03.010A**		
Navn	Integrering av input enheter		
Beskrivelse	3D mus Space Navigator, Tobii EyeX Controller og Multi Touch Control Wheel skal integreres i applikasjonen og kontrollere kameraet.		
Underkrav	Krav	Beskrivelse	Test ID
	01A	3D Space Navigator skal integreres og styre kamera.	T03.010.01
	02A	Tobii EyeX skal integreres og styre kamera.	T03.010.02
	03A	MTCW skal integreres og styre kamera.	T03.010.03
Kryssreferanser			
Use Case ID	UC01.001, UC02.001, UC02.002, UC03.001, UC04.001, UC05.001		
Test ID			
Informasjon			
Opphav til krav	Gjennomgang med Rune (22.04.2016, Krona)		
Kommentar			

5.4 K04 – TOBII EYEX CONTROLLER

K04 – Tobii EyeX Controller	
Krav ID	**K04.001A**
Navn	Deteksjon
Beskrivelse	Eyetrackeren skal vite hvor på skjermen brukeren ser etter kalibrering.
Kryssreferanser	
Use Case ID	UC01.001, UC02.001, UC02.002, UC03.001, UC04.001, UC05.001
Test ID	T04.001
Informasjon	
Opphav til krav	Presentasjon bacheloroppgave og oppgavetekst.
Kommentar	

5.5 K05 – 3D MUS SPACE NAVIGATOR

K05 – 3D mus Space Navigator	
Krav ID	**K05.002A**
Navn	Posisjon
Beskrivelse	Skal kunne kontrolleres og registrere posisjon/endring ved bruk.
Kryssreferanser	
Use Case ID	UC01.001, UC02.001, UC02.002, UC03.001, UC04.001, UC05.001
Test ID	T05.002
Informasjon	
Opphav til krav	Presentasjon bacheloroppgave og oppgavetekst.
Kommentar	

5.6 K06 – MULTI TOUCH CONTROL WHEEL

K06 – Multi Touch Control Wheel	
Krav ID	**K06.001A**
Navn	Driver
Beskrivelse	For at man skal kunne benytte seg av MTCW skal det lages en applikasjonsdriver som kan benyttes i Qt på Windows 7, 64-bits operativsystem.
Kryssreferanser	
Use Case ID	UC01.001, UC02.001, UC02.002, UC03.001, UC04.001, UC05.001
Test ID	T06.001
Informasjon	
Opphav til krav	Presentasjon bacheloroppgave og oppgavetekst.
Kommentar	Greyhill har et eksempel på driver for MTCW man kan ta utgangspunkt i.

K06 – Multi Touch Control Wheel			
Krav ID	**K06.007A**		
Navn	Grensesnitt		
Beskrivelse	Kameraet skal være motorisert og kunne styres med API fra krav K07.006A.		
Underkrav	Krav	Beskrivelse	Test ID
	01A	Skal kunne detektere fingre	T06.007.01
	02A	Skal kunne bruke og kontrollere rotasjonsenkoderen	T06.007.02
	03A	Skal kunne brukes til å zoome inn og ut på kameraet	T06.007.03
	04A	Skal kunne styre kameraposisjon i x/y retning	T06.007.04
Kryssreferanser			
Use Case ID	UC01.001, UC02.001, UC02.002, UC03.001, UC04.001, UC05.001		
Test ID	T07.005		
Informasjon			
Opphav til krav	Presentasjon bacheloroppgave og oppgavetekst.		
Kommentar			

5.7 K07 - VIDEOKAMERA

K07 – Videokamera	
Krav ID	**K07.001A**
Navn	Oppløsning
Beskrivelse	Skal ha støtte for H.264.
Kryssreferanser	
Use Case ID	UC01.001, UC02.001, UC04.001
Test ID	T07.001
Informasjon	
Opphav til krav	Presentasjon bacheloroppgave og oppgavetekst.
Kommentar	

K07 – Videokamera			
Krav ID	**K07.002A**		
Navn	Bildefrekvens		
Beskrivelse	Kameraet skal kunne strømme med forskjellig bildefrekvens.		
Underkrav	Krav	Beskrivelse	Test ID
	01A	Kameraet skal kunne streame med 25 FPS.	T07.002.01
	02A	Kameraet skal kunne streame med 50 FPS.	T07.002.02
Kryssreferanser			
Use Case ID	UC01.001, UC02.001		
Test ID	T07.002		
Informasjon			
Opphav til krav	Presentasjon bacheloroppgave og oppgavetekst.		
Kommentar	Her er det ønskelig med 60 FPS hvis det er mulig innenfor ønsket pris.		

K07 – Videokamera			
Krav ID	**K07.005A**		
Navn	Motorisert		
Beskrivelse	Kameraet skal være motorisert og kunne styres med API fra krav K07.006A.		
Underkrav	Krav	Beskrivelse	Test ID
	01A	Skal kunne styres fra -170 til +170 grader horisontalt.	T07.005.01
	02A	Skal kunne styres fra -20 til +90 grader vertikalt.	T07.005.02
Kryssreferanser			
Use Case ID	UC02.002, UC05.001		
Test ID	T07.005		
Informasjon			
Opphav til krav	Presentasjon bacheloroppgave og oppgavetekst.		
Kommentar			

K07 – Videokamera	
Krav ID	**K07.006A**
Navn	API
Beskrivelse	Kamera skal ha en API som skal kunne benyttes i Qt Quick applikasjoner. Det skal utvikles en generell API slik at det kan benyttes flere forskjellige kameraer, og at dette kan byttes ut hvis ønskelig.
Kryssreferanser	
Use Case ID	UC02.002, UC05.001
Test ID	T07.006
Informasjon	
Opphav til krav	Presentasjon bacheloroppgave og oppgavetekst.
Kommentar	

K07 – Videokamera			
Krav ID	**K07.007A**		
Navn	Grensesnitt		
Beskrivelse	Man skal kunne koble seg til kameraets real-time videobilde.		
Kryssreferanser			
Use Case ID	UC01.001, UC02.001, UC02.002, UC04.001, UC05.001		
Test ID	T07.007		
Informasjon			
Opphav til krav	Presentasjon bacheloroppgave og oppgavetekst.		
Kommentar			

K07 – Videokamera			
Krav ID	**K07.008B**		
Navn	Modus		
Beskrivelse	Kameraet skal være motorisert og kunne styres med API fra krav K07.006A.		
Underkrav	Krav	Beskrivelse	Test ID
	01C	Skal ha infrarød modus.	
	02A	Skal ha nattmodus	
Kryssreferanser			
Use Case ID	UC02.002, UC04.001		
Test ID			
Informasjon			
Opphav til krav	Presentasjon bacheloroppgave og oppgavetekst.		
Kommentar			

VEDLEGG

TIDLIGERE DOKUMENTHISTORIKK

Tabell 4 - Dokumenthistorikk versjon 0-1.1.1

Versjon	Dato	Skrevet av	Dato
0.1.0	Opprettet alle krav	Vegard, Alexander	22.01.2016
0.2.0	La til Test ID på samtlige krav og underkrav	Vegard	24.01.2016
0.2.1	Noen Use Case nummer er lagt inn.	CP	27.01.2016
0.2.2	Slettet krav K04.003A Kameraposisjon og fjernet krav om HDMI inngang på krav K07.001A	Svein	28.01.2016
0.2.3	Det meste av Use Case nummer er lagt inn, fortsatt noe småtteri som gjenstår.	CP	28.01.2016
0.2.4	Use Case UC06.001 ble byttet til UC02.002, dette er nå blitt oppdatert.	CP	28.01.2016
0.3.0	Alle Use Case nummer er oppdatert	CP	29.01.2016
0.3.1	Oppdatert referanser om tag og skjermdump	Svein	29.01.2016
0.4.0	Gjennomgang av krav i forhold til tester. K01.002B.01B ble byttet til K01.002B.01A, K03.001A.10A ble byttet til K03.001A.10C.	CP	03.02.2016
0.5.0	Lagt til Qt Quick krav i beskrivelse for krav: K03.001A, K03.002B, K03.003A, K03.004B, K03.005A, K03.006B og K03.007B. Fjernet info om database i beskrivelse for krav K01.001A. Omformulering av beskrivelse i krav K05.002A, K06.004A, K07.005A og K07.006A. Lagt til krav: K01.005A og K03.008A. Endret prioritet på krav: K01.001A.03B til K01.001A.03C, K03.006B til K03.006C, K01.002B til K01.002C, K03.007B til K03.007C. Fjernet alle underkrav til K01.003A, K07.007A, K06.003A, K06.005A, K06.006A, K05.002A,	Alexander	05.02.2016
0.6.0	Endret kravbeskrivelse på K01.001A.02A fra 4 klienter til 3 klienter. Endret kravbeskrivelse på K01.001A.03C fra at den skal sende rå videostrøm til at den skal sende H.264. Endret kravbeskrivelse K01.003A fra: «Skal kunne motta video fra videokamera» til « Skal kunne motta video på server». Underkrav K01.004A.01C ble flyttet til K01.001A.05C. K01.004A slettet. La til underkravene K02.001A.04A, K02.001A.05A og K02.001A.06A La til underkrav K03.001A.13A Endret prioritering på spoling: K03.004B til K03.004C Fjernet alle underkrav til K03.004C Slettet kravene: K03.008A, K06.003A, K06.004A, K06.005A, K06.006A, K06.002A, K05.001A, K04.002A, K07.003B,	Vegard	18.02

	K07.004C Opprettet kravet K03.009A, K06.007A med underkravene 01A,02A,03A og 04A. Endret status på samtlige K07, bortsett fra K07.006A krav til: Ferdig K07.002A endret kravbeskrivelse. Endret navn på K01.002C fra Database til Lagring. Endret prioritet på K01.002C til K01.002A. Endret prioritet på underkravene: K01.002A.02B til K01.002A.02C, K01.002A.04B til K01.002A.04C, K01.002A.05B til K01.002A.05C		
1.0.0	Endret kravene: K03.001A.03A til K03.001A.03C , K03.001A.04A til K03.001A.04C, K03.001A.07A til K03.001A.07C, K03.001A.08A til K03.001A.08C, K03.001A.13A til K03.001A.13B, K03.003A til K03.003B, K01.002A.04B til K01.002A.04C, K01.002A.05B til K01.002A.05C, K01.002A.02B til K01.002A.02C, K02.001A.05A til K02.001A.05B, K03.002A til K03.002B, Endret beskrivelse: K06.007A	Felles	02.03.2016
1.0.1	Endret tekst i K01.001A.04A: 3B -> 3C Endret tekst i K06.007A.02A: å til og		
1.1.0	Endret tekst: K01.005A, K02.001A, K06.007A, K07.001A, K07.002A.01A, K07.002A.02B. Endret prioritet: K03.001A.05A til K03.001A.05B K03.001A.06A til K03.001A.06B K07.002A.02B til K07.002A.02A Opprettet krav: K03.010A Slettet krav: K03.009A K03.001A.06A	Felles	22.04.2016
1.1.1	Endret prioritet: K01.001A.03C til K01.001A.03A Endret tekst: K01.001A.03A	Carl Peder	29.04.2016

REFERANSER

[1] – Resultatet av design: http://www.ntnu.no/lie/fag/maler-standarder/UP/resultat-av-design.html (04.03.2016)
[2] – Designdokument: Bacheloroppgave. Consulant Calendar Overview (04.03.2016)
[3] – UML IN A NUTSHELL: Sinan Si Albir (03.03.2016)
[4] – UML Distilled Third Edition: Martin Fowler (03.03.2016)
[5] – Schaum's Outline's UML: Simon Bennett, John Skelton (17.02.2016)

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

TESTSPESIFIKASJON – v 2.0

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENT 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 4
1.3 HISTORIKK 4
2 FIGURER OG TABELLER 5
2.1 AKRONYMER OG FORKORTELSER 5
2.2 ORDBOK 5
2.3 TABELLER 5
2.4 FIGURER OG BILDER 5
3 KLASSER 6
4 TESTUTSTYR 6
5 TESTSPESIFIKASJON 7
VEDLEGG 25
TIDLIGERE DOKUMENTHISTORIKK 25
REFERANSER 26

1 DOKUMENT

Dette dokumentet beskriver godkjenningstester slik de foreligger pr. revisjonsdato for systemet under utvikling. Testspesifikasjoner og scenarioer beskrevet i dette dokumentet bygger på revisjoner av kravspesifikasjonene. Revisjonsnummeret i dette dokumentet vil følge revisjonsnummeret i dokumentet for kravspesifikasjoner.

1.1 OVERSIKT

Testspesifikasjon er direkte knyttet opp mot kravspesifikasjon ved å gi et kort overblikk over hvordan hvert krav skal testes. For hver test er det listet opp krav ID, beskrivelse av kravet, prioritet, risiko, test ID, testprosedyre og akseptkriteria. Testprosedyren forteller hvordan testen skal gjennomføres. Akseptkriteria beskriver hva som kreves for at testen skal bli godkjent. Prioritet forteller om testen som skal gjennomføres, er viktig eller ønskelig. Risikoklasse forteller hvilken risiko det har i forhold til resten av systemet hvis testen feiler. Ved enkelte tester er det satt opp underkrav med påfølgende undertester. Disse må utføres før man går videre og tester hovedkravet.

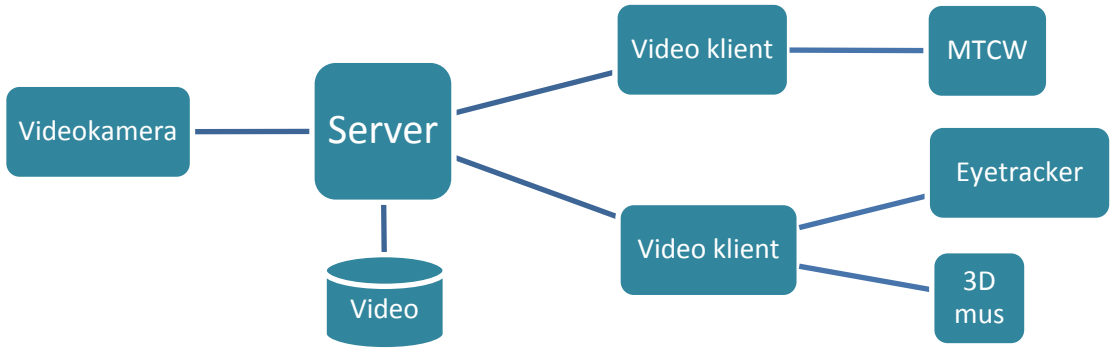
FORMÅL

Hensikten med testene beskrevet i dokumentet er for å verifisere systemet opp mot systemkravene.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 – Systemoversikt

1.3 HISTORIKK

Se vedlegg for tidligere dokumenthistorikk før V2.0.

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
2.0	19.05.2016	Alexander Schinnes	Korrekturlesning og retting. Oppdatert endringer i forhold til kravspesifikasjon.

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 4
Tabell 2 - Akronymer og forkortelser 5
Tabell 3 - Ordbok 5
Tabell 4 - Klasser 6
Tabell 5 – Historikk V0.1 – V1.1.2 25

2.4 FIGURER OG BILDER

Figur 1 – Systemoversikt 4

3 KLASSER

Tabell 4 - Klasser

Prioriteringsklasser	
Klasse	*Beskrivelse*
A	Svært viktig
B	Viktig
C	Ønskelig

Risikoklasser	
Klasse	*Beskrivelse*
1	Høy
2	Middels
3	Lav

Prioriteringsklasser forteller om kravet skal, bør eller kan gjøres.

Risikoklasser påpeker risikoen for resten av systemet hvis dette kravet ikke oppfylles.

4 TESTUTSTYR

- Kamera: Axis PTZ V5915
- Videokort: Picolo H.264
- Tobii Eyetracker
- Multi Touch Control Wheel (MTCW)
- 3D mus Space Navigator
- 4x stasjonære maskiner med tilbehør

5 TESTSPESIFIKASJON

T01 - System og server				
Test ID	**T01.001**			
Navn	Videostrøm			
Beskrivelse	Skal kunne motta video fra videokort og videresende strømmen til klient.			
Risikoklasse	A1			
Kryssreferanser				
Krav ID	K01.001A			
Underkrav	Krav ID	Beskrivelse	Test ID	Klasse
	K01.001A.01A	Skal ha maksimal forsinkelse på 200 ms fra handling oppstår til den vises i klientens videospiller.	T01.001.01	A1
	K01.001A.02A	Skal kunne sende video til 3 klienter.	T01.001.02	A1
	K01.001A.03A	Skal sende komprimert H.264 strøm til lagring, se krav K01.002C.	T01.001.03	A3
	K01.001A.04A	Skal kunne gjøre krav 02A og 03A samtidig.	T01.001.04	A1
	K01.001A.05C	Skal kunne stå på i 24 timer i døgnet i 7 dager i strekk.	T01.001.05	C1
Informasjon				
Testbeskrivelse	Black-box			
Testprosedyrer	T01.001	Gjennomfør individuelle tester til hvert underkrav.		
	Akseptkriteria	Alle underkrav av prioritet A er testet og godkjent.		
	T01.001.01	1. Hele systemet må være koblet opp. 2. Live videostrøm vises til klient. 3. Måler forsinkelsen det er fra kamera tar opp og sender videostrøm, til klient mottar og kan se videostrøm.		
	Akseptkriteria	Forsinkelse skal ikke være større enn 200ms.		
	T01.001.02	1. Kobler opp systemet til 3 klienter. 2. Verifiserer at alle 3 klienter har videostrøm.		
	Akseptkriteria	Alle klienter mottar og kan vise videostrøm.		
	T01.001.03	1. Kjører et videoopptak. 2. Verifiserer at videoopptak er mottatt og lagret.		
	Akseptkriteria	Rå videostrøm blir lagret i database.		
	T01.001.04	1. Kobler opp systemet til 3 klienter samtidig som komprimert H.264 videostrøm lagres i database. 2. Verifiserer at videostrøm blir sendt ut til alle klienter. 3. Verifiser at videostrøm blir lagret på server.		
	Akseptkriteria	Hvis krav 02A og 03A er tilfredsstilt samtidig.		
	T01.001.05	1. Lar server kjøre med et aktivt system kontinuerlig gjennom 7 dager. 2. Verifiser at serveren ikke krasjer eller lignende før tiden er gått.		
	Akseptkriteria	Serveren står uten komplikasjoner gjennom hele testperioden.		
Test laget av	Carl Peder Jansen			
Kommentarer				

T01 - System og server				
Test ID	**T01.002**			
Navn	Lagring			
Beskrivelse	Skal kunne lagre/hente opp alle videoene, tags (ref. K03.006C), og skjermdumper (ref. K03.007C)			
Risikoklasse	A1			
Kryssreferanser				
Krav ID	K01.002A			
Underkrav	Krav ID	Beskrivelse	Test ID	Klasse
	K01.002A.01A	Skal lagre all video som blir tatt opp.	T01.002.01	A1
	K01.002A.02C	Skal lagre alle tags som blir opprettet.	T01.002.02	C3
	K01.002A.03A	Skal kunne hente opp video.	T01.002.03	A1
	K01.002A.04C	Skal kunne hente opp tags.	T01.002.04	C3
	K01.002A.05C	Skal kunne hente opp skjermdumper.	T01.002.05	C3
Informasjon				
Testbeskrivelse	Black-box			
Testprosedyrer	T01.002	Gjennomfør individuelle tester til hvert underkrav.		
	Akseptkriteria	Alle underkrav av prioritet A er testet og godkjent.		
	T01.002.01, T01.002.02	1. Video blir tatt opp. 2. Under videoopptak blir en tag opprettet av bruker. 3. Verifiserer at video og tag blir lagret på server.		
	Akseptkriteria	Video og tags skal bli lagret systematisk på server.		
	T01.002.03, T01.002.04, T01.002.05	1. Video, tags og skjermdumper blir lagret på server. 2. Henter opp video eller skjermdump via GUI meny. 3. Verifiserer at riktig data blir hentet opp.		
	Akseptkriteria	Hvis all data blir hentet opp uten tap av data eller lignende.		
Test laget av	Carl Peder Jansen			
Kommentarer				

T01 - System og server				
Test ID	**T01.003**			
Navn	Videokort			
Beskrivelse	Skal kunne motta video fra videokamera på server.			
Risikoklasse	A1			
Kryssreferanser				
Krav ID	K01.003A			
Underkrav	Krav ID	Beskrivelse	Test ID	Klasse
Informasjon				
Testbeskrivelse	Black-box			
Testprosedyrer	T01.003	1. Kobler opp kamera mot videokort som står i serveren. 2. Slå på kamera. 3. Verifiser at det er videosignal på videokortet.		
	Akseptkriteria	Hvis det er video input på videokortet.		
Test laget av	Carl Peder Jansen			
Kommentarer				

<table>
<tr><th colspan="5">T01 - System og server</th></tr>
<tr><td>Test ID</td><td colspan="4">T01.005</td></tr>
<tr><td>Navn</td><td colspan="4">Analyse av videostrøm</td></tr>
<tr><td>Beskrivelse</td><td colspan="4">Det skal utføres analyse av hvordan forandring av FPS og oppløsning påvirker videostrømmens forsinkelse.</td></tr>
<tr><td>Risikoklasse</td><td colspan="4">A2</td></tr>
<tr><td colspan="5">Kryssreferanser</td></tr>
<tr><td>Krav ID</td><td colspan="4">K01.005A</td></tr>
<tr><td rowspan="2">Underkrav</td><td>Krav ID</td><td>Beskrivelse</td><td>Test ID</td><td>Klasse</td></tr>
<tr><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="5">Informasjon</td></tr>
<tr><td>Testbeskrivelse</td><td colspan="4">Black-box</td></tr>
<tr><td rowspan="2">Testprosedyrer</td><td>T01.005</td><td colspan="3">1. Systemet blir testet med forskjellige videostrømmer som har forskjellig FPS og oppløsning.
2. Oppløsning 1080p (1920 x 1080) ved FPS: 25 og 50.
3. Måler systemets forsinkelse.
4. Oppløsning 720p (1280 x 720) ved FPS: 25 og 50
5. Oppløsning 480 x 270 ved FPS: 25 og 50
6. Analyserer forskjellen i systemets forsinkelse.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Sluttanalysen må være rimelig.</td></tr>
<tr><td>Test laget av</td><td colspan="4">Carl Peder Jansen</td></tr>
<tr><td>Kommentarer</td><td colspan="4">Denne testen er kun aktuell når hele systemet begynner å komme på plass.</td></tr>
</table>

<table>
<tr><th colspan="5">T02 - Klient</th></tr>
<tr><td>Test ID</td><td colspan="4">T02.001</td></tr>
<tr><td>Navn</td><td colspan="4">Kommunikasjon</td></tr>
<tr><td>Beskrivelse</td><td colspan="4">Alle klienter skal ha kommunikasjon og dataoverføring fra server.</td></tr>
<tr><td>Risikoklasse</td><td colspan="4">A1</td></tr>
<tr><td colspan="5">Kryssreferanser</td></tr>
<tr><td>Krav ID</td><td colspan="4">K02.001A</td></tr>
<tr><td rowspan="8">Underkrav</td><td>Krav ID</td><td>Beskrivelse</td><td>Test ID</td><td>Klasse</td></tr>
<tr><td>K02.001A.01A</td><td>Skal kunne koble seg opp mot server.</td><td>T02.001.01</td><td>A1</td></tr>
<tr><td>K02.001A.02A</td><td>Skal kunne motta video fra server.</td><td>T02.001.02</td><td>A1</td></tr>
<tr><td>K02.001A.03A</td><td>Skal kunne videresende video til videoavspiller.</td><td>T02.001.03</td><td>A1</td></tr>
<tr><td>K02.001A.04A</td><td>Skal kunne styre kameraposisjon.</td><td>T02.001.04</td><td>A1</td></tr>
<tr><td>K02.001A.05B</td><td>Skal kunne velge videostrøm fra server</td><td>T02.001.05</td><td>B1</td></tr>
<tr><td>K02.001A.06A</td><td>Skal kunne velge å ta opptak av videostrøm på server.</td><td>T02.001.06</td><td>A1</td></tr>
<tr><td colspan="5">Informasjon</td></tr>
<tr><td>Testbeskrivelse</td><td colspan="4">Black-box</td></tr>
<tr><td rowspan="15">Testprosedyrer</td><td>T02.001</td><td colspan="3">Gjennomfør individuelle tester til hvert underkrav.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Alle underkrav av prioritet A er testet og godkjent.</td></tr>
<tr><td>T02.001.01</td><td colspan="3">1. Klient forsøker å koble seg til server.
2. Verifiser kommunikasjon mellom klient og server.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Klient får koblet seg opp mot server uten problemer.</td></tr>
<tr><td>T02.001.02</td><td colspan="3">1. Sender video server.
2. Verifiserer at klient mottar video.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Klienter skal motta video.</td></tr>
<tr><td>T02.001.03</td><td colspan="3">1. Video blir sendt til klient.
2. Video blir åpnet i videoavspiller.
3. Verifiser at video blir åpnet uten problemer.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Video blir videresendt til videoavspiller uten problemer.</td></tr>
<tr><td>T02.001.04</td><td colspan="3">1. Kjører kamera fra klientsiden via statiske kommandoer, eventuelt input enheter.
2. Verifiser at kommunikasjonen mellom input enheter-klient-server-kamera fungerer og at kameraet lar seg styre som tiltenkt.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Alle klienter skal kunne styre kamera. Kun en om gangen.</td></tr>
<tr><td>T02.001.05</td><td colspan="3">1. Klient velger videostrøm (H.264, H.264 nedskalert)
2. Verifiser at den valgte videostrøm aktiveres.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Alle klienter skal kunne velge videostrøm fra server.</td></tr>
<tr><td>T02.001.06</td><td colspan="3">1. Klient aktiverer opptaks knapp.
2. Verifiser at opptak av video starter.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Alle klienter skal kunne ta opptak av videostrøm på server.</td></tr>
<tr><td>Test laget av</td><td colspan="4">Carl Peder Jansen</td></tr>
<tr><td>Kommentarer</td><td colspan="4"></td></tr>
</table>

<table>
<tr><th colspan="5">T03 - GUI</th></tr>
<tr><td>Test ID</td><td colspan="4">T03.001</td></tr>
<tr><td>Navn</td><td colspan="4">Meny</td></tr>
<tr><td>Beskrivelse</td><td colspan="4">Skal kunne gi brukeren valgmuligheter som oppgitt i underkrav. Qt Quick skal brukes til all GUI for applikasjonen.</td></tr>
<tr><td>Risikoklasse</td><td colspan="4">A1</td></tr>
<tr><td colspan="5">Kryssreferanser</td></tr>
<tr><td>Krav ID</td><td colspan="4">K03.001A</td></tr>
<tr><td rowspan="12">Underkrav</td><td>Krav ID</td><td>Beskrivelse</td><td>Test ID</td><td>Klasse</td></tr>
<tr><td>K03.001A.01A</td><td>Skal kunne velge å se forsinkelse.</td><td>T03.001.01</td><td>A1</td></tr>
<tr><td>K03.001A.02A</td><td>Skal kunne velge å se bilder i sekundet (FPS).</td><td>T03.001.02</td><td>A1</td></tr>
<tr><td>K03.001A.03C</td><td>Skal kunne velge å vise «packet loss».</td><td>T03.001.03</td><td>C2</td></tr>
<tr><td>K03.001A.04C</td><td>Skal kunne velge å se datatrafikk.</td><td>T03.001.04</td><td>C2</td></tr>
<tr><td>K03.001A.05B</td><td>Skal kunne velge å se video i H.264 format.</td><td>T03.001.05</td><td>A1</td></tr>
<tr><td>K03.001A.07C</td><td>Skal kunne velge å se CPU forbruk.</td><td>T03.001.07</td><td>C2</td></tr>
<tr><td>K03.001A.08C</td><td>Skal kunne velge å se RAM forbruk.</td><td>T03.001.08</td><td>C2</td></tr>
<tr><td>K03.001A.09A</td><td>Skal kunne velge å vise videoavspilleren uten effekter.</td><td>T03.001.09</td><td>A1</td></tr>
<tr><td>K03.001A.10C</td><td>Skal kunne velge å vise videoavspilleren med IR.</td><td>T03.001.10</td><td>C1</td></tr>
<tr><td>K03.001A.11A</td><td>Skal kunne velge å vise videoavspilleren med nattmodus.</td><td>T03.001.11</td><td>B1</td></tr>
<tr><td>K03.001A.12A</td><td>Skal kunne velge å vise et «overlay» med all tilgjengelig informasjon.</td><td>T03.001.12</td><td>A2</td></tr>
<tr><td>K03.001A.13B</td><td>Skal kunne velge FPS på videostrømmen.</td><td>T03.001.13</td><td>B2</td></tr>
<tr><td colspan="5">Informasjon</td></tr>
<tr><td>Testbeskrivelse</td><td colspan="4">Black-box</td></tr>
<tr><td rowspan="14">Testprosedyrer</td><td>T03.001</td><td colspan="3">Gjennomføre alle individuelle krav for hvert underkrav.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Alle underkrav av prioritet A er testet og godkjent.</td></tr>
<tr><td>T03.001.01,
T03.001.02,
T03.001.03,
T03.001.04,
T03.001.07,
T03.001.08</td><td colspan="3">1. Velger de nevnte dataene i hvert av de aktuelle underkravene.
2. Overlay med aktuell informasjon vises.
3. Verifiser at informasjon vises på tiltenkt måte og at den er korrekt.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Data informasjon blir vist på tiltenkt måte og er korrekt.</td></tr>
<tr><td>T03.001.05</td><td colspan="3">1. H.264 format blir valgt.
2. Verifiser at det valgte formatet stemmer.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">H.264 format kan velges uten problemer.</td></tr>
<tr><td>T03.001.09,
T03.001.10,
T03.001.11</td><td colspan="3">1. Trykker på valgt videomodus.
2. Verifiser at riktig videomodus aktiveres.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Alle videomoduser fungerer som de skal i forhold til kravene.</td></tr>
<tr><td>T03.001.12</td><td colspan="3">1. Velger overlay.
2. Verifiser at alt av informasjon stemmer og er vist på tiltenkt måte.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Overlay viser korrekt informasjon på en oversiktlig måte.</td></tr>
<tr><td>T03.001.13</td><td colspan="3">1. Det endres FPS via GUI.
2. Verifiser at FPS endres.</td></tr>
</table>

	Akseptkriteria	Det er mulig å endre FPS på videostrømmen.
Test laget av	Carl Peder Jansen	
Kommentarer		

T03 - GUI				
Test ID	**T03.002**			
Navn	Play/Pause			
Beskrivelse	Skal kunne pause og spille av video.			
Risikoklasse	B1			
Kryssreferanser				
Krav ID	K03.002B			
Underkrav	Krav ID	Beskrivelse	Test ID	Klasse
	K03.002A.01A	Skal kunne spille av video fra gitt punkt.	T03.002.01	A1
	K03.002A.02A	Skal kunne pause video.	T03.002.02	A1
Informasjon				
Testbeskrivelse	Black-box			
Testprosedyrer	T03.002, T03.002.01, T03.002.02	1. Video er hentet opp og er åpnet i videoavspiller. 2. Trykker på play. 3. Verifiser at video starter. 4. Trykker på pause. 5. Verifiser at video pauser.		
	Akseptkriteria	Video starter eller pauser når knappene blir trykket på.		
Test laget av	Carl Peder Jansen			
Kommentarer				

<table>
<tr><th colspan="5">T03 - GUI</th></tr>
<tr><td>Test ID</td><td colspan="4">T03.003</td></tr>
<tr><td>Navn</td><td colspan="4">Tidslinje</td></tr>
<tr><td>Beskrivelse</td><td colspan="4">Skal kunne vise hvor lenge du har sett på en video på en tidslinje, hvor 0 er starten på videoen og slutten viser lengden på videoen. Qt Quick skal brukes til all GUI for applikasjonen.</td></tr>
<tr><td>Risikoklasse</td><td colspan="4">B1</td></tr>
<tr><td colspan="5">Kryssreferanser</td></tr>
<tr><td>Krav ID</td><td colspan="4">K03.003B</td></tr>
<tr><td rowspan="3">Underkrav</td><td>Krav ID</td><td colspan="2">Beskrivelse</td><td>Test ID</td><td>Klasse</td></tr>
<tr><td>K03.003B.01B</td><td colspan="2">Skal kunne trykke på tidslinjen og hoppe til gitt tidspunkt.</td><td>T03.003.01</td><td>B3</td></tr>
<tr><td>K03.003B.02C</td><td colspan="2">Skal kunne vise «Tags» som er blitt opprettet for gitt video på gitt tidspunkt.</td><td>T03.003.02</td><td>C3</td></tr>
<tr><td colspan="5">Informasjon</td></tr>
<tr><td>Testbeskrivelse</td><td colspan="4">Black-box</td></tr>
<tr><td rowspan="7">Testprosedyrer</td><td>T03.003</td><td colspan="3">1. Video er hentet opp og er åpnet i videoavspiller.
2. Ved start av video skal tidslinje vise 00:00 sekunder
3. Starter video.
4. Verifiser at tiden og tidslinjen beveger seg etter hvor lenge filmen har gått.
5. Spiller filmen ferdig.
6. Verifiser at tidslinjen er i enden og filmens totale lengde er korrekt.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Tidslinje viser og beveger seg riktig i forhold til videoens avspilt lengde.</td></tr>
<tr><td>T03.003.01</td><td colspan="3">1. Video er hentet opp og er åpnet i videoavspiller.
2. «Hopp» til filmens halverte totallengde ved å trykke på tidslinjen.
3. Verifiser at tidslinjen hopper dit den skal og at tiden stemmer i forhold til filmens totale lengde.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Man kan trykke på tidslinjen og hoppe til ønsket tidspunkt.</td></tr>
<tr><td>T03.003.02</td><td colspan="3">1. Video med lagrede tags er hentet opp og er åpnet i videoavspiller.
2. Spill av film
3. Verifiser at tags blir vist når videoen er på det gitte tidspunkt hvor taggen befinner seg.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Taggen blir vist ved gitt tidspunkt.</td></tr>
<tr><td>Test laget av</td><td colspan="4">Carl Peder Jansen</td></tr>
<tr><td>Kommentarer</td><td colspan="4"></td></tr>
</table>

T03 - GUI				
Test ID	**T03.004**			
Navn	Spole			
Beskrivelse	Skal kunne spole video framover og bakover. Qt Quick skal brukes til all GUI for applikasjonen.			
Risikoklasse	C1			
Kryssreferanser				
Krav ID	K03.004C			
Underkrav	Krav ID	Beskrivelse	Test ID	Klasse
Informasjon				
Testbeskrivelse	Black-box			
Testprosedyrer	T03.004.	1. Video er åpnet i videoavspiller 2. Spill av video i 5x hastighet fremover og bakover. 3. Verifiser at video spilles av i 5x hastighet både fremover og bakover.		
	Akseptkriteria	Video kan spilles av i 5x hastighet både fremover og bakover.		
Test laget av	Carl Peder Jansen			
Kommentarer				

T03 - GUI				
Test ID	**T03.005**			
Navn	Live			
Beskrivelse	Skal kunne hoppe direkte til «live stream» ved ett klikk. Qt Quick skal brukes til all GUI for applikasjonen.			
Risikoklasse	A1			
Kryssreferanser				
Krav ID	K03.005A			
Underkrav	Krav ID	Beskrivelse	Test ID	Klasse
Informasjon				
Testbeskrivelse	Black-box			
Testprosedyrer	T03.005	1. Kamera er aktivt. 2. Videoavspiller er åpnet og et filmopptak spilles av. 3. Live knapp aktiveres. 4. Filmopptak stanses og videoavspilleren hopper direkte til live videostrøm. 5. Verifiser at den lagrede videoen stopper. 6. Verifiser at videoavspilleren får live videostrøm fra kamera som ny input.		
	Akseptkriteria	Videoavspiller hopper direkte til live videostrøm.		
Test laget av	Carl Peder Jansen			
Kommentarer				

<table>
<tr><th colspan="5">T03 - GUI</th></tr>
<tr><td>Test ID</td><td colspan="4">T03.006</td></tr>
<tr><td>Navn</td><td colspan="4">Tags</td></tr>
<tr><td>Beskrivelse</td><td colspan="4">Skal opprette en indikasjon på tidslinjen, som kalles tags. Tags er et lite bokmerke som plasseres på valgte tidspunkter i en video. Her skal det være mulig å legge til kommentarer slik at man raskt finner fram til spesielle hendelser på en video. Qt Quick skal brukes til all GUI for applikasjonen.</td></tr>
<tr><td>Risikoklasse</td><td colspan="4">C2</td></tr>
<tr><td colspan="5">Kryssreferanser</td></tr>
<tr><td>Krav ID</td><td colspan="4">K03.006C</td></tr>
<tr><td rowspan="4">Underkrav</td><td>Krav ID</td><td colspan="2">Beskrivelse</td><td>Test ID</td><td>Klasse</td></tr>
<tr><td>K03.006C.01A</td><td colspan="2">Indikasjonen skal være plassert på nåværende tidspunkt i videoen.</td><td>T03.006.01</td><td>A3</td></tr>
<tr><td>K03.006C.02B</td><td colspan="2">Skal kunne «hovre» over indikasjonen for å få mer informasjon.</td><td>T03.006.02</td><td>A3</td></tr>
<tr><td>K03.006C.03A</td><td colspan="2">Skal kunne ha mulighet til å skrive inn en kommentar.</td><td>T03.006.03</td><td>B3</td></tr>
<tr><td colspan="5">Informasjon</td></tr>
<tr><td>Testbeskrivelse</td><td colspan="4">Black-box</td></tr>
<tr><td rowspan="9">Testprosedyrer</td><td>T03.006</td><td colspan="3">Gjennomfør individuelle tester til hvert underkrav.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Alle underkrav må være oppfylt.</td></tr>
<tr><td>T03.006.01</td><td colspan="3">1. En ferdig videofil med inkluderende tags åpnes i videoavspiller.
2. Video spilles av.
3. Verifiserer at tag er på riktig tidspunkt på tidslinjen samt at den vises når videoen kommer til dette tidspunktet.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Indikasjon er plassert på riktig sted og vises ved gitt tidspunkt.</td></tr>
<tr><td>T03.006.02</td><td colspan="3">1. En ferdig videofil med inkluderende tags åpnes i videoavspiller.
2. «Hovrer» over den aktuelle tag/indikasjonen ved hjelp av input enhet.
3. Verifiser at mer informasjon rundt den aktuelle taggen kommer opp.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Informasjon kommer opp når det «hovres» over den aktuelle taggen.</td></tr>
<tr><td>T03.006.03</td><td colspan="3">1. En ferdig videofil med inkluderende tags åpnes i videoavspiller.
2. Skriver inn en kommentar i tag/indikasjon.
3. Verifiser at notatet blir lagret.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Kommentar kan opprettes og lagres.</td></tr>
<tr><td>Test laget av</td><td colspan="4">Carl Peder Jansen</td></tr>
<tr><td>Kommentarer</td><td colspan="4"></td></tr>
</table>

<table>
<tr><th colspan="5">T03 - GUI</th></tr>
<tr><td>Test ID</td><td colspan="4">T03.007</td></tr>
<tr><td>Navn</td><td colspan="4">Skjermdump</td></tr>
<tr><td>Beskrivelse</td><td colspan="4">Skal kunne ta et bilde av videoavspilleren. Qt Quick skal brukes til all GUI for applikasjonen.</td></tr>
<tr><td>Risikoklasse</td><td colspan="4">C2</td></tr>
<tr><td colspan="5">Kryssreferanser</td></tr>
<tr><td>Krav ID</td><td colspan="4">K03.007C</td></tr>
<tr><td rowspan="4">Underkrav</td><td>Krav ID</td><td>Beskrivelse</td><td>Test ID</td><td>Klasse</td></tr>
<tr><td>K03.007C.01A</td><td>Skal lagres med informasjon om tidspunkt.</td><td>T03.007.01</td><td>A3</td></tr>
<tr><td>K03.007C.02A</td><td>Skal lagres med informasjon om hvilken video.</td><td>T03.007.02</td><td>A3</td></tr>
<tr><td>K03.007C.03B</td><td>Skal lagres med en kommentar gitt av brukeren.</td><td>T03.007.03</td><td>B3</td></tr>
<tr><td colspan="5">Informasjon</td></tr>
<tr><td>Testbeskrivelse</td><td colspan="4">Black-box</td></tr>
<tr><td rowspan="4">Testprosedyrer</td><td>T03.007,
T03.007.01,
T03.007.02</td><td colspan="3">1. Live videostrøm er aktiv.
2. Tar skjermdump av video.
3. Verifiser at skjermdump blir lagret med informasjon om tidspunkt for skjermdumpen og hvilken video den ble tatt fra.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Skjermdump av video kan utføres samt informasjon om tidspunkt og video opphav blir lagret.</td></tr>
<tr><td>T03.007.03</td><td colspan="3">1. Live videostrøm er aktiv.
2. Tar skjermdump av video.
3. Legger til kommentar i skjermdump.
4. Verifiser at kommentaren blir lagret med skjermdumpen.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Kommentar blir lagret i skjermdump.</td></tr>
<tr><td>Test laget av</td><td colspan="4">Carl Peder Jansen</td></tr>
<tr><td>Kommentarer</td><td colspan="4"></td></tr>
</table>

K03 - GUI				
Test ID	**T03.010**			
Navn	Integrering av input enheter			
Beskrivelse	3D mus Space Navigator, Tobii EyeX Controller og Multi Touch Control Wheel skal Integreres i applikasjonen og kontrollere kameraet.			
Risikoklasse	A1			
Kryssreferanser				
Krav ID	K03.010A			
Underkrav	Krav ID	Beskrivelse	Test ID	Klasse
	K03.010A.01A	3D Space Navigator skal integreres og styre kamera.	T03.010.01	A1
	K03.010A.02A	Tobii EyeX skal integreres og styre kamera.	T03.010.02	A1
	K03.010A.03A	MTCW skal integreres og styre kamera.	T03.010.03	A1
Informasjon				
Testbeskrivelse	Black-box			
Testprosedyrer	T03.010	Gjennomfør individuelle tester til hvert underkrav.		
	Akseptkriteria	Alle underkrav må være oppfylt.		
	T03.010.01	1. 3D Space Navigator kobles til PC. 2. Forsøker å kontrollere kameraets x, y og z akse.		
	Akseptkriteria	Kameraet x, y og z akse kan kontrolleres uten problemer		
	T03.010.02	1. Tobii EyeX kobles til PC. 2. Forsøker å kontrollere kameraets x, y og z akse.		
	Akseptkriteria	Kameraet x og y akse kan kontrolleres uten problemer		
	T03.010.03	1. MTCW kobles til PC. 2. Forsøker å kontrollere kameraets x, y og z akse.		
	Akseptkriteria	Kameraet x, y og z akse kan kontrolleres uten problemer		
Test laget av	Carl Peder Jansen			
Kommentarer				

T04 – Tobii EyeX Controller				
Test ID	**T04.001**			
Navn	Deteksjon			
Beskrivelse	Eyetrackeren skal vite hvor på skjermen brukeren ser etter kalibrering.			
Risikoklasse	A1			
Kryssreferanser				
Krav ID	K04.001A			
Underkrav	Krav ID	Beskrivelse	Test ID	Klasse
Informasjon				
Testbeskrivelse	Black-box			
Testprosedyrer	T04.001	1. Eyetracker kalibreres med Tobii software. 2. Ser på skjerm. 3. Verifiserer at eyetrackeren er kalibrert slik at eyetrackeren fungerer slik den er tiltenkt.		
	Akseptkriteria	Eyetracker skal vite hvor på skjermen brukeren ser, etter kalibrering.		
Test laget av	Carl Peder Jansen			
Kommentarer				

T05 – 3D mus Space Navigator				
Test ID	**T05.002**			
Navn	Posisjon			
Beskrivelse	Skal kunne kontrolleres og registrere posisjon/endring ved bruk.			
Risikoklasse	A2			
Kryssreferanser				
Krav ID	K05.002A			
Underkrav	Krav ID	Beskrivelse	Test ID	Klasse
Informasjon				
Testbeskrivelse	Black-box			
Testprosedyrer	T05.002	1. Prøver hver av 3D musens funksjoner: • Opp, ned, høyre, venstre • Rotasjon høyre/venstre • Trykk ned • Dra opp • Sideknapper 2. Verifiser at alle funksjoner sender tilbake datainformasjon om posisjon/status og at disse er korrekte.		
	Akseptkriteria	Alle funksjoner sender tilbake datainformasjon som er korrekt.		
Test laget av	Carl Peder Jansen			
Kommentarer				

T06 – Multi Touch Control Wheel				
Test ID	**T06.001**			
Navn	Driver			
Beskrivelse	For at man skal kunne benytte seg av MTCW skal det lages en applikasjonsdriver som kan benyttes i Qt på Windows 7, 64-bits operativsystem.			
Risikoklasse	A1			
Kryssreferanser				
Krav ID	K06.001A			
Underkrav	Krav ID	Beskrivelse	Test ID	Klasse
Informasjon				
Testbeskrivelse	Black-box			
Testprosedyrer	T06.001	1. Kobler MTCW opp mot datamaskin som har Windows 7, 64-bits operativsystem. 2. Prøve alle funksjoner i en Qt applikasjon. 3. Verifiserer at funksjonen fungerer på den måten de var tiltenkt.		
	Akseptkriteria	MTCW kan brukes i Qt på Windows 7, 64-bits operativsystem.		
Test laget av	Carl Peder Jansen			
Kommentarer				

<table>
<tr><th colspan="5">T06 – Multi Touch Control Wheel</th></tr>
<tr><td>Test ID</td><td colspan="4">T06.007</td></tr>
<tr><td>Navn</td><td colspan="4">Grensesnitt</td></tr>
<tr><td>Beskrivelse</td><td colspan="4">Kameraet skal være motorisert og kunne styres med API fra krav K07.006A.</td></tr>
<tr><td>Risikoklasse</td><td colspan="4">A1</td></tr>
<tr><td colspan="5">Kryssreferanser</td></tr>
<tr><td>Krav ID</td><td colspan="4">K06.007A</td></tr>
<tr><td rowspan="5">Underkrav</td><td>Krav ID</td><td>Beskrivelse</td><td>Test ID</td><td>Klasse</td></tr>
<tr><td>K06.007A.01A</td><td>Skal kunne detektere fingre</td><td>T06.007.01</td><td>A1</td></tr>
<tr><td>K06.007A.02A</td><td>Skal kunne bruke å kontrollere rotasjonsenkoderen</td><td>T06.007.02</td><td>A3</td></tr>
<tr><td>K06.007A.03A</td><td>Skal kunne brukes til å zoome inn og ut på kameraet</td><td>T06.007.03</td><td>A3</td></tr>
<tr><td>K06.007A.04A</td><td>Skal kunne styre kameraposisjon i x/y retning</td><td>T06.007.04</td><td>A1</td></tr>
<tr><td colspan="5">Informasjon</td></tr>
<tr><td>Testbeskrivelse</td><td colspan="4">Black-box</td></tr>
<tr><td rowspan="11">Testprosedyrer</td><td>T06.007</td><td colspan="3">Gjennomfør individuelle tester til hvert underkrav.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Alle tester til underkravene er godkjent.</td></tr>
<tr><td>T06.007.01</td><td colspan="3">1. Holder 1, 2 og 3 fingre på flaten.
2. Verifiser at riktig antall fingre er registrert.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">MTCW Skal kunne detektere opp til 3 fingre på flaten samtidig.</td></tr>
<tr><td>T06.007.02</td><td colspan="3">1. Roterer med klokken.
2. Verifiser at tallverdi øker.
3. Roter mot klokken.
4. Verifiser at tallverdi senkes.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Man skal kunne bruke rotasjonsenkoderen til å øke og senke en tallverdi.</td></tr>
<tr><td>T06.007.03</td><td colspan="3">1. Zoom ved hjelp av rotasjonsenkoderen eller ved 2 fingre som dras fra hverandre for å zoome inn og mot hverandre for å zoome ut.
2. Verifiser at kameraet zoomer ut/inn.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Kameraet skal zoome inn og ut ved bruk av MTCW.</td></tr>
<tr><td>T06.007.04</td><td colspan="3">1. Styrer posisjon ved å dra en finger rundt på padden.
2. Verifiser at x og y posisjon blir registrert og at kameraet beveger seg deretter.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Skal registrere om man trykker på piltastene, disse verdiene skal da brukes til å styre kameraets x og y akse.</td></tr>
<tr><td>Test laget av</td><td colspan="4">Carl Peder Jansen</td></tr>
<tr><td>Kommentarer</td><td colspan="4"></td></tr>
</table>

T07 – Videokamera				
Test ID	**T07.001**			
Navn	Oppløsning			
Beskrivelse	Skal ha støtte for H.264.			
Risikoklasse	A3			
Kryssreferanser				
Krav ID	K07.001A			
Underkrav	Krav ID	Beskrivelse	Test ID	Klasse
Informasjon				
Testbeskrivelse	Black-box			
Testprosedyrer	T07.001	Verifiserer at kamera har støtte for H.264 format.		
	Akseptkriteria	Skal ha støtte for H.264 format.		
Test laget av	Carl Peder Jansen			
Kommentarer				

T07 – Videokamera				
Test ID	**T07.002**			
Navn	Bildefrekvens			
Beskrivelse	Kameraet skal kunne strømme med forskjellig bildefrekvens.			
Risikoklasse	A3			
Kryssreferanser				
Krav ID	K07.002A			
Underkrav	Krav ID	Beskrivelse	Test ID	Klasse
	K07.002.01A	Kameraet skal kunne streame med 25 FPS.	T07.002.01	A3
	K07.002.02A	Kameraet skal kunne streame med 50 FPS.	T07.002.02	A3
Informasjon				
Testbeskrivelse	Black-box			
Testprosedyrer	T07.002	Gjennomfør individuelle tester til hvert underkrav.		
	Akseptkriteria	Alle tester til underkravene er godkjent.		
	T07.002.01, T07.002.02	Verifiserer at kamera har støtte for streaming med 25 og 50 FPS.		
	Akseptkriteria	Kamera har støtte for 25 og 50 FPS.		
Test laget av	Carl Peder Jansen			
Kommentarer				

<table>
<tr><th colspan="5">T07 – Videokamera</th></tr>
<tr><td>Test ID</td><td colspan="4">T07.005</td></tr>
<tr><td>Navn</td><td colspan="4">Motorisert</td></tr>
<tr><td>Beskrivelse</td><td colspan="4">Kameraet skal være motorisert og kunne styres med API fra krav K07.006A.</td></tr>
<tr><td>Risikoklasse</td><td colspan="4">A1</td></tr>
<tr><td colspan="5">Kryssreferanser</td></tr>
<tr><td>Krav ID</td><td colspan="4">K07.005A</td></tr>
<tr><td rowspan="3">Underkrav</td><td>Krav ID</td><td>Beskrivelse</td><td>Test ID</td><td>Klasse</td></tr>
<tr><td>K07.005A.01A</td><td>Skal kunne styres fra -170 til +170 grader horisontalt.</td><td>T07.005.01</td><td>A1</td></tr>
<tr><td>K07.005A.02A</td><td>Skal kunne styres fra -20 til +90 grader vertikalt.</td><td>T07.005.02</td><td>A1</td></tr>
<tr><td colspan="5">Informasjon</td></tr>
<tr><td>Testbeskrivelse</td><td colspan="4">Black-box</td></tr>
<tr><td rowspan="7">Testprosedyrer</td><td>T07.005</td><td colspan="3">Gjennomfør individuelle tester til hvert underkrav.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Alle underkrav må være testet og godkjent.</td></tr>
<tr><td>T07.005.01</td><td colspan="3">1. Styrer kamera via API nevnt i K07.006A.
2. Kjører kamera til hver sin horisontale ende.
3. Verifiser at kamera flyttes fra -170 til +170 grader horisontalt.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Kamera kan kjøres fra -170 til +170 grader horisontalt.</td></tr>
<tr><td>T07.005.02</td><td colspan="3">1. Styrer kamera via API nevnt i K07.006A.
2. Kjører kamera til hver sin vertikale ende.
3. Verifiser at kamera flyttes fra -20 til +90 grader vertikalt.</td></tr>
<tr><td>Akseptkriteria</td><td colspan="3">Kamera kan kjøres fra -20 til +90 grader vertikalt.</td></tr>
<tr><td>Test laget av</td><td colspan="4">Carl Peder Jansen</td></tr>
<tr><td>Kommentarer</td><td colspan="4"></td></tr>
</table>

T07 – Videokamera				
Test ID	**T07.006**			
Navn	API			
Beskrivelse	Kamera skal ha en API som skal kunne benyttes i Qt Quick applikasjoner. Det skal utvikles en generell API slik at det kan benyttes flere forskjellige kameraer, og at dette kan byttes ut hvis ønskelig.			
Risikoklasse	A1			
Kryssreferanser				
Krav ID	K07.006A			
Underkrav	Krav ID	Beskrivelse	Test ID	Klasse
Informasjon				
Testbeskrivelse	Black-box			
Testprosedyrer	T07.006	1. Verifiser at API er kompatibel med Qt Quick applikasjoner. 2. Kjører API opp mot det kamera vi har. 3. Verifiser at API gjør det mulig å styre kameraet gjennom Qt Quick.		
	Akseptkriteria	API er kompatibel med Qt Quick applikasjoner og kan styre kamera på en tilfredsstillende måte.		
Test laget av	Carl Peder Jansen			
Kommentarer	Det kan sjekkes at generell API fungerer ved å velge Logitech Webkamera. Da vil det dukke opp feilmeldinger hvis f.eks. nightvision velges.			

T07 – Videokamera				
Test ID	**T07.007**			
Navn	Grensesnitt			
Beskrivelse	Man skal kunne koble seg til kameraets real-time videobilde.			
Risikoklasse	A1			
Kryssreferanser				
Krav ID	K07.007A			
Underkrav	Krav ID	Beskrivelse	Test ID	Klasse
Informasjon				
Testbeskrivelse	Black-box			
Testprosedyrer	T07.007	Verifiserer at kamera har oppkoblingsmuligheter mot videoserver.		
	Akseptkriteria	Kamera har grensesnitt som skal kunne koble seg til kameraets real-time videobilde.		
Test laget av	Carl Peder Jansen			
Kommentarer				

T10 - SYSTEM TEST	
Test ID	**T10.000**
Navn	System test
Beskrivelse	Fullskala test når hele systemet er koblet sammen.
Testsekvens	
Nr.	Beskrivelse
1	Det opprettes en server
2	3 klienter kobler seg opp mot server
3	En klient tar kontroll på kamera ved å trykke på CAMERA knappen i høyre meny. Her starter klientene ut ifra hvilken input enhet de har koblet til, rekkefølgen er: - Space navigator - Eye Tracker - MTCW
4	Verifiser at kameraets «rettigheter» blir låst til klient og at status oppdateres hos alle klienter.
5	Klient med prioritet på kamera velger fra venstre meny CAMERA-> Logitech webcamera
6	Trykk RECORD for å ta opp video.
7	Verifiser at opptaks status bytter fra «INACTIVE» til «ACTIVE» og at opptaks ikon er synlig.
8	Trykk RECORD igjen for å avslutte video opptak.
9	Verifiser at opptaks status bytter fra «ACTIVE» til «INACTIVE» og at opptaks ikon ikke er synlig.
10	Fra høyre meny velg ARKIV -> Video
11	Verifiser at video filen fra opptaket finnes og er lagret med riktig tid og dato. Deretter trykk på den aktuelle filen.
12	Verifiser at kort loader kommer opp før videoavspilleren blir åpnet.
13	Start video ved å trykke på play ikonet.
14	Pause video ved å trykke på pause ikonet. Verifiser at videoen kan spilles av, startes, stoppes og at tidslinjen oppdateres.
15	Trykk på live knappen (rødt ikon med kamera på)
16	Verifiser at loader kommer opp før det blir vist en live videostrøm.
17	Trykk på knappene SWIPE og NIGHTVISON.
18	Verifiser at error vindu kommer opp og viser riktig error.
19	Trykk på INFO knappen for å skru på info overlayet. Gå deretter ut av meny visningen.
20	Verifiser at overlayet blir vist og blir oppdatert ut ifra FPS, hvilken input enhet som er i bruk og forsinkelsen på videostrømmen(latency). Sjekk også at status blir oppdatert.
21	Bytt kamera ved å trykke CAMERA ->AXIS V5915.
22	Verifiser at den aktuelle videostrømmen blir vist.
23	Gjenta nå punkt 8 til 19 for dette kameraet.
24	Trykk på SWIPE.
25	Verifiser at swipe funksjonen blir satt i gang, og at status oppdateres hos alle klienter.
26	Ved swipets slutt, verifiser at status oppdateres hos alle klienter og at kameraet går tilbake til sentrert posisjon.
27	Trykk på SWIPE.
28	Verifiser at swipe funksjonen blir satt i gang, og at status oppdateres hos alle klienter.
29	Trykk på SWIPE for å stoppe underveis i swipet.
30	Verifiser at swipe funksjonen stopper ved at kameraet stopper i den posisjonen det var når det ble stoppet og at videooppptaket blir avsluttet. Verifiser også at status oppdateres hos alle klienter.

31	Trykk på NIGHTVISION knappen for å skru av og på nattmodus funksjonen.
32	Verifiser at IR cutfilter blir slått på i kamera og at status oppdateres hos alle klienter.
33	Trykk på INFO knappen for å skru av og på info overlayet.
34	Verifiser at overlayet blir vist og blir oppdatert ut ifra FPS, hvilken input enhet som er i bruk og forsinkelsen på videostrømmen(latency). Sjekk også at status blir oppdatert.
35	Trykk på CROSSHAIR knappen for å skru av og på trådkorset.
36	Verifiser at trådkors blir synlig/ikke-synlig og at status blir oppdatert.
37	Velg input enhet ved å trykke DEVICE -> (aktuelle device).
38	Gjenta punkt 8-19 og 25-33.
39	Bruk aktuell input enhet til å styre kameraet.
40	Verifiser at input enhetens tiltenkte funksjon fungerer som den skal.
41	Klient med prioritet på kameraet «kobler» seg fra kamera ved å trykke på CAMERA knappen i høyre meny.
42	Verifiser at status endres til «FREE» for alle klienter.
43	Gjenta fra punkt 3 til 42 helt til alle klienter med hver sin input enhet har gått igjennom test sekvensen.
44	System test ferdig.
Akseptkriteria	Systemets funksjonalitet skal dekke alle A krav.
Test laget av	Carl Peder Jansen
Kommentarer	

VEDLEGG

TIDLIGERE DOKUMENTHISTORIKK

Her er en oversikt over historikk som er gjort før versjon 2.0

Tabell 5 – Historikk V0.1 – V1.1.2

Versjon	Dato	Skrevet av	Beskrivelse
0.1.0	29.01.16	Svein	Sydd sammen testdokumentet.
0.1.1	30.01.16	Svein	Endret på sammendraget og lagt til underkrav og undertester i testspesifikasjonene for hvert enkelt krav. Punktet: Følgende test bortfaller er fjernet.
0.2.0	03.02.16	Carl Peder	Gjennomgang av krav i forhold til tester. Flere små feil som er fikset.
0.2.0	10.05.16	Carl Peder	Endret risikoklasse på T01.002.01 fra B2 til A1 Endret akseptkriteria for T01.001, T01.002, T02.001 og T03.001. Fra alle krav til kun A krav for godkjent test. Endret testprosedyre T01.001.04, fra rå videostrøm til komprimert H.264 videostrøm.
0.3.0	11.05.16		Lagt til systemtest T10.000
1.1.0	24.02.16	Carl Peder	Oppdatert hele testspesifikasjonen ut ifra endringene i kravspesifikasjonen etter møte med kunde/ekstern veileder.
1.1.1	04.03.16	Carl Peder	Oppdatert etter noen små endringer i kravspesifikasjonen. Stort sett endring av kravenes prioritet.
1.1.2	29.04.16	Carl Peder	Oppdatert test beskrivelse og krav prioritet etter endring av kravspesifikasjon 22.04. Slettet T03.009 og T03.001.06. Opprettet T03.010.
1.1.3	09.05.16	Carl Peder	Opprettet systemtest T10.000

REFERANSER

[1] - Olaf Hallan Graven, Veiledning for prosjekt 2016, 05.02.16

[2] - BachelorTangstad2015, tidligere bacheloroppgave, 05.02.16

[3] – UML IN A NUTSHELL: Sinan Si Albir (03.03.2016)

[4] – UML Distilled Third Edition: Martin Fowler (03.03.2016)

[5] – Schaum's Outline's UML: Simon Bennett, John Skelton (17.02.2016)

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

KOMMUNIKASJON – v 1.0

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENT 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIKK 3
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 KOMMUNIKASJON 5
3.1 MELDINGER 5
3.1 HVORDAN TOLKES EN MELDING 5
3.1.2 KLIENT 6
3.1.2.1 RESPONS PÅ CAMERA 6
3.1.3 SERVER 7
3.2 ERROR 7
4 STEP BY STEP 8
4.1 OPPKOBLING MOT SERVER 8
5 FUNKSJONER 9
6 PROBLEMER 10
6.1 SERVER OG KLIENT HENGER SEG 10
6.1.1 LØSNING VI GIKK FOR 10
6.1.2 ALTERNATIV LØSNING #1 10
6.1.2 ALTERNATIV LØSNING #2 11
6.2 FOR FÅ EGENSKAPER 11
6.2.1 LØSNING 11
REFERANSER 11

1 DOKUMENT

Dette dokumentet omfatter hvordan kommandoer blir tolket for systemet Real Time Observation.

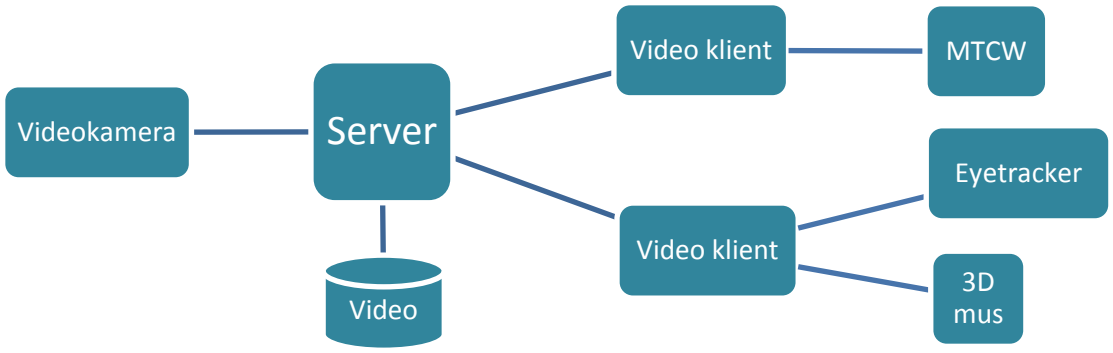
1.1 OVERSIKT

Hensikten med dette dokumentet er å få en oversikt over alle kommandoer som er akseptert av systemetet.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 – Systemoversikt

1.3 HISTORIKK

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
1.0	12.04	Vegard	Første utkast
1.1	11.05	Vegard	Nye meldinger

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 3
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4
Tabell 4- Kategorier 5
Tabell 5 - Eksempel på melding 5
Tabell 6 - Klient meldinger 6
Tabell 7 - Server meldinger 7
Tabell 8 - Error meldinger 7
Tabell 9 - Beskrivelse 8
Tabell 10 - Oppkobling mot server 8
Tabell 11 - Produksjon av melding 9

2.4 FIGURER OG BILDER

Figur 1 – Systemoversikt 3
Figur 2 - Continuous values 6

3 KOMMUNIKASJON

Det er blitt brukt TCP for kommunikasjon mellom server og klient. Dette er fordi det er en pålitelig måte å sende informasjon mellom to systemer. Qt har noen klasser som er beregnet for nettopp dette. Disse klassene er QTcpServer og QTcpSocket.

3.1 MELDINGER

Vi har to hovedkategorier. Disse kategoriene sier noe om hvem som har sendt meldingen. Det vil si, hvis en melding starter med 1:x:x så er det serveren som sender ut en melding. Den er enten en eller alle klientene. Hvis meldingen sterter med 2:x:x, da er det en klient som sender en melding til serveren.

Tabell 4- Kategorier

Kategorier	Hvem
1	Server
2	Klient

3.1 HVORDAN TOLKES EN MELDING

Klienten sender en QByteArray, som blir fanget opp av serveren. Serveren konverterer QByteArray'en om til en QString som igjen blir splittet basert på karakteren ':'. Denne informasjonen blir lagret i en QStringList. Systemet sjekker så opp hvilken kategori meldingen har, deretter hva den skal gjøre. De x siste tallene er parametere med informasjon som hjelper systemet som er gitt i «Hva» parameteret.

Meldingen *2:2:100:100:2* vil bli splittet opp som tabellen under viser. Denne meldingen vil sette kamera i posisjonen: X = 100, Y = 100, Z = 2. Hvor Z er zoom.

Tabell 5 - Eksempel på melding

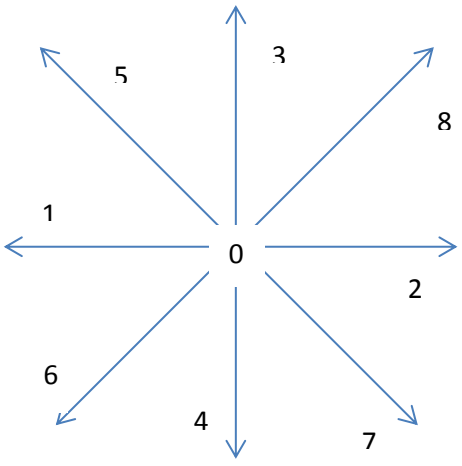
Kategori	Hva	Parameter1	Parameter2	Parameter3
2	2	100	100	2

3.1.2 KLIENT

Tabellen under viser hvilke meldinger klienten kan sende. Det er ikke alle meldingene som får en respons fra serveren.

Tabell 6 - Klient meldinger

Melding	Beskrivelse	Verdier	Respons
2:1:0:0:0	Sender forespørsel om å bruke kamera.		1:4:(int)
2:1:1	Sender forespørsel om å bruke swipe.		
2:1:2	Sender forespørsel om å bruke recording		
2:2:(int)x:(int)y :(int)z	Setter x,y,z verdier på kamera. Dette er absolutt posisjon	X: -180 – 180 Y: -180 – 180 Z: 1 - 9999	Se avsnitt 3.1.2.1
2:5:(int)direction	Continuous move i bestemt retning	0: Stop 1-8: Se figur 2	
2:7:(int)x:(int)y	Setter continuouse move med fart	X: -180 – 180 Y: -180 – 180	
2:8:(int)z	Setter zoom	Z: 1-9999	
2:9:(int)x:(int)y:(int)z	Continuouse move	X: -180 – 180 Y: -180 – 180 Z: 1 - 9999	
2:10:(int)	Setter mode på kamera	1: Default 0: IR Cutfilter	



Figur 2 - Continuous values

3.1.2.1 RESPONS PÅ CAMERA

Hvis kameraet sitt API støtter funksjonen, vil den gå gjennom uten at serveren sender en respons tilbake. Hvis kameraet sitt API ikke har en støttende funksjon, så vil serveren sende en Error melding tilbake.

3.1.3 SERVER

Tabell 7 - Server meldinger

Kommando	Beskrivelse	Verdier
1:0:0	«Request» update on server.	
1:1:(int)ErrorID	Sier hvilken error som har oppstått	
1:2:(int)TotalClients	Sier hvor mange klienter som er koblet til server	
1:3:(int)MyID	Sier hvilken ID du har blitt gitt	
1:4:(int)ClientUsingCam	Sier hvilken klient som bruker kamera	
1:5:(int)camera	Sier hvilket kamera som valgt	
1:6:		
1:7:(int)clientSwipe	Sier hvilken klient som bruke swipe	
1:8:(int)clientRecord	Sier hvilken klient som recorder	
1:9:		
1:10:(int)cameraMode	Sier hvilken mode kamera er satt til	

3.2 ERROR

Dette er de error meldingen som er støttet av systemet.
Error meldinger med ID 1 til 3 er globale erroer. Det vil si, meldinger som blir sendt til alle klientene.

Errorene med ID 10 og utover, er error meldinger som blir trigget av CameraRTO. Det vil si hvis et kamera ikke har støtte for gitt funksjon, så vil det bli sendt en error melding til klienten som prøver å styre kamera med gitt funksjon.

Tabell 8 - Error meldinger

ErrorID	Beskrivelse
1	Serveren har avsluttet
2	Error 2
3	Error 3
10	
11	setPosition is not available for this camera
12	setContinuousPosition is not available for this camera
13	setContinuousMove is not available for this camera
14	setMoveWithSpeed is not available for this camera
15	setStream is not available for this camera
16	setBrightness is not available for this camera
17	setZoom is not available for this camera
18	setMode is not available for this camera
19	autoSwipe is not available for this camera
20	There is no camera with that ID

4 STEP BY STEP

Gjennomgang av hvordan kommunikasjonene forgår.

Den mest kritiske fasen er når en klient kobler seg opp mot serveren. Da er det mye som skjer og det er ikke alltid klienten får med seg alt som skjer. Det er mulig at klienten ikke fanger opp alle meldingene som blir sendt til den.

For å øke sjansen for at serveren henger med i svingene har vi lagt inn et par «thread.sleep(200)» på server siden, før en melding blir sendt. Dette er ikke en god løsning. Men den virker og siden vi har et så lite system har vi valgt å beholde denne løsningen.

Tabell 9 - Beskrivelse

CLIENT	SERVER

Tabellen over viser hvilken farge klient og server har videre i dokumentet.

4.1 OPPKOBLING MOT SERVER

Tabell 10 - Oppkobling mot server

REQUEST	RESPONS
«Kobler seg opp»	1:3:(int)MyID 1:4:(int)ClientUsingCam

Når en klient kobler seg opp mot en server, så vil serveren sende tre svar. Første melding fra server inneholder hvilken ID klienten har fått tildelt. Deretter får klienten en melding som inneholder informasjon om hvem som bruker kamera, deretter får den en tredje melding som inneholder hvor mange klienter som er koblet på serveren.

De to første meldingene er sendt til klienten som nylig har koblet seg til. Mens den siste meldingen blir sendt til alle klientene.

5 FUNKSJONER

Tabell 11 - Produksjon av melding

Funksjon	Produserer melding
cameraSetPosition(int, int, int)	2:2:x:y:z
cameraSetContPos(int, int, int)	2:9:x:y:z
cameraSetContinuousMove(int)	2:5:direction
cameraSetStream()	Brukes ikke
cameraSetBrightness(int)	Brukes ikke
cameraSetAutoSwipe(int)	Brukes ikke
cameraSetMoveWithSpeed(int,int)	2:7:x:y
cameraSetZoom(int)	2:8:z
cameraSetMode(int)	2:10:value

6 PROBLEMER

6.1 SERVER OG KLIENT HENGER SEG

Når kommandoen ”readyRead()” blir trigget av QTcpSocket på både server siden (ClientSocket) og ClientTCP. Så går meldingen inn i en funksjon som konverterer meldingen fra QByteArray til en QString Deretter så blir meldingen splittet opp og håndtert deretter.

La oss si vi sitter på en klient. Denne klienten har en enhet som skal styre kameraet. For å styre kameraet så må vi sende informasjon fra klienten til serveren.
En enhet sin posisjon blir oppdatert konstant. Hver gang enheten blir oppdatert, så er serveren interessert i den nye posisjonen slik at den kan oppdatere kamera sin posisjon. I teorien så vil en konstant strøm av meldinger som oppdaterer kameraet sin posisjon være ideelt, slik at brukeren får en naturlig følelse av at han faktisk styrer kameraet.

Det som skjedde da vi prøvde dette, var at serveren ikke greide å holde følge med antall meldinger som ble sendt fra klienten. Dette førte til at klienten falt ”etter”. Det vil si at serveren jobbet med en melding som skjedde for f.eks 10x meldinger siden (fra klienten sin side).

Samme problem skjer om serveren sender for mange meldinger fort til en klient.

6.1.1 LØSNING VI GIKK FOR

For å forhindre at klienten hang etter, så la vi inn en QTimer(klokke) på klienten. Klokkens intervall ble satt til 200ms. Når en enhet ble valgt, så ble klokken startet. Hver gang klokken ble trigget(hvert 200ms), ble enhetens posisjon sendt til serveren, slik at den kunne oppdatere kameraet.

En lignende løsning ble laget på serveren sin side. Hver gang serveren skulle sende ut ”mye” informasjon ble det lagt inn en thread.sleep(200). Det vli si at serveren ”sover” i 200 ms.

Denne løsningen funker greit til vårt formål. Siden vi har litt forsinkelse på videobildet, så legger vi ikke så veldig merke til denne forsinkelsen på enhetene. Men om vi skulle utviklet systemet lengre så ville en alternativ #2 metode vært bedre.

6.1.2 ALTERNATIV LØSNING #1

En annen metode som kunne bltt implementert, kunne vært å lage en buffer. Hver gang ”readyRead()” ble trigget så skulle en buffer bli trigget. Denne bufferen legger til ny informasjon inn i en liste. Når det er gjort, så sjekker den om det er mer informasjon på nettverket. Hvis ja, blir den nye informasjonen lagt inn i bufferen. Når det ikke er mer informasjon på nettverket, så blir melding for melding ”løst” av serveren.

Det som kan være problem med dette er at vi låser bufferen ved å sende meldinger konstant hver gang en enhet endrer posisjon. På den måten så kommer ikke serveren til det stadiet hvor den skal behandle og utføre handlingene som ligger lagret i bufferen.

6.1.2 ALTERNATIV LØSNING #2

Den beste metoden vill vært å lage en buffer som blir lagt i en tråd. Den tråden blir behandlet kontinuerlig samtidig som bufferen legger til nye meldinger som eventuelt har kommet.

6.2 FOR FÅ EGENSKAPER

Systemet vårt kan håndtere et kamera. Om en klient velger å ta kontroll over kameraet, utføre en swipe eller ta opp videostrøm, så vil alle klientene bli oppdatert med uavhengig av hvilket kamera de faktisk bruker.

Situasjon
La oss si vi har tre klienter som er koblet opp mot en server.

Klient 1: Har valgt Axis V5915
Klient 2: Har valgt Logitech
Klient 3: Har valgt Axis V5915

Hvis klient en velger å ta kontroll over kameraet, så blir det sendt ut informasjon fra server til *alle* klienten om at klient 1 bruker kameraet. Dette er ugunstig for klient 2 som ikke har noe med det kameraet å gjøre. Samme problemstilling oppstår om klient 1 velger å utføre en ”swipe”, begynne å ta opp eller endre mode. I alle situasjonene så får klient 2 informasjon som ikke omhandler sitt kamera.

6.2.1 LØSNING

Det kan enkelt bli lagt til noen variabler i ClientSocket som sier noe om hvilket kamera klienten bruker. Så kan serveren sjekke denne variabelen og sende riktig informasjon til riktige klienter.

REFERANSER

[1] - Vapiz_ptz_52933_en_1307.pdf

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

ETTERANALYSE – v 1.0

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENT 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIKK 4
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 PROSJEKTRESULTAT 5
3.1 KRAV OG TESTING 5
3.2 EVALUERING AV PRODUKTET 6
3.2.1 DESIGN AV REAL TIME OBSERVATION APPLIKASJONEN 6
3.2.2 KOMMUNIKASJON MELLOM SERVER OG KLIENT 6
3.2.3 VIDEOSTRØM 6
3.2.4 MMI-ENHETER 7
3.2.5 ANNEN FUNKSJONALITET 7
4 PROSJEKTGJENNOMFØRING 8
4.1 ARBEIDSMETODER 8
4.2 SAMARBEID 8
4.2.1 I PROSJEKTGRUPPEN 8
4.2.2 OPPDRAGSGIVER, EKSTERN VEILEDER 8
4.2.3 INTERN VEILEDER 8
4.2.4 SYKDOM/FRAVÆR 8
4.3 PROSJEKTADMINISTRERING 9
4.3.1 ANSVARSFORDELING 9
4.3.2 MØTER 9
4.4 TIMEBRUK 9
5 VEIEN VIDERE 11
6 KONKLUSJON 11
REFERANSER 11

1 DOKUMENT

Dette dokumentet omfatter en etteranalyse for systemet Real Time Observation.

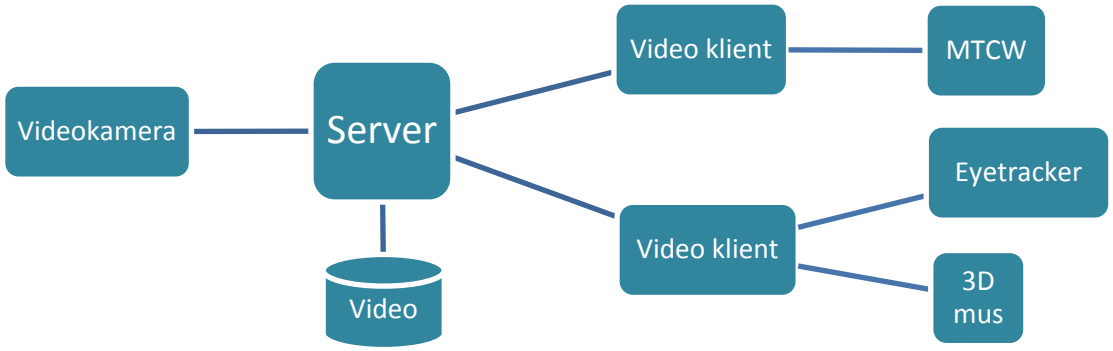
1.1 OVERSIKT

Denne analysen er gjort etter at prosjektet er fullført. Analysen tar for seg hva vi har oppnådd, hva som i hovedtrekk har vært våre utfordringer og hva vi har lykkes med i prosjektet. Det er alltid deler av et prosjekt som både har gått bra og mindre bra. Vi vil forsøke å få frem alle deler i dette dokumentet.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 – Systemoversikt

1.3 HISTORIKK

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
1.0	18.05.2016	Alexander Schinnes	Første versjon av dokumentet

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 4
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4

2.4 FIGURER OG BILDER

Figur 1 – Systemoversikt 3
Figur 2 - Estimerte timer vs. faktiske 10
Figur 3 - Timer brukt pr person 10

3 PROSJEKTRESULTAT

Gjennom arbeidet vi har lagt ned i oppgaven har vi nådd de viktigste målene og implementert alle A og B hovedkravene i tillegg til noen C krav. Vi vil her se litt nærmere på hva vi har oppnådd.

3.1 KRAV OG TESTING

Vi har generelt sett hatt mange krav og underkrav i vårt prosjekt. Alle krav vi mente var viktige for at systemet skulle fungere tilfredsstillende ble satt som A krav. Krav vi mener er en viktig del av systemet, men ikke avgjørende for funksjonaliteten ble satt som B krav. Alt resterende av ønsket funksjonalitet som vil forbedre systemet, men som ikke er avgjørende for måloppnåelsen ble satt som C krav.

Vi startet med å finne ut av hva kunden ønsket at produktet skulle kunne gjøre. Ut ifra dette skrev vi kravene. Her var vi stort sett på rett spor hele veien, men fikk samtidig oppklart flere misforståelser underveis i prosessen rundt hvordan arkitekturen og systemet skulle fungere. Dette var misforståelser som hadde påvirkning på implementeringen av produktet. Vi fikk heldigvis klarhet i dette før vi påbegynte implementeringen, slik at det ikke gikk mye tid til spille.

Etter at visjonen og kravene for systemet var blitt satt, var det på tide og finne ut hvordan vi ønsket å løse de forskjellige utfordringene. Her var det mye nytt stoff for oss å sette oss inn i fordi vi hadde begrenset erfaring med videostrøm og nettverkskommunikasjon fra før, dermed gikk det med en del tid til research og testing den første tiden.

Gjennom «elaboration» fasen fokuserte vi på å få en stabil arkitektur for dernest å implementere noen viktige punkter som kommunikasjon og sløyfe for videostrøm fra server til klient. Dette er to elementære biter av systemet som må være på plass for at resten av funksjonaliteten skal fungere. Vi var relativt tidlig ute med å få på plass en sløyfe for flyt i systemet fra server til klient. Den var langt i fra feilfri, men gav oss et godt grunnlag å jobbe videre ut ifra.

I «construction» satte vi opp en enkel GUI med basisfunksjoner. Det ble ikke fokusert på design og feilfrie funksjoner de første ukene, men heller å være innom alle A kravene og få disse til å fungere slik at vi hadde noe å vise til. Når denne biten falt på plass og vi hadde ønsket funksjonalitet begynte vi å fokusere på å optimalisere systemet. Her ble det fikset mange bugs og GUI fikk en overhaling til nytt design. Dette viste seg å være et smart trekk slik at vi ikke satt igjen med noen krav som ikke var oppfylt mot slutten av prosjektet.

3.2 EVALUERING AV PRODUKTET

Det som kjennetegner produktet vårt kan kort oppsummeres med:

- Allsidig overvåkningssystem som kan benyttes av flere brukere og forskjellige målgrupper.
- API for styringsenheter slik at nye MMI-enheter enkelt kan implementeres i systemet senere.
- Godt brukervennlig design som ikke er komplisert å bruke.
- Jevn synkroniseringen av data mellom server og klienter.
- Mulighet for å styre kameraposisjon på flere ulike måter ved hjelp av ønsket MMI-enhet.
- Smart system som deaktiverer/aktiverer funksjonalitet i forhold til hvilken MMI-enhet som er valgt og tilkoblet.

3.2.1 DESIGN AV REAL TIME OBSERVATION APPLIKASJONEN

Det som har vært viktig for både oss og KDA er at systemet skal være fungerende slik at man får testet ut systemet når det er ferdig. For at systemet skal kunne testes fullt ut er det viktig at designet tillater dette og er enkelt å bruke. I den siste iterasjonen har vi hatt spesielt fokus på design og brukervennlighet. Dette har ført til en langt bedre opplevelse for brukeren enn det de første versjonene kunne tilby. Verktøy vi har benyttet for design er i hovedsak Visual Studio, Photoshop, Illustrator og Qt-Quick.

3.2.2 KOMMUNIKASJON MELLOM SERVER OG KLIENT

Fordi produktet skal ha støtte for flere klienter tilkoblet til en server er det viktig at datakommunikasjonen mellom disse foregår sømløst. Vi har laget kommunikasjonen i Qt ved hjelp av QTcpServer og QTcpSocket. Her utveksles all informasjon mellom klient og server med unntak av videostrømmen. Klient kan se antall tilkoblede brukere av systemet i real-time og kan se hvilken bruker som benytter seg av hvilket kamera og lignende. I tillegg sendes all informasjon om posisjonering av kamera ved hjelp av MMI-enhetene seg over dette nettverket. Dette har vært et stort og viktig element å få på plass med mange utfordringer underveis, men vi har fått til en godt fungerende kommunikasjon. Eneste feil er at serveren enkelte ganger ikke fanger opp meldinger hvis de kommer for tett på hverandre. Vi har derfor lagt inn en liten forsinkelse mellom hver melding for å forhindre dette.

3.2.3 VIDEOSTRØM

Klientene skal kunne få overført videobildet fra kameraet via serveren. Det som er viktig her er å ende opp med en minst mulig end-to-end forsinkelse fra kamera til klientskjerm. Etter mye testing og research fant vi ut at en stor flaskehals for dette er forsinkelsen som oppstår i Picolo videokortet. Hardware komprimeringen her fører med seg en forsinkelse i videostrømmen på 400 ms. I tillegg er det rundt 40ms forsinkelse på HDMI-utgangen fra kameraet. Kravet vi hadde om forsinkelse på 200 ms har derfor ikke blitt oppfylt. Ved å hente ut videostrømmen fra videokortet og sende denne videre til klienter over nettverket endte vi opp med 1250 ms forsinkelse. Dette gjorde at opplevelsen av systemet og responsen for styring av kameraet føltes veldig tregt. Etter avtale med KDA bestemte vi oss derfor for å hente ut videoen rett fra H.264 nettverksstrømmen som kameraet leverer. Dermed

sitter vi nå igjen med en forsinkelse på 550 ms, som gir en bedre respons. Videokvaliteten er også meget god. Mer detaljert info rundt dette er gitt i analysedokumentet.

3.2.4 MMI-ENHETER

Vi har fått til å integrere og implementere alle enhetene som KDA ønsket. Applikasjonsdriver til MTCW er laget, i tillegg til at det er implementert styring av kamera for alle enhetene. Det har gått med en del tid for å finne gode metoder å styre kameraet på. Vi har funnet ut både positive og negative sider ved å bruke de ulike enhetene til å styre kameraet. Mer om dette i analysedokumentet.

I begynnelsen hadde vi et krav fra KDA at MMI-enhetene skulle integreres inn i en felles ren device-API, slik at det skulle være enkelt å legge til nye enheter senere. Vi forsøkte å oppfylle dette, men fant ut at det kanskje ikke var den beste måten å løse det på. De enhetene vi har fått utlevert har veldig lite til felles med hverandre, noe som gjør det lite effektivt å knytte disse sammen gjennom en felles API. En eyetracker oppfører seg på en helt annen måte enn for eksempel en 3D mus. I samarbeid med ekstern veileder droppet vi derfor å lage en felles API. Vi gikk for å implementere de ulike MMI-enhetene som selvstendige komponenter i stedet. De ulike enhetene kaller da på funksjoner som er utviklet i klienten for å styre kamera. Dette gjør at enhetene kan bli styrt og implementert på ulike måter, men allikevel bruker de samme funksjonene uten at disse trenger å endres.

3.2.5 ANNEN FUNKSJONALITET

For å vise noe av potensialet til systemet har vi lagt inn en del funksjonalitet for brukeren. Som en oppsummering kan vi nevne:

- Opptak av video fra hvilket som helst kamera.
- Mulighet for å hente opp og avspille video i en videoavspiller.
- Nightvision/IR modus kan settes på kamera.
- Swipe er en funksjon som automatisk panorerer kameraet fra -170 grader til +170 grader samtidig som det blir gjort opptak.
- Crosshair for illustrasjon av eventuelle muligheter for å sikte seg inn på ulike objekter.
- Mulighet for å vise info i et overlay med FPS verdi, forsinkelse og hvilken enhet som er i bruk.

4 PROSJEKTGJENNOMFØRING

4.1 ARBEIDSMETODER

Som arbeidsmodell og utviklingsmetodikk valgte vi å bruke Unified Process. Dette har fungert meget bra for oss, og har vært en god modell å følge i forhold til oppgaven. Vi valgte denne modellen fordi vi var usikre på hvilke løsninger vi skulle velge og hvor lang tid ting ville ta underveis i prosjektet. Unified Process er en modell som gjør det oversiktlig å jobbe selv om oppgaven kan virke noe uforståelig og løsningene er uklare. I begynnelsen brukte vi mye tid på å sette oss godt inn i oppgaven og være konsise når blant annet krav ble definert. Underveis har vi hatt stor nytte av å kunne ha bestemte fokusområder å følge i hver iterasjon. Da hadde vi en viss tid til å få ting ferdige og det var lett å oppdage hvis ting ikke gikk etter planen. Vi har hatt god kontroll på tidsbruk og arbeidsmengde hele veien slik at vi aldri har blitt liggende etter.

4.2 SAMARBEID

4.2.1 I PROSJEKTGRUPPEN

Vi har hatt stort fokus på å arbeide jevnt gjennom hele perioden fra jul til presentasjon av prosjektet i juli. Alle har møtt opp og gjort arbeidsoppgavene som har blitt avtalt. Vi har i hovedsak arbeidet fra 08-16 tre dager i uken frem til påske. Etter eksamen i april har vi arbeidet fem fulle arbeidsdager hver uke. Dette har gjort at vi alltid har ligget bra an tidsmessig slik at vi ikke har trengt å gjøre noen store skippertak de siste ukene. Som i alle prosjekter har vi møtt utfordringer med tanke på sykdom og lignende. Ved hjelp av godt motiverte gruppemedlemmer har dette blitt løst på en god måte ved at arbeid har blitt delt ut eller at andre medlemmer har overtatt oppgaver. Gruppen har vært samlet og i full drift hele tiden.

4.2.2 OPPDRAGSGIVER, EKSTERN VEILEDER

Våre eksterne veiledere, Arnt-Levi Alstad og Rune Værnes, har vi hatt et godt samarbeid med hele veien. De har vært tilgjengelige og det har blitt ført en åpen dialog hele veien om de ulike utfordringene vi har hatt og om generell fremdrift. De har også tilbudt teknisk hjelp der det har vært behov for dette.

4.2.3 INTERN VEILEDER

Joakim Bjørk har vært vår interne veileder fra skolens side. Vi har stort sett hatt ukentlige møter hvor vi har gitt info om status og hvordan planen er videre for prosjektet. Det har også vært lett å stille spørsmål vi har hatt rundt dokumentasjon, prosess, modell, presentasjoner og lignende. Dette har fungert bra.

4.2.4 SYKDOM/FRAVÆR

Vi har generelt hatt lite sykdom som har ført til fravær i gruppen. Carl Peder var syk under andre presentasjon, men dette ble løst ved at resterende gruppemedlemmer fordelte hans del av presentasjonen. Det har også vært noe fravær i gruppen som følge av deltidsjobber, jobbintervjuer og lignende, men dette anser vi som helt normalt. Arbeidet har blitt godt fordelt.

4.3 PROSJEKTADMINISTRERING

Vi har benyttet standard verktøy for prosjektadministrering som Word, Excel og Microsoft Project. Ellers har mye av den interne kommunikasjonen foregått på egen gruppe på Facebook. Dette har fungert bra, og vi har ikke hatt noen spesielle problemer.

4.3.1 ANSVARSFORDELING

På et av de første møtene vi hadde internt i prosjektgruppen, fordelte vi roller. Dette var roller som prosjektleder, systemarkitekt, dokumentasjon og test ansvarlig. For strukturen sin del og for å sikre at disse områdene ble fulgt opp var det en fornuftig måte gjøre det på. Det var ikke meningen at den ansvarlige skulle utføre alt innenfor sitt område, men han skulle følge opp at det ble gjort. Dette har stort sett fungert bra. Vi har også fokusert på at alle har vært innom de fleste sider av systemet.

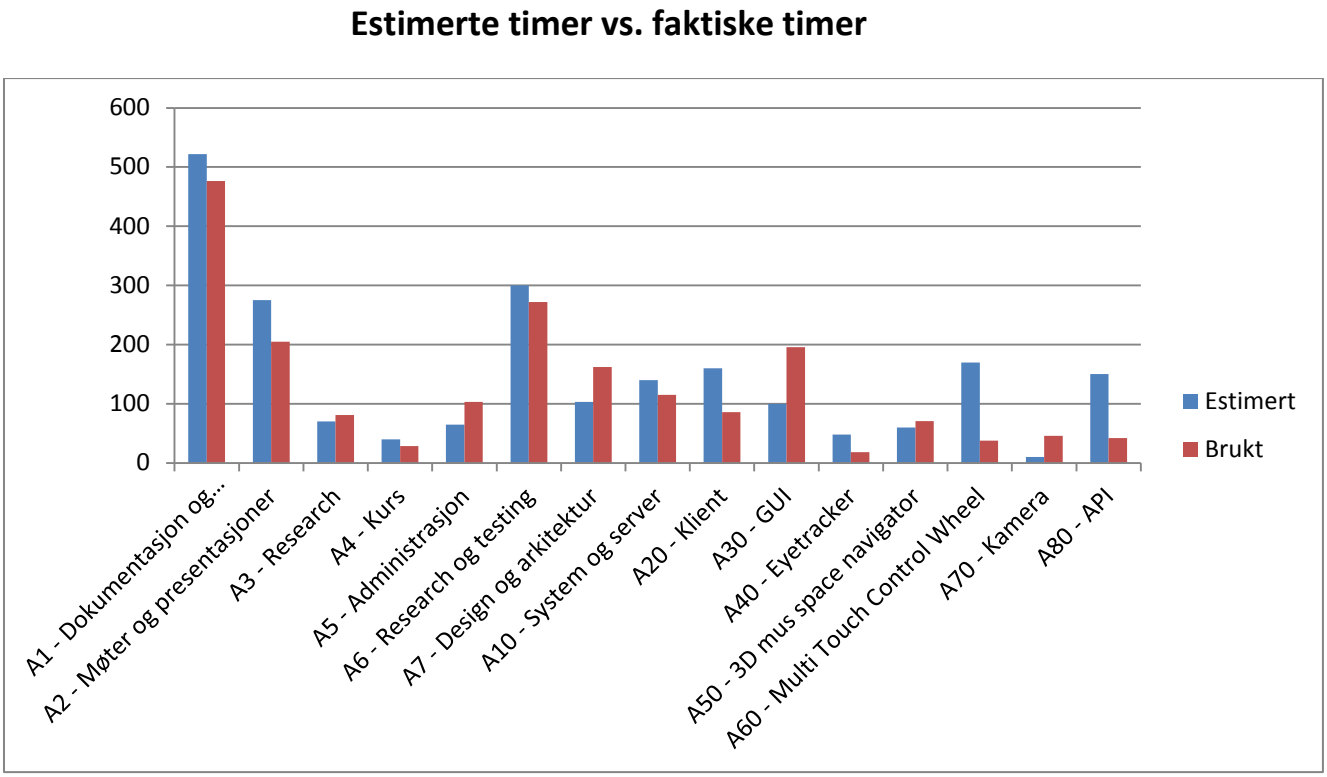
4.3.2 MØTER

Vi har hatt jevnlige møter både innad i gruppen, med ekstern veileder og intern veileder. Med veiledere har vi hatt møter med 1-2 uker mellomrom, mens vi har hatt daglige Scrum møter internt i prosjektgruppen. Vi har hatt fast møteleder og referent, noe som har gjort at møtene har gått bra fordi hver enkelt blir drillet på sitt felt.

4.4 TIMEBRUK

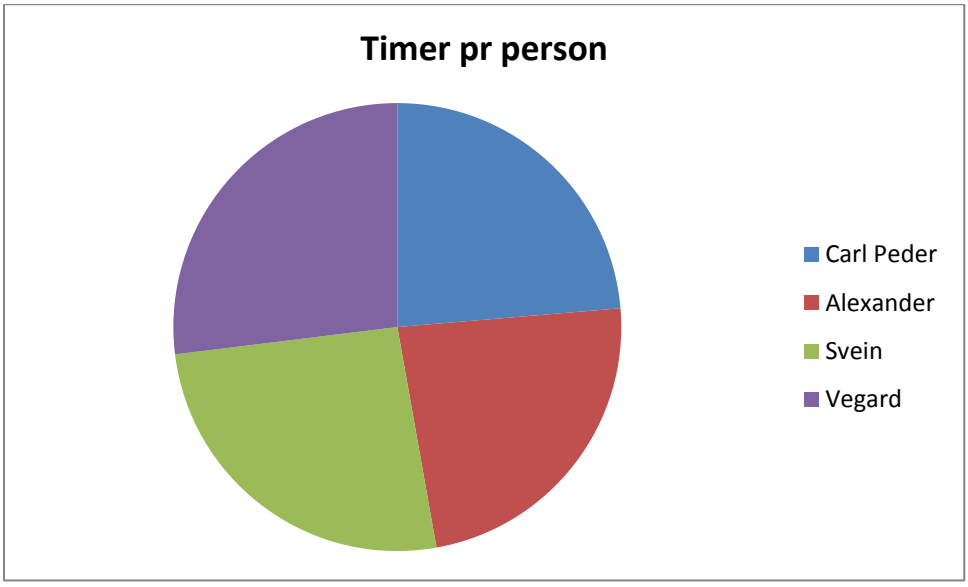
For å registrere timer har vi benyttet Excel. Sett i etterkant er dette noe vi med fordel kunne unngått. Det har blitt noe rot underveis, og det er mer krevende å holde aktiviteter og timelister oppdatert i Excel enn i et prosjektverktøy som er beregnet kun til dette. Vi ser også at estimeringen av tidsforbruk per oppgave har vært noe ute å kjøre til tider fordi vi ikke alltid visste hvordan vi skulle løse de ulike oppgavene før de ble påbegynt. Derimot har alle arbeidsoppgaver blitt løst innenfor hver iterasjon. Fordelingen i tidsbruk per person er relativt jevnt, men noen forskjeller blir det naturlig nok. Vi vil ha mer konkret oversikt over tidsforbruk når alt arbeid og dokumentasjon er unnagjort, men vi forventer å ende opp med rundt 600 timer per person.

På neste side ser vi en figur som viser hvor mye tid vi hadde beregnet å bruke på hver hovedkategori og hvor mye tid vi faktisk har brukt. Her ser vi at vi stort sett har vært i nærheten av faktisk tid når vi estimerte hvor lang tid hver aktivitet ville ta. Men noen unntak er det blant annet når det kommer til GUI, Multi Touch Control Wheel og API. På GUI har vi brukt nærmere dobbelt så mange timer i forhold til hva vi hadde beregnet. Dette kommer på grunn av at vi valgte å gi den en relativt stor overhaling mot slutten av prosjektet. MTCW og API har vi brukt mindre tid enn vi hadde beregnet da driver var enklere å lage enn vi hadde trodd, og en API for devicene ble droppet.



Figur 2 - Estimerte timer vs. faktiske

Under vises en oversikt over den faktiske timefordelingen i gruppen. Her ser vi at det er brukt relativt likt antall timer, men med noen forskjeller.



Figur 3 - Timer brukt pr person

5 VEIEN VIDERE

Gjennom den tiden vi har hatt til rådighet for å lage produktet, har vi fått oppfylt det som har vært kravet for systemet. Det skulle være en slags prototype for å teste overføring av H.264 video fra kamera til en server som distribuerer videostrømmen videre til ulike klienter. Samt at de ulike MMI-enhetene skulle kunne styre dette kameraet via serveren. Det skulle også være mulighet for å lagre og spille av videoen igjen på et senere tidspunkt.

Alt dette har vi løst og fått til å fungere, men vi har også noen tanker rundt forbedringer og videre utvikling av systemet. Under følger en kort oversikt over det vi har tenkt på:

- Når eyetracker velges som device så styres denne med ulike taster på tastaturet. Det bør dukke opp info om hvilke hotkeys som gjelder og hva disse gjør i GUI så lenge eyetracker er valgt (G,H,B,J,I og O).
- Hvis rektanglene til eyetrackeren er framme for å styre kamera når det velges en ny device, bør disse forsvinne. Det gjør de ikke nå.
- Tags, screenshot og database er noe vi ikke har fått tid til som har vært C-krav. Dette hadde vært fint å få på plass.
- Byttet ut hardwareløsningen for å få ned forsinkelsen på videostrømmen.
- Øke kapasiteten på datamengden (TCP meldinger) serveren tåler. Det kan lugge litt hvis det kommer for mange meldinger på kort tid.
- Sett på flere bruksområder enn bare å styre kamera for MMI-devicene. For eksempel å bruke eyetracker til å åpne meny og lignende. Evt. å kombinere bruk av devicene bedre.

6 KONKLUSJON

Selve produktet vårt er nå ferdig og det gjenstår kun noe avsluttende dokumentasjon og fremførelse av tredje og siste presentasjon. Vi har utviklet det vi skulle og er enige med oppdragsgiver om at det har vært et godt gjennomført prosjekt. All funksjonalitet er på plass og fungerer tilfredsstillende. Det er noen småfeil og ting vi ser kunne vært gjort bedre, men det er å regne med ut ifra den korte tiden vi har hatt til å utvikle systemet. Vi har hatt utfordringer underveis og banket hodet i veggen, men vi har også fått ting til å fungere og vist at samarbeidet har fungert godt både innad i gruppen og med andre involverte parter. Selv om det til tider har vært vanskelige og krevende utfordringer har vi taklet disse bra, og sitter nå igjen med et produkt som vi er meget stolte av.

REFERANSER

[1] – Bacheloroppgave Consultant Calendar Overview (2013).
Kilde – Bibliotek Brage Kongsberg

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

TESTRAPPORT – v 1.1

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENT 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIKK 4
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 INTRODUKSJON 5
4 TESTOVERSIKT 6
5 KONKLUSJON 7
6 VEDLEGG 1: TESTLOGG 8
REFERANSER 27

1 DOKUMENT

Dette dokumentet inneholder en rapport for testing av alle krav for systemet RTO.

1.1 OVERSIKT

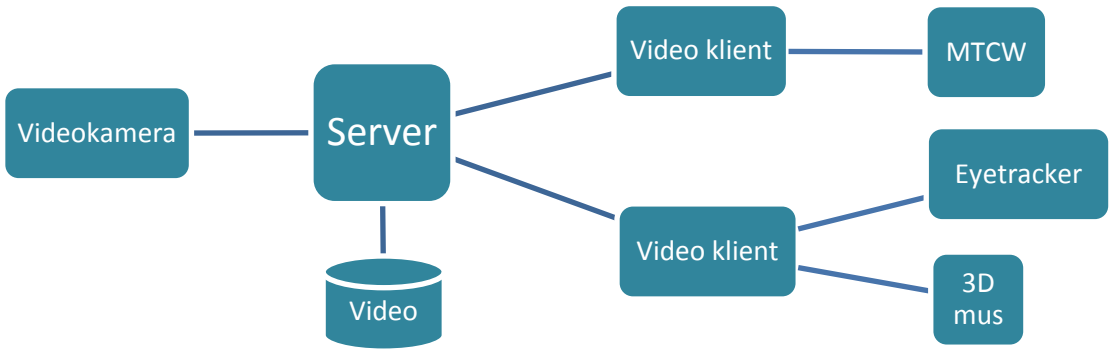
Dokumentet er satt opp med en oversikt over hvilke tester som er gjennomført, statusen på disse og om de er godkjente eller ikke. Hver test har da kryssreferanser til kravene for å sikre en god sporbarhet.

Videre er det en logg med mer detaljert informasjon om hver test og undertester.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 – Systemoversikt

1.3 HISTORIKK

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
1.0	19.05.2016	Carl Peder Jansen	Første utkast
1.1	19.05.2016	Alexander Schinnes	Korrektur og retting

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 4
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4

2.4 FIGURER OG BILDER

Figur 1 - Systemoversikt 3

3 INTRODUKSJON

Vi tester etter «bottom-up» prinsippet, det betyr at vi først tester hvert enkelt krav for seg selv i det vi har kalt for en deltest, deretter når mer av systemet begynner å komme på plass tester vi hele systemet i en systemtest som fokuserer mer på hvordan hele systemet fungerer som en komplett modul.

Deltestene begynner vi å gjennomføre hver fredag når vi kommer til det punktet at vi har noe som kan testes. I disse testene er det svært viktig at vi ikke tester vårt eget arbeid, men at hver enkelt tester hverandres arbeid. Hvis en deltest er godkjent vil ikke denne bli testet flere ganger. Systemtesten vil bli gjennomført når vi ser at alle A krav i deltestene er godkjent. Hvis ikke systemtesten blir godkjent må feilene dokumenteres og fikses, det må så gjennomføres en ny systemtest.

4 TESTOVERSIKT

Test ID	Krav ID	Beskrivelse	Utført av	Status
T01.001	K01.001A	System og server - Videostrøm	Carl Peder Jansen	
T01.002	K01.002A	System og server - Lagring	Alexander Schinnes	
T01.003	K01.003A	System og server - Videokort	Alexander Schinnes	
T01.005	K01.005A	System og server – Analyse av videostrøm	Alexander Schinnes	
T02.001	K02.001A	System og server - Klient - Kommunikasjon	Alexander Schinnes	
T03.001	K03.001A	GUI - Meny	Carl Peder Jansen	
T03.002	K03.002B	GUI - Play/Pause	Vegard Smedsvik	
T03.003	K03.003B	GUI - Tidslinje	Vegard Smedsvik	
T03.004	K03.004C	GUI - Spole	Ikke testet	Ikke testet
T03.005	K03.005A	GUI - Live knapp	Vegard Smedsvik	
T03.006	K03.006C	GUI - Tags	Ikke testet	Ikke testet
T03.007	K03.007C	GUI - Skjermdump	Ikke testet	Ikke testet
T03.010	K03.010A	GUI - Integrering av input enheter	Carl Peder Jansen	
T04.001	K04.001A	Eye Tracker - Deteksjon	Carl Peder Jansen	
T05.002	K05.002A	Space Navigator - Posisjon	Alexander Schinnes	
T06.001	K06.002A	MTCW - Driver	Vegard Smedsvik	
T06.007	K06.007A	MTCW - Grensesnitt	Carl Peder Jansen	
T07.001	K07.001A	Videokamera - Oppløsning	Svein Bekkevold	
T07.002	K07.002A	Videokamera - Bildefrekvens	Svein Bekkevold	
T07.005	K07.005A	Videokamera - Motorisert	Svein Bekkevold	
T07.006	K07.006A	Videokamera - API	Svein Bekkevold	
T07.007	K07.007A	Videokamera - Grensesnitt	Svein Bekkevold	
T10.000		Systemtest	Samtlige i gruppen	

5 KONKLUSJON

All testing er unnagjort og vi mener vi har et godt system. Alle A krav er godkjent bortsett fra kravet om video forsinkelse(latency) mindre enn 200ms. Dette har vært avklart med oppdragsgiver tidligere i prosjektet så dette er vi komfortable med. Det er også noen B og C krav som har blitt nedprioritert på grunn av tidsmangel, disse er derfor ikke blitt testet.

Deltestingen som ble utført hver fredag fra 22.04 har vist seg å være svært verdifulle da disse ga oss en god oversikt over hvordan vi lå an i forhold til kravene. Ved at vi alltid testet hverandres arbeid bidro disse testene til å finne flere feil som den som skrev koden ikke hadde lagt merke til selv.

Systemtestingen fungerte også godt. Vi måtte gjennomføre tre komplette systemtester før systemet var ansett som godkjent. Det som var en gjennomgående feil i disse testene var data kommunikasjonen mellom klienter og server. Dette ga en følgefeil som også påvirket resten av systemet. Når dette da ble fikset var det kun snakk om å rette opp i noen små feil før systemet fungerte som ønsket og testen ble ansett som godkjent.

6 VEDLEGG 1: TESTLOGG

DELTEST

Dato: 29.04.2016

Hvor: Høgskolen i Sørøst-Norge

T01 – System og server			
Test ID	Beskrivelse	Klasse	Krav ID
T01.001	Skal kunne motta video fra videokort og videresende strømmen til klient.	A1	K01.001A

Underkrav				
Krav ID	Beskrivelse	Test ID	Klasse	Status
K01.001A.01A	Skal ha maksimal forsinkelse på 200 ms fra handling oppstår til den vises i klientens videospiller.	T01.001.01	A1	
K01.001A.02A	Skal kunne sende video til 3 klienter.	T01.001.02	A1	
K01.001A.03A	Skal sende komprimert H.264 strøm til lagring, se krav K01.002A.	T01.001.03	A3	
K01.001A.04A	Skal kunne gjøre krav 02A og 03A samtidig.	T01.001.04	A1	
K01.001A.05C	Skal kunne stå på i 24 timer i døgnet i 7 dager i strekk.	T01.001.05	C1	Ikke testet

Status				
Krav	Godkjent	Testet av	Ansvarlig	Dato
A		Carl Peder Jansen	Carl Peder Jansen	29.04.2016
C			Carl Peder Jansen	

Dato	Kommentar
22.04	Utført test 01 og 02. Test 01 er ikke tilfredsstilt, men kravet er diskutert med oppdragsgiver. På grunn av begrensninger i hardware vil ikke dette kravet kunne bli oppfylt med de midler vi har blitt tildelt. Test utført av Alexander Schinnes.
29.04	Utført test 02A, 03C og 04A, alt fungerte som ønsket. Test 01A ble ikke utført pga samme årsak som forrige test. Test 05C er ikke testet på grunn av tidsmangel. Test utført Carl Peder Jansen

DELTEST

Dato: 06.05.2016

Hvor: Høgskolen i Sørøst-Norge

T01 – System og server			
Test ID	Beskrivelse	Klasse	Krav ID
T01.002	Skal kunne lagre/hente opp alle videoene, tags (ref. K03.006C), og skjermdumper (ref. K03.007C)	A1	K01.002A

Underkrav				
Krav ID	Beskrivelse	Test ID	Klasse	Status
K01.002A.01A	Skal lagre all video som blir tatt opp.	T01.002.01	A1	
K01.002A.02C	Skal lagre alle tags som blir opprettet.	T01.002.02	C3	Ikke testet
K01.002A.03A	Skal kunne hente opp video.	T01.002.03	A1	
K01.002A.04C	Skal kunne hente opp tags.	T01.002.04	C3	Ikke testet
K01.002A.05C	Skal kunne hente opp skjermdumper.	T01.002.05	C3	Ikke testet

Status				
Krav	Godkjent	Testet av	Ansvarlig	Dato
A		Alexander Schinnes	Carl Peder Jansen	29.04.2016
C			Carl Peder Jansen	06.05.2016

Dato	Kommentar
29.04	Test 01 og 02 testet og godkjent. Alexander
06.05	Alle C kravene blir her nedprioritert på grunn av tidsmangel, disse er derfor ikke testet. Carl Peder Jansen

DELTEST

Dato: 22.04.2016

Hvor: Høgskolen i Sørøst-Norge

T01 – System og server			
Test ID	Beskrivelse	Klasse	Krav ID
T01.003	Skal kunne motta video fra videokamera på server.	A1	K01.003A

Status				
Krav	Godkjent	Testet av	Ansvarlig	Dato
A		Alexander Schinnes	Carl Peder Jansen	22.04.2016

Dato	Kommentar
22.04	Utført test T01.003 (har ingen undertester). Alexander

DELTEST

Dato: 29.04.2016

Hvor: Høgskolen i Sørøst-Norge

T01 – System og server			
Test ID	Beskrivelse	Klasse	Krav ID
T01.005	Det skal utføres analyse av hvordan forandring av FPS og oppløsning påvirker videostrømmens forsinkelse.	A2	K01.005A

Status				
Krav	Godkjent	Testet av	Ansvarlig	Dato
A		Alexander Schinnes	Carl Peder Jansen	29.04.2016

Dato	Kommentar
22.04	Test og krav bør skrives om. Alexander
29.04	Denne testen blir vanskelig å gjennomføre før hele systemet er på plass. Carl Peder
13.05	Test utført. Resultatet på forsinkelse er som følger: 1080p – 25 FPS gir forsinkelse = 500ms 1080p – 50 FPS gir forsinkelse = 550ms 720p – 25 FPS gir forsinkelse = 450ms 720p – 50 FPS gir forsinkelse = 500ms 480x270 – 25 FPS gir forsinkelse = 500ms 480x270 – 50 FPS gir forsinkelse = 500ms Alexander

DELTEST

Dato: 22.04.2016

Hvor: Høgskolen i Sørøst-Norge

T02 - Klient			
Test ID	Beskrivelse	Klasse	Krav ID
T02.001	Test av kommunikasjon mellom klient og server	A1	K02.001A

Underkrav				
Krav ID	Beskrivelse	Test ID	Klasse	Status
K02.001A.01A	Skal kunne koble seg opp mot server.	K02.001A.01A	A1	
K02.001A.02A	Skal kunne motta video fra server.	K02.001A.02A	A1	
K02.001A.03A	Skal kunne videresende video til videoavspiller.	K02.001A.03A	A1	
K02.001A.04A	Skal kunne styre kameraposisjon.	K02.001A.04A	A1	
K02.001A.05B	Skal kunne velge videostrøm fra server	K02.001A.05B	B1	Ikke Testet
K02.001A.06A	Skal kunne velge å ta opptak av videostrøm på server.	K02.001A.06A	A1	

Status				
Krav	Godkjent	Testet av	Ansvarlig	Dato
A		Alexander Schinnes	Carl Peder Jansen	22.04.2016
B	Ikke Testet		Carl Peder Jansen	06.05.2016

Dato	Kommentar
22.04	Det gjenstår 3 tester som ikke har vært mulig å utføre fordi vi ikke har utviklet en funksjon for lagring av video. Dette blir utført ved et senere tidspunkt. Kommunikasjonen fungerer ellers og vi har kommunikasjonen på plass. -- Svein
29.04	Testet 03A og 06A. Alle A krav er godkjent. -- Alexander
06.05	B kravet 05B blir her nedprioritert på grunn av tidsmangel, dette er derfor ikke testet. Carl Peder Jansen

DELTEST

Dato: 06.05.2016

Hvor: Høgskolen i Sørøst-Norge

T03 – GUI			
Test ID	Beskrivelse	Klasse	Krav ID
T03.001	Skal kunne gi brukeren valgmuligheter som oppgitt i underkrav. Qt Quick skal brukes til all GUI for applikasjonen.	A1	K03.001A

Underkrav				
Krav ID	Beskrivelse	Test ID	Klasse	Status
K03.001A.01A	Skal kunne velge å se forsinkelse.	T03.001.01	A1	
K03.001A.02A	Skal kunne velge å se bilder i sekundet (FPS).	T03.001.02	A1	
K03.001A.03C	Skal kunne velge å vise «packet loss».	T03.001.03	C2	Ikke testet
K03.001A.04C	Skal kunne velge å se datatrafikk.	T03.001.04	C2	Ikke testet
K03.001A.05B	Skal kunne velge å se video i H.264 format.	T03.001.05	B1	Ikke testet
K03.001A.07C	Skal kunne velge å se CPU forbruk.	T03.001.07	C2	Ikke testet
K03.001A.08C	Skal kunne velge å se RAM forbruk.	T03.001.08	C2	Ikke testet
K03.001A.09A	Skal kunne velge å vise videoavspilleren uten effekter.	T03.001.09	A1	
K03.001A.10C	Skal kunne velge å vise videoavspilleren med IR.	T03.001.10	C1	Ikke testet
K03.001A.11A	Skal kunne velge å vise videoavspilleren med nattmodus.	T03.001.11	B1	
K03.001A.12A	Skal kunne velge å vise et «overlay» med all tilgjengelig informasjon.	T03.001.12	A2	
K03.001A.13B	Skal kunne velge FPS på videostrømmen.	T03.001.13	B2	Ikke testet

Status				
Krav	Godkjent	Testet av	Ansvarlig	Dato
A		Carl Peder Jansen	Carl Peder Jansen	06.05.2016
B		Carl Peder Jansen	Carl Peder Jansen	06.05.2016
C		Carl Peder Jansen	Carl Peder Jansen	06.05.2016

Dato	Kommentar
29.04	We have the technology to make the A great again. - Vegard Smedsvik
06.05	Alle A krav er testet og godkjent, B og C kravene som ikke er godkjent er ikke testet da disse ikke kommer til å bli prioritert i denne prosjekt perioden. Carl Peder Jansen

DELTEST

Dato: 29.04.2016

Hvor: Høgskolen i Sørøst-Norge

T03 – GUI			
Test ID	Beskrivelse	Klasse	Krav ID
T03.002	Skal kunne pause og spille av video.	B1	K03.002B

Underkrav				
Krav ID	Beskrivelse	Test ID	Klasse	Status
K03.002A.01A	Skal kunne spille av video fra gitt punkt.	T03.002.01	A1	
K03.002A.02A	Skal kunne pause video.	T03.002.02	A1	

Status				
Krav	Godkjent	Testet av	Ansvarlig	Dato
A		Vegard Smedsvik	Carl Peder Jansen	29.04.2016

Dato	Kommentar
29.04	Fungerer som det skal. Vegard Smedsvik

DELTEST

Dato: 29.04.2016

Hvor: Høgskolen i Sørøst-Norge

T03 – GUI			
Test ID	Beskrivelse	Klasse	Krav ID
T03.003	Skal kunne vise hvor lenge du har sett på en video på en tidslinje, hvor 0 er starten på videoen og slutten viser lengden på videoen. Qt Quick skal brukes til all GUI for applikasjonen.	B1	K03.003B

Underkrav				
Krav ID	Beskrivelse	Test ID	Klasse	Status
K03.003B.01B	Skal kunne trykke på tidslinjen og hoppe til gitt tidspunkt.	T03.003.01	B3	
K03.003B.02C	Skal kunne vise «Tags» som er blitt opprettet for gitt video på gitt tidspunkt.	T03.003.02	C3	Ikke Testet

Status				
Krav	Godkjent	Testet av	Ansvarlig	Dato
B		Vegard Smedsvik	Carl Peder Jansen	29.04.2016
C			Carl Peder Jansen	

Dato	Kommentar
29.04	Tidslinjen beveger seg riktig i forhold til videoens lengde, det kan hoppes til ønsket tidspunkt ved å trykke på tidslinjen. Tiden som blir vist til venstre for tidslinjen er ikke alltid riktig men testen e likevel ansett som godkjent . Vegard Smedsvik
06.05	C kravet 02C blir ikke testet da dette kravet er nedprioritert på grunn av tidsmangel. Carl Peder Jansen

DELTEST

Dato: 18.05.2016

Hvor: Høgskolen i Sørøst-Norge

T03 – GUI			
Test ID	Beskrivelse	Klasse	Krav ID
T03.004	Skal kunne spole video framover og bakover. Qt Quick skal brukes til all GUI for applikasjonen.	C1	K03.004C

Status				
Krav	Godkjent	Testet av	Ansvarlig	Dato
C	Ikke Testet		Carl Peder Jansen	

Dato	Kommentar
18.05	Denne testen er ikke blitt gjennomført da dette kravet er nedprioritert på grunn av tidsmangel. Carl Peder Jansen

DELTEST

Dato: 29.04.2016

Hvor: Høgskolen i Sørøst-Norge

T03 – GUI			
Test ID	Beskrivelse	Klasse	Krav ID
T03.005	Skal kunne hoppe direkte til «live stream» ved ett klikk. Qt Quick skal brukes til all GUI for applikasjonen.	A1	K03.005A

Status				
Krav	Godkjent	Testet av	Ansvarlig	Dato
A		Vegard Smedsvik	Carl Peder Jansen	29.04.16

Dato	Kommentar
29.04	Fungerer som det skal. Vegard Smedsvik

DELTEST

Dato: 06.05.2016

Hvor: Høgskolen i Sørøst-Norge

T03 – GUI			
Test ID	Beskrivelse	Klasse	Krav ID
T03.006	Skal opprette en indikasjon på tidslinjen, som kalles tags. Tags er et lite bokmerke som plasseres på valgte tidspunkter i en video. Her skal det være mulig å legge til kommentarer slik at man raskt finner fram til spesielle hendelser på en video. Qt Quick skal brukes til all GUI for applikasjonen.	C2	K03.006C

Underkrav				
Krav ID	Beskrivelse	Test ID	Klasse	Status
K03.006C.01A	Indikasjonen skal være plassert på nåværende tidspunkt i videoen.	T03.006.01	A3	Ikke Testet
K03.006C.02B	Skal kunne «hovre» over indikasjonen for å få mer informasjon.	T03.006.02	A3	Ikke Testet
K03.006C.03A	Skal kunne ha mulighet til å skrive inn en kommentar.	T03.006.03	B3	Ikke Testet

Status				
Krav	Godkjent	Testet av	Ansvarlig	Dato
A			Carl Peder Jansen	
B			Carl Peder Jansen	

Dato	Kommentar
06.05	Denne testen er ikke blitt gjennomført da dette kravet er nedprioritert på grunn av tidsmangel. Carl Peder Jansen

DELTEST

Dato: 06.05.2016

Hvor: Høgskolen i Sørøst-Norge

T03 – GUI			
Test ID	Beskrivelse	Klasse	Krav ID
T03.007	Skal kunne ta et bilde av videoavspilleren. Qt Quick skal brukes til all GUI for applikasjonen.	C2	K03.007C

Underkrav				
Krav ID	Beskrivelse	Test ID	Klasse	Status
K03.007C.01A	Skal lagres med informasjon om tidspunkt.	T03.007.01	A3	Ikke Testet
K03.007C.02A	Skal lagres med informasjon om hvilken video.	T03.007.02	A3	Ikke Testet
K03.007C.03B	Skal lagres med en kommentar gitt av brukeren.	T03.007.03	B3	Ikke Testet

Status				
Krav	Godkjent	Testet av	Ansvarlig	Dato
A			Carl Peder Jansen	
B			Carl Peder Jansen	

Dato	Kommentar
06.05	Denne testen er ikke blitt gjennomført da dette kravet er nedprioritert på grunn av tidsmangel. Carl Peder Jansen

DELTEST

Dato: 06.05.2016

Hvor: Høgskolen i Sørøst-Norge

T03 – GUI			
Test ID	Beskrivelse	Klasse	Krav ID
T03.010	3D mus Space Navigator, Tobii EyeX Controller og Multi Touch Control Wheel skal Integreres i applikasjonen og kontrollere kameraet.	A1	K03.010A

Underkrav				
Krav ID	Beskrivelse	Test ID	Klasse	Status
K03.010A.01A	3D Space Navigator skal integreres og styre kamera.	T03.010.01	A1	
K03.010A.02A	Tobii EyeX skal integreres og styre kamera.	T03.010.02	A1	
K03.010A.03A	MTCW skal integreres og styre kamera.	T03.010.03	A1	

Status				
Krav	Godkjent	Testet av	Ansvarlig	Dato
A		Carl Peder Jansen	Carl Peder Jansen	06.05

Dato	Kommentar
06.05	3D Space navigator fungerer meget godt, både styring av posisjon og zoom fungerer bra. EyeTracker fungerer også veldig fint. MTCW kan kontrollere kamera, men det er en noe dårlig respons til tider, zoom funksjonen er heller ikke så lett å styre. Testen er likevel godkjent da vi har kontroll på alle parametere.

DELTEST

Dato: 29.04.2016

Hvor: Høgskolen i Sørøst-Norge

T04 – Tobii EyeX Controller			
Test ID	Beskrivelse	Klasse	Krav ID
T04.001	Eyetrackeren skal vite hvor på skjermen brukeren ser etter kalibrering.	A1	K04.001A

Status				
Krav	Godkjent	Testet av	Ansvarlig	Dato
A		Carl Peder Jansen	Carl Peder Jansen	29.04.2016

Dato	Kommentar
29.04	Fungerte utmerket. Carl Peder Jansen

DELTEST

Dato: 22.04.2016

Hvor: Høgskolen i Sørøst-Norge

T05 – 3D mus Space Navigator			
Test ID	Beskrivelse	Klasse	Krav ID
T05.002	Skal kunne kontrolleres og registrere posisjon/endring ved bruk.	A2	K05.002A

Status				
Krav	Godkjent	Testet av	Ansvarlig	Dato
A		Alexander Schinnes	Carl Peder Jansen	29.04.2016

Dato	Kommentar
22.04	Testen er ikke godkjent da det fortsatt mangler funksjonalitet for sideknappene. Dette er da det eneste som avviker fra kravet. Carl Peder
29.04	Testet. Alle funksjoner fungerer og returnerer korrekt verdi. Alexander

DELTEST

Dato: 22.04.2016

Hvor: Høgskolen i Sørøst-Norge

T06 – Multi Touch Control Wheel			
Test ID	Beskrivelse	Klasse	Krav ID
T06.001	For at man skal kunne benytte seg av MTCW skal det lages en applikasjonsdriver som kan benyttes i Qt på Windows 7, 64-bits operativsystem.	A1	K06.001A

Status				
Krav	Godkjent	Testet av	Ansvarlig	Dato
A		Vegard Smedsvik	Carl Peder Jansen	22.04.2016

Dato	Kommentar
22.04	Det er laget en modul driver i Qt Quick. SDKen som er gitt av Grayhill er også implementert i driveren. Vegard Smedsvik

DELTEST

Dato: 09.04.2016

Hvor: Høgskolen i Sørøst-Norge

T06 – Multi Touch Control Wheel			
Test ID	Beskrivelse	Klasse	Krav ID
T06.007	Kameraet skal være motorisert og kunne styres med API fra krav K07.006A.	A1	K06.007A

Underkrav				
Krav ID	Beskrivelse	Test ID	Klasse	Status
K06.007A.01A	Skal kunne detektere fingre	K06.007.01	A1	
K06.007A.02A	Skal kunne bruke å kontrollere rotasjonskoderen	K06.007.02	A3	
K06.007A.03A	Skal kunne brukes til å zoome inn og ut på kamera	K06.007.03	A3	
K06.007A.04A	Skal kunne styre kameraposisjon i x, y	K06.007.04	A1	

Status				
Krav	Godkjent	Testet av	Ansvarlig	Dato
A		Carl Peder Jansen	Carl Peder Jansen	09.04.2016

Dato	Kommentar
22.04	MTCW kan detektere opp til 5 fingre og styre kameraposisjon i x og y-aksen. Grunnen til at status på 02A er rød, er fordi vi ikke har fått rotasjonskoderen til å fungere. Grunnen til at status på 03A er rød er fordi vi ikke har laget en funksjon som trigger Zoom funksjonen på kamera APIen. Det samme gjelder for styring av z-aksen. Testet av Vegard Smedsvik
29.04	MTCW kan detektere opp til 5 fingre og styre kameraposisjon i x og y-aksen. Kameraet kan styres i x,y retning. Det ligger funksjonalitet for 02A og 03A inne men denne er ikke god nok til å godkjennes. Testet av Carl Peder
09.05	MTCW fungerer nå brukbart, det er lagt inn en ny funksjonalitet for zoom, dette blir nå gjort på «mobil metoden» ved å dra sammen to fingere for å zoome ut og dra de fra hverandre for å zoome inn. Dette fungerer greit og testen ansett som godkjent. Testet av Carl Peder

DELTEST

Dato: 22.04.2016

Hvor: Høgskolen i Sørøst-Norge

T07 – Videokamera			
Test ID	Beskrivelse	Klasse	Krav ID
T07.001	Krav om støtte for H.264 format	A3	K07.001A

Status				
Krav	Godkjent	Testet av	Ansvarlig	Dato
A		Svein Bekkevold	Carl Peder Jansen	22.04.16

Dato	Kommentar
22.04	Ifølge databladet til kameraet er det støtte for H.264 format. Testen er derfor godkjent. Svein Bekkevold

DELTEST

Dato: 22.04.2016

Hvor: Høgskolen i Sørøst-Norge

T07 – Videokamera			
Test ID	Beskrivelse	Klasse	Krav ID
T07.002	Krav om støtte for bildefrekvens	A3	K07.002A

Status				
Krav	Godkjent	Testet av	Ansvarlig	Dato
A		Svein Bekkevold	Carl Peder Jansen	22.04.16

Dato	Kommentar
22.04	Ifølge databladet til kameraet er det mulig å gjøre opptak med bildefrekvens på minimum 30 fps. Svein Bekkevold

DELTEST

Dato: 22.04.2016

Hvor: Høgskolen i Sørøst-Norge

T07 – Videokamera			
Test ID	Beskrivelse	Klasse	Krav ID
T07.005	Kravet tilsier at vi vil ha et kamera som har en motor som gjør det mulig for oss å styre det fra en GUI.	A1	K07.005A

Underkrav				
Krav ID	Beskrivelse	Test ID	Klasse	Status
K07.005A.01A	Skal kunne styres fra -170 til +170 grader horisontalt.	T07.005.01	A1	
K07.005A.02A	Skal kunne styres fra -20 til +90 grader vertikalt.	T07.005.02	A1	

Status				
Krav	Godkjent	Testet av	Ansvarlig	Dato
A		Svein Bekkevold	Carl Peder Jansen	22.04.2016
A		Svein Bekkevold	Carl Peder Jansen	22.04.2016

Dato	Kommentar
22.04	Kameraet har mulighet til å styres i forhold til kravene. Svein Bekkevold

DELTEST

Dato: 22.04.2016

Hvor: Høgskolen i Sørøst-Norge

T07 – Videokamera			
Test ID	Beskrivelse	Klasse	Krav ID
T07.006	Kamera skal ha en API som skal kunne benyttes i Qt Quick applikasjoner. Det skal utvikles en generell API slik at det kan benyttes flere forskjellige kameraer, og at dette kan byttes ut hvis ønskelig.	A1	K07.006A

Status				
Krav	Godkjent	Testet av	Ansvarlig	Dato
A		Svein Bekkevold	Carl Peder Jansen	22.04.16

Dato	Kommentar
22.04	Vi har utviklet en API som gir mulighet for at forskjellige kameraer kan koble seg på og benytte seg av et generalisert styringssystem. Svein Bekkevold

DELTEST

Dato: 22.04.2016

Hvor: Høgskolen i Sørøst-Norge

T07 – Videokamera			
Test ID	Beskrivelse	Klasse	Krav ID
T07.007	Man skal kunne koble seg til kameraets real-time videobilde.	A1	K07.007A

Status				
Krav	Godkjent	Testet av	Ansvarlig	Dato
A		Svein Bekkevold	Carl Peder Jansen	22.04.16

Dato	Kommentar
22.04	Kameraet besitter flere typer grensesnitt som gjør det mulig for oss å koble kameraet opp mot server.

SYSTEM TEST

Dato: 13.05.2016

Hvor: Høgskolen i Sørøst-Norge

T10 - SYSTEM TEST	
Test ID	Beskrivelse
T10.000	Fullskala test når hele systemet er koblet sammen.

Status			
Godkjent	Testet av	Ansvarlig	Dato
	Carl Peder Jansen, Alexander Schinnes, Vegard Smedsvik og Svein Bekkevold	Carl Peder Jansen	13.05.2016

Dato	Kommentar
11.05	Det ble funnet flere store bugs i programmet og testen er defor ikke godkjent. Bugs: 1 Kamera stream må velges før Device, server krasjer hvis device blir valgt først 2 MTCW må bli bedre 3 EyeTracker Bug 4 Logitech stream 5 Legge til default camera (background pic) og device (arrow keys) 6 Camera connect bug, fungerer fra pc'en som hoster, men ikke de andre klientene 7 NV status blir ikke oppdatert for alle klienter 8 Swipe, NV og Connect to server knapper blir disablet og skal ikke kunne hovres over hvis kameraet er Busy 9 Det er kun en klient som kan gjøre record om gangen, record alert skal vises for alle klienter 10 Swipe har flere bugs, bla inactive status blir ikke oppdatert når swipet er ferdig. Det funker heller ikke å kjøre swipe samtidig som devicer er valgt.
12.05	Systemet er mye bedre men det finnes fortsatt flere bugs som må fikses. Det meste omhandler kommunikasjon mellom server og flere klienter. Bugs: 1 EyeTracker fungerer dårlig når den ikke hoster server selv 2 MTCW er bedre men bør fortsatt forbedres 3 Swipe funksjonen bør kunne stoppes underveis i swipet 4 Swipe setter ikke zoom (z akse til 0 hvis den er zoomet inn) 5 Record crash med axis og logitech samtidig 6 Record prioritet, skal klienter ha forskjellig prioritet på kameraet? 7 Klient ID fungerer noe spesielt… 8 Når logitech kamera er valgt bør NV og Swipe funksjonene disables

13.05	Det ble funnet noen små bugs. Bugs: 1. Live button 2. Logitech webcam 3. Latency webcam 4. Nightvision status Disse ble raskt fikset og systemet kjørte deretter feilfritt, testen er derfor ansett som godkjent.

REFERANSER

[1] – Testrapport: Bacheloroppgave. Consulant Calendar Overview (04.03.2016)
[2] - BachelorTangstad2015, tidligere bacheloroppgave, 05.02.16
[3] – UML IN A NUTSHELL: Sinan Si Albir (03.03.2016)
[4] – UML Distilled Third Edition: Martin Fowler (03.03.2016)
[5] – Schaum's Outline's UML: Simon Bennett, John Skelton (17.02.2016)
[6] - Olaf Hallan Graven, Veiledning for prosjekt 2016, 05.02.16

Høgskolen i Sørøst-Norge

KONGSBERG

GRUPPE 2

ITERASJONSPLAN – I1

13.01.2016 – 31.01.2016

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENT 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIKK 3
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 INNLEDNING 5
4 ITERASJON 5
4.1 HENSIKTEN 5
4.2 MÅ-MÅL 5
4.3 BØR-MÅL 6
5 AKTIVITETER 7
6 RESSURSER 8
6.1 PERSONER 8
6.2 UTSTYR FRA KDA 8
6.3 ANNET UTSTYR 8
6.4 LOKALER 8
REFERANSER 8

1 DOKUMENT

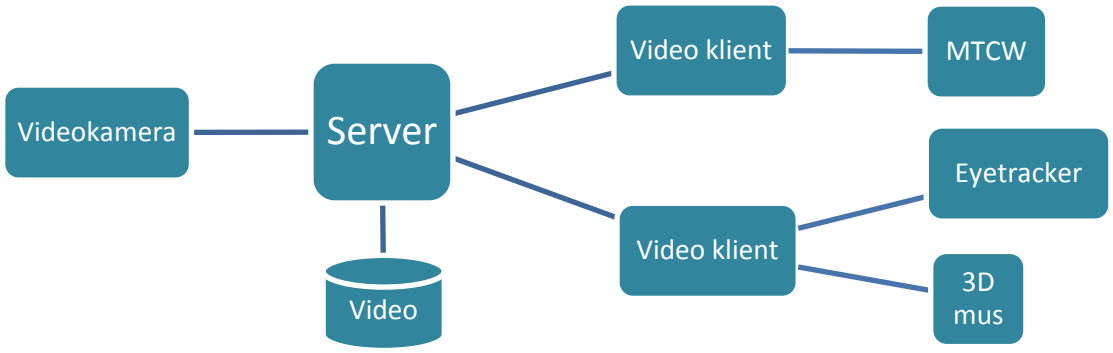
1.1 OVERSIKT

Dette dokumentet inneholder samtlige «må» og «bør» mål. Hvor alle «må» målene er førsteprioritet og «bør» målene vil bli påbegynt etter hvert som «må» målene er oppnådd og ferdigstilt. Dokumentet inneholder også hvilke ressurser, aktiviteter som er relevant for å oppnå målene og hvem som er ansvarlig. Samt en tidsfrist.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 – Systemoversikt

1.3 HISTORIKK

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
1.0	27.01.2016	Vegard Smedsvik	Første versjon

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Mmachine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 3
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4
Tabell 3 - Administrative må-mål 5
Tabell 4 - Tekniske må-mål 6
Tabell 5 - Administrative bør-mål 6
Tabell 6 - Tekniske bør-mål 6
Tabell 7 - Aktiviteter 7

2.4 FIGURER OG BILDER

Figur 1 - Systemoversikt 3

3 INNLEDNING

Hensikten med å ha en iterasjonsplan er å sette mål for neste iterasjon. Denne planen inneholder, hvilke ressurser og aktiviteter man trenger for å oppnå målene. Det er også en oversikt over hvilke mål man «må» og «bør» gjennomføre, og hvilke aktiviteter som er relevant for å nå disse. Hvem som er ansvarlig og ressurser som blir tatt i bruk står i aktivitetsoversikten.

4 ITERASJON

4.1 HENSIKTEN

Hensikten med denne iterasjonen er å samle inn informasjon. Hovedfokuset i starten av iterasjonen er å samle inn så mye informasjon så mulig, for så å produsere en grov versjon av diverse dokumenter. Under utviklingen av dokumentene vil vi ha jevnlige møter med intern og ekstern veileder. Dette er for å sikre at vi er enig om hvordan systemet skal fungere, samt at dokumentene holder en god standard.

4.2 MÅ-MÅL

Tabell 4 - Administrative må-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet
Lage prosjektplan	Alexsander	20	A1-001
Lage kravspesifikasjon	Alexsander, Vegard	20	A1-005
Lage testspesifikasjon	Svein, Carl Peder	30	A1-006
Første utkast av tidsplan	Vegard	20	A1-002
Lag generelle Use-Caser til systemet	Carl Peder Jansen	30	A1-004
Møter med KDA	Alle	20	A2-001
Møter med intern veileder	Alle	8	A2-002
Valg av prosjektmodell	Gruppen	10	A1-001
Sette av dato for 1 presentasjon	Vegard	0.5	A2-005
Ferdigstille kontrakt	Svein	4	A1-007
Lage risikovurdering	Vegard, Alexander	10	A1-008
Lage iterasjonsplan I1	Vegard	6	A1-010
Lage iterasjonsplan I2	Vegard	4	A1-011
Lage oppfølgingsdokument	Svein	3	A5-002
Lage aktivitetsplan	Svein	10	A1-003

Tabell 5 - Tekniske må-mål

Bekrivelse	Ansvar	Tidsestimat	Aktivitet
Kurs i QT	Vegard, Alexander	2	A4-001
Kurs i C++	Vegard, Alexander	4	A4-002
Få og hente utstyr fra KDA	Alle	8	A5-001

4.3 BØR-MÅL

Tabell 6 - Administrative bør-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet
Lage standard mal	Alexander	4	A1-009
Finne på navn	Gruppen	2	A5-005

Tabell 7 - Tekniske bør-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet
Prototype av systemet	Vegard	4	A6-009
Lage en logo til prosjektet	Vegard	4	A5-005

5 AKTIVITETER

Tabell 8 - Aktiviteter

#	Aktivitet	Ansvarlig	Kort beskrivelse	Tidsestimat
A1-001	Prosjektplan	Svein	Første utkast	20
A1-002	Tidsplan	Alexander	Første utkast	15
A1-003	Aktivitetsplan	Svein	Lage	15
A1-004	Use Case	Carl Peder	Lage overordnet system	20
A1-005	Kravspesifikasjon	Alexander, Vegard	Første utkast	30
A1-006	Testspesifikasjon	Carl Peder, Svein	Første utkast	25
A1-007	Kontrakter	Svein	Sørge for at de oppdatert	5
A1-008	Risikodokument	Alexander	Lage	8
A1-009	Lage dokumentmaler	Alexander	Lage en standard mal	10
A1-010	Iterasjons plan	Vegard	Lage mal og lage plan for første iterasjon	3
A1-011	Iterasjons plan	Vegard	Lage iterasjons plan for andre iterasjon	3
A2-001	Kongsberg Gruppen	Alle	Møte med KDA	10
A2-002	Veiledermøte internt	Alle	Møte med intern veileder	8
A4-001	Kurs i QT	Vegard, Alexander	Kurs for resten av gruppen	8
A4-002	Kurs i C++	Vegard, Alexander	Kurs for resten av gruppen	4
A5-001	Hente og installere utstyr	Alle	Hente utstyr hos KDA, skaffe rom samt installere oss der.	8
A5-002	Oppfølgnings-dokument	Svein	Lage mal og oppdatere oppfølgningsdokument	4
A5-003	Skrive møtereferat	Svein	Skrive møtereferat etter alle møter	2
A5-004	Forberedelser til møter	Vegard	Sende innkalling til alle møter	4
A5-005	Annet	Alle	Diskutere navn og logo	3
A6-009	Prototype	Vegard	Lage en enkel prototype	4

6 RESSURSER

6.1 PERSONER

- Alexander Schinnes
- Svein Bekkevold
- Carl Peder Jansen
- Vegard Smedsvik

6.2 UTSTYR FRA KDA

- Stasjonær pc med utstyr til hver av medlemmene
- 3D Mus
- Tobii EyeTracker
- Multi Touch Control Wheel

6.3 ANNET UTSTYR

- Private bærbare

6.4 LOKALER

Vi har fått tildelt Utviklings lab – 1204 av Høgskolen i Sørøst-Norge avd. Kongsberg

REFERANSER

Ingen

Høgskolen i Sørøst-Norge

KONGSBERG

GRUPPE 2

ITERASJONSPLAN – I2

01.02.2016 – 12.02.2016

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENTOMFANG 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIKK 3
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 INNLEDNING 5
4 ITERASJON 5
4.1 HENSIKTEN 5
4.2 MÅ-MÅL 5
4.3 BØR-MÅL 6
5 AKTIVITETER 7
6 RESSURSER 7
6.1 PERSONER 7
6.2 UTSTYR FRA KDA 7
6.3 ANNET UTSTYR 7
6.4 LOKALER 7
REFERANSER 8

1 DOKUMENTOMFANG

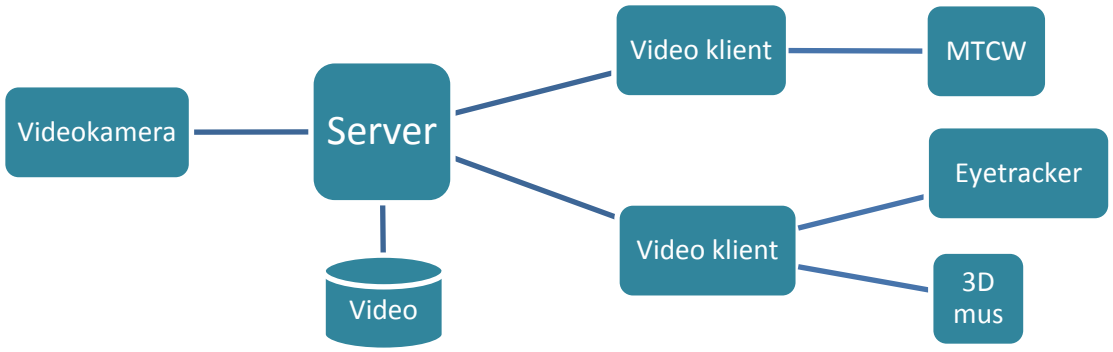
1.1 OVERSIKT

Dette dokumentet inneholder samtlige «må» og «bør» mål. Hvor alle «må» målene er førsteprioritet og «bør» målene vil bli påbegynt etter hvert som «må» målene er oppnådd og ferdigstilt. Dokumentet inneholder også hvilke ressurser, aktiviteter som er relevant for å oppnå målene og hvem som er ansvarlig. Samt en tidsfrist.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 - Systemoversikt

1.3 HISTORIKK

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
1.0	27.01.2016	Vegard Smedsvik	Første versjon

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Mmachine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 3
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4
Tabell 3 - Administrative må-mål 5
Tabell 4 - Tekniske må-mål 5
Tabell 5 - Administrative bør-mål 6
Tabell 6 - Tekniske bør-mål 6
Tabell 7 - Aktiviteter 7

2.4 FIGURER OG BILDER

Figur 1 - Systemoversikt 3

3 INNLEDNING

Hensikten med å ha en iterasjonsplan er å sette mål for neste iterasjon. Denne planen inneholder hvilke ressurser og aktiviteter man trenger for å oppnå målene. Det er også en oversikt over hvilke mål man «må» og «bør» gjennomføre, og hvilke aktiviteter som er relevant for å nå disse. Hvem som er ansvarlig og ressurser som blir tatt i bruk står i aktivitetsoversikten.

4 ITERASJON

4.1 HENSIKTEN

Hensikten med denne iterasjonen er å sette sammen og sørge for at hvert dokument inneholder riktig informasjon. Det vil også bli diskutert forbedringer til oppsett. I tillegg så kommer vi til å ha fokus på få laget en mal til alle dokumenter slik at de er klare for første innlevering. Slutten på denne iterasjonen vil vi bli markert ved å ha en presentasjon, hvor vi skal presentere oppgaven og hva vi har tenkt framover.

4.2 MÅ-MÅL

Tabell 4 - Administrative må-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet
Oppdatere prosjektplan	Alexander	20	A1-001
Oppdatere kravspesifikasjon	Alexander	20	A1-005
Oppdatere testspesifikasjon	Svein, Carl Peder	30	A1-006
Oppdatere tidsplan	Vegard	20	A1-002
Møter med KDA	Alle	20	A2-001
Møter med intern veileder	Alle	8	A2-002
Ferdigstille kontrakt	Svein	4	A1-007
Oppdatere risikovurdering	Alexander	10	A1-008
Lage iterasjonsplan I3	Vegard	3	A1-012
Lage oppfølgingsdokument	Svein	3	A5-002
Oppdatere aktivitetsliste	Alexander	10	A1-003
Lage presentasjon	Vegard	5	A2-006
Øve til presentasjon	Alle	40	A2-006
Lage visjonsdokument	Svein	8	A1-030

Tabell 5 - Tekniske må-mål

Bekrivelse	Ansvar	Tidsestimat	Aktivitet

4.3 BØR-MÅL

Tabell 6 - Administrative bør-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet
Finne på navn	Alle	2	A5-005

Tabell 7 - Tekniske bør-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet
Lage en logo	Vegard	2	A5-005

5 AKTIVITETER

Tabell 8 - Aktiviteter

#	Aktivitet	Ansvarlig	Kort beskrivelse	Tidsestimat
A1-001	Prosjektplan	Alexander	Oppdatere prosjektplan	20
A1-002	Tidsplan	Vegard	Oppdatere tidsplan	20
A1-003	Aktivitetsliste	Alexander	Oppdatere aktivitetsliste	10
A1-005	Kravspesifikasjon	Alexander	Oppdatere kravspesifikasjon	20
A1-006	Testspesifikasjon	Svein, Carl Peder	Oppdatere testspesifikasjon	30
A1-007	Kontrakt	Svein	Ferdigstille kontrakt	4
A1-008	Risikodokument	Alexander	Oppdatere risikodokument	10
A1-012	Lage iterasjonsplan 3	Vegard	Lage iterasjonsplan 3	3
A1-030	Visjonsdokument	Svein	Lage visjonsdokument	8
A2-001	Møte med KDA	Alle	Møte med KDA	8
A2-002	Møte intern	Alle	Møte med intern veileder	4
A2-006	Presentasjon	Vegard, Alle	Lage og øve til presentasjon	45
A5-002	Oppfølgnings-dokument	Svein	Lage oppfølgningsdokument for uke 7	3

6 RESSURSER

6.1 PERSONER

- Alexander Schinnes
- Svein Bekkevold
- Carl Peder Jansen
- Vegard Smedsvik

6.2 UTSTYR FRA KDA

- Stasjonær pc med utstyr til hver av medlemmene
- 3D Mus
- Tobii EyeTracker
- Multi Touch Control Wheel

6.3 ANNET UTSTYR

- Private bærbare

6.4 LOKALER

Vi har fått tildelt Utviklings lab – 1204 av Høgskolen i Sørøst-Norge avd. Kongsberg

REFERANSER

Ingen

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

ITERASJONSPLAN – E1

13.02.2016 – 21.02.2016

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENTOMFANG 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIE 3
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 INNLEDNING 5
4 ITERASJON 5
4.1 HENSIKTEN 5
4.2 MÅ-MÅL 5
4.3 BØR-MÅL 6
5 AKTIVITETER 6
6 RESSURSER 7
6.1 PERSONER 7
6.2 UTSTYR FRA KDA 7
6.3 ANNET UTSTYR 7
6.4 LOKALER 7
REFERANSER 7

1 DOKUMENTOMFANG

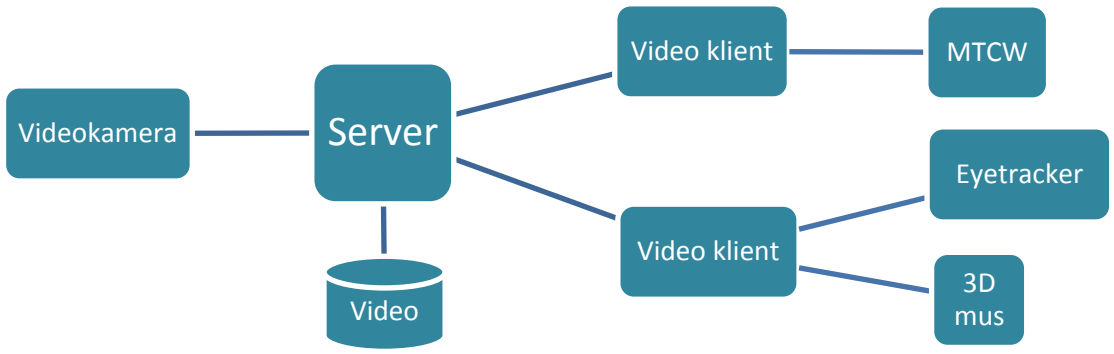
1.1 OVERSIKT

Dette dokumentet inneholder samtlige «må» og «bør» mål. Hvor alle «må» målene er førsteprioritet og «bør» målene vil bli påbegynt etter hvert som «må» målene er oppnådd og ferdigstilt. Dokumentet inneholder også hvilke ressurser, aktiviteter som er relevant for å oppnå målene og hvem som er ansvarlig. Samt en tidsfrist.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 - Systemoversikt

1.3 HISTORIE

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
1.0	17.02.2016	Vegard	Første utkast

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 3
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4
Tabell 3 - Administrative må-mål 5
Tabell 4 - Tekniske må-mål 5
Tabell 5 - Administrative bør-mål 6
Tabell 6 - Tekniske bør-mål 6
Tabell 7 - Aktiviteter 6

2.4 FIGURER OG BILDER

Figur 1 - Systemoversikt 3

3 INNLEDNING

Hensikten med å ha en iterasjonsplan er å sette mål for neste iterasjon. Denne planen inneholder hvilke ressurser og aktiviteter man trenger for å oppnå målene. Det er også en oversikt over hvilke mål man «må» og «bør» gjennomføre, og hvilke aktiviteter som er relevant for å nå disse. Hvem som er ansvarlig og ressurser som blir tatt i bruk står i aktivitetsoversikten.

4 ITERASJON

4.1 HENSIKTEN

Hensikten med denne iterasjonen er å samle inn informasjon om hvordan vi skal bygge opp arkitekturen. Hovedfokuset på denne iterasjonen vil være å skape kommunikasjon mellom to PCer ved hjelp av UDP og TCP. Designdokument og implementasjonsdokument vil også bli påbegynt, slik at vi kan fylle dem ut i neste iterasjon.

4.2 MÅ-MÅL

Tabell 4 - Administrative må-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet
Skrive iterasjonsplan 3	Vegard	2	A1-012
Skrive iterasjonsrapport 2	Vegard	4	A1-011
Skrive iterasjonsplan 4	Vegard	2	A1-013
Ferdigstille kravspesifikasjon	Alexander	4	A1-005
Arkitektur	Svein	8	A7-001
Endre systembeskrivelse	Alexander	4	A1-001
Designdokument	Carl Peder	8	A1-031
Implementasjonsdokument	Carl Peder	8	A1-032

Tabell 5 - Tekniske må-mål

Bekrivelse	Ansvar	Tidsestimat	Aktivitet
Kommunikasjon med TCP	Svein	16	A6-002
Kommunikasjon med UDP	Alexander	16	A6-003
Reseach innen det å lage driver	Vegard	16	A60-001

4.3 BØR-MÅL

Tabell 6 - Administrative bør-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet
Møte med intern veileder på Krona onsdag 17.02.16, Svein skriver møtereferat	Alle	4	A2-002
Møte med ekstern veileder på Krona torsdag 18.02.16, Svein skriver møtereferat	Alle	4	A2-003

Tabell 7 - Tekniske bør-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet

5 AKTIVITETER

Tabell 8 - Aktiviteter

#	Aktivitet	Ansvarlig	Kort beskrivelse	Tidsestimat
A1-001	Prosjektplan	Alexander	Endre systembeskrivelse	4
A1-005	Ferdigstille kravspesifikasjon	Alexander	Ferdigstille kravspesifikasjon	4
A1-011	iterasjonsrapport 2	Vegard	Skrive iterasjonsrapport 2	4
A1-012	iterasjonsplan 3	Vegard	Skrive iterasjonsplan 3	2
A1-013	Skrive iterasjonsplan 4	Vegard	Skrive iterasjonsplan 4	2
A1-031	Designdokument	Carl Peder	Designdokument	8
A1-032	Implementasjon dokument	Carl Peder	Implementasjonsdokument	8
A2-002	Møte intern	Alle	Møte med intern veileder på Krona onsdag 17.02.16, Svein skriver møtereferat	4
A2-003	Møte ekstern	Alle	Møte med ekstern veileder på Krona torsdag 18.02.16, Svein skriver møtereferat	4
A60-001	Driver	Vegard	Research innen det å lage driver	16
A6-002	TCP	Svein	Kommunikasjon med TCP	16
A6-003	UDP	Alexander	Kommunikasjon med UDP	16
A7-001	Arkitektur	Svein	Arkitektur	8

6 RESSURSER

6.1 PERSONER

- Alexander Schinnes
- Svein Bekkevold
- Carl Peder Jansen
- Vegard Smedsvik

6.2 UTSTYR FRA KDA

- Stasjonær pc med utstyr til hver av medlemmene
- 3D Mus
- Tobii EyeTracker
- Multi Touch Control Wheel

6.3 ANNET UTSTYR

- Private bærbare

6.4 LOKALER

Vi har fått tildelt Utviklings lab – 1204 av Høgskolen i Sørøst-Norge avd. Kongsberg

REFERANSER

Ingen

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

ITERASJONSPLAN – E2

22.02.2016 – 28.02.2016

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENTOMFANG 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIE 3
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 INNLEDNING 5
4 ITERASJON 5
4.1 HENSIKTEN 5
4.2 MÅ-MÅL 5
4.3 BØR-MÅL 5
5 AKTIVITETER 7
6 RESSURSER 7
6.1 PERSONER 7
6.2 UTSTYR FRA KDA 7
6.3 ANNET UTSTYR 7
6.4 LOKALER 7
REFERANSER 8

1 DOKUMENTOMFANG

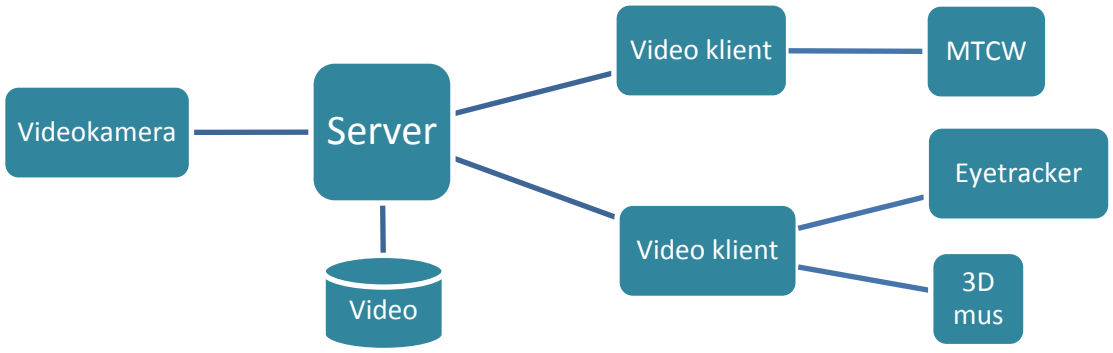
1.1 OVERSIKT

Dette dokumentet inneholder samtlige «må» og «bør» mål. Hvor alle «må» målene er førsteprioritet og «bør» målene vil bli påbegynt etter hvert som «må» målene er oppnådd og ferdigstilt. Dokumentet inneholder også hvilke ressurser, aktiviteter som er relevant for å oppnå målene og hvem som er ansvarlig. Samt en tidsfrist.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 - Systemoversikt

1.3 HISTORIE

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
1.0	17.02.2016	Vegard	Første utkast

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Mmachine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 3
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4
Tabell 3 - Administrative må-mål 5
Tabell 4 - Tekniske må-mål 5
Tabell 5 - Administrative bør-mål 5
Tabell 6 - Tekniske bør-mål 6
Tabell 7 - Aktiviteter 7

2.4 FIGURER OG BILDER

Figur 1 - Systemoversikt 3

3 INNLEDNING

Hensikten med å ha en iterasjonsplan er å sette mål for neste iterasjon. Denne planen inneholder hvilke ressurser og aktiviteter man trenger for å oppnå målene. Det er også en oversikt over hvilke mål man «må» og «bør» gjennomføre, og hvilke aktiviteter som er relevant for å nå disse. Hvem som er ansvarlig og ressurser som blir tatt i bruk står i aktivitetsoversikten.

4 ITERASJON

4.1 HENSIKTEN

Hensikten med denne iterasjonen er å begynne å designe og analysere systemer og delsystemer. Grunnarkitektur og rammeverk vil bli utviklet, slik at videre utvikling skal gå lettere. Hovedfokuset vil bli design og arkitektur, hvor vi skal lage et designdokument og diverse diagrammer for å beskrive systemets oppbygning.

4.2 MÅ-MÅL

Tabell 4 - Administrative må-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet
Design	Carl Peder	32	A7-003
Skrive iterasjonsplan 5	Vegard	2	A1-014
Skrive iterasjonsrapport 3	Vegard	3	A1-013
Arkitektur	Svein, Vegard	32	A7-001
Møte med intern veileder på Krona onsdag 24.02.16, Svein skriver møtereferat	Alle	4	A2-002
Møte med ekstern veileder og teknisk veileder på Krona torsdag 23.02.16, Svein skriver møtereferat	Alle	8	A2-003

Tabell 5 - Tekniske må-mål

Bekrivelse	Ansvar	Tidsestimat	Aktivitet
Sende videostrøm til klienter og motta videostrøm fra server	Alexander	32	A10-002. A20-02

4.3 BØR-MÅL

Tabell 6 - Administrative bør-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet

Tabell 7 - Tekniske bør-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet

5 AKTIVITETER

Tabell 8 - Aktiviteter

#	Aktivitet	Ansvarlig	Kort beskrivelse	Tidsestimat
A1-013	Skrive iterasjonsrapport 3	Vegard	Skrive iterasjonsrapport 3	3
A1-014	Skrive iterasjonsplan 5	Vegard	Skrive iterasjonsplan 5	2
A2-002	Veiledermøte internt	Alle	Møte med intern veileder på Krona onsdag 24.02.16, Svein skriver møtereferat	4
A2-003	Veiledermøte eksternt	Alle	Møte med ekstern veileder og teknisk veileder på Krona torsdag 23.02.16, Svein skriver møtereferat	8
A7-001	Arkitektur	Svein, Vegard	Arkitektur	32
A7-003	Design	Carl Peder	Design	32
A10-002. A20-02	Sende videostrøm til klienter, motta videostrøm fra server	Alexander	Sende videostrøm til klienter og motta videostrøm fra server	32

6 RESSURSER

6.1 PERSONER

- Alexander Schinnes
- Svein Bekkevold
- Carl Peder Jansen
- Vegard Smedsvik

6.2 UTSTYR FRA KDA

- Stasjonær pc med utstyr til hver av medlemmene
- 3D Mus
- Tobii EyeTracker
- Multi Touch Control Wheel

6.3 ANNET UTSTYR

- Private bærbare

6.4 LOKALER

Vi har fått tildelt Utviklings lab – 1204 av Høgskolen i Sørøst-Norge avd. Kongsberg

REFERANSER

Ingen

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

ITERASJONSPLAN – E3

29.02.2016 – 06.03.2016

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENTOMFANG 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIE 3
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 INNLEDNING 5
4 ITERASJON 5
4.1 HENSIKTEN 5
4.2 MÅ-MÅL 5
4.3 BØR-MÅL 6
5 AKTIVITETER 6
6 RESSURSER 7
6.1 PERSONER 7
6.2 UTSTYR FRA KDA 7
6.3 ANNET UTSTYR 7
6.4 LOKALER 7
REFERANSER 7

1 DOKUMENTOMFANG

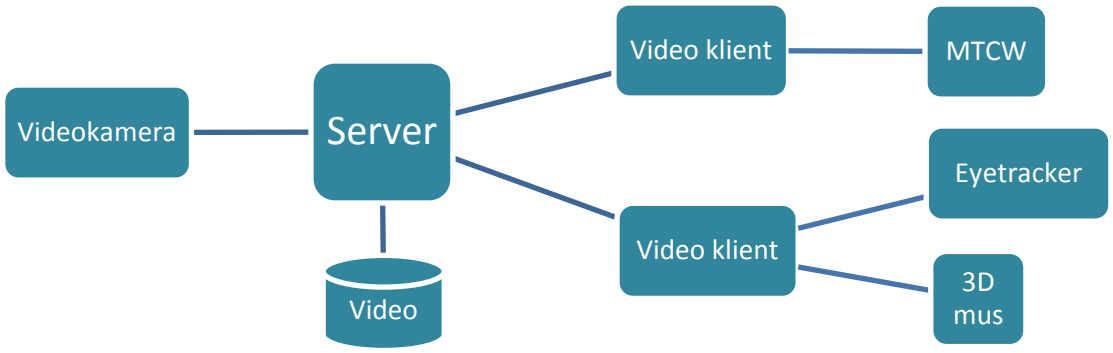
1.1 OVERSIKT

Dette dokumentet inneholder samtlige «må» og «bør» mål. Hvor alle «må» målene er førsteprioritet og «bør» målene vil bli påbegynt etter hvert som «må» målene er oppnådd og ferdigstilt. Dokumentet inneholder også hvilke ressurser, aktiviteter som er relevant for å oppnå målene og hvem som er ansvarlig. Samt en tidsfrist.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 - Systemoversikt

1.3 HISTORIE

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
1.0	26.02.2016	Svein	Første utkast

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 3
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4
Tabell 3 - Administrative må-mål 5
Tabell 4 - Tekniske må-mål 5
Tabell 5 - Administrative bør-mål 6
Tabell 6 - Tekniske bør-mål 6
Tabell 7 - Aktiviteter 6

2.4 FIGURER OG BILDER

Figur 1 - Systemoversikt 3

3 INNLEDNING

Hensikten med å ha en iterasjonsplan er å sette mål for neste iterasjon. Denne planen inneholder hvilke ressurser og aktiviteter man trenger for å oppnå målene. Det er også en oversikt over hvilke mål man «må» og «bør» gjennomføre, og hvilke aktiviteter som er relevant for å nå disse. Hvem som er ansvarlig og ressurser som blir tatt i bruk står i aktivitetsoversikten.

4 ITERASJON

4.1 HENSIKTEN

Hensikten med denne perioden er å avslutte elaboration fasen. Alle nødvendige dokumenter skal være produsert, slik at vi kan fortsette arbeidet. Tidsplan skal være oppdatert, samt sekvensdiagram, klasse diagram, prosjektplan osv. Samtidig skal et designdokument oppdrettes.
I tillegg så må vi sette en dato for presentasjon 2. Denne presentasjonen skal ha teknisk fokus.

4.2 MÅ-MÅL

Tabell 4 - Administrative må-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet
Skrive iterasjonsplan C1	Carl Peder	3	A1-015
Skrive iterasjonsrapport E3	Alexander	3	A1-014
Møte med intern veileder på Krona onsdag 03.03.16, Svein skriver møtereferat	Alle	4	A2-002
Møte med teknisk veileder på Krona torsdag 04.03.16, Svein skriver møtereferat	Alle	4	A2-003
Oppdatere Gantt og tidsplan	Vegard	4	A1-002
Kontakte Grayhill ang driver og kode	Vegard	2	A60-001
Skrive første utkast til designdokument som skal inneholde class, component, GUI og sekvens diagram. (Få med tekstbeskrivelse i sekvensdiagrammet)	Vegard	12	A1-031
Use case. Oppdatering av sekvensbeskrivelser	Carl Peder	4	A1-004
Oppdatere nettside	Svein	2	A7-005
Analysedokument mot videostrøm	Alexander	8	A1-034

Tabell 5 - Tekniske må-mål

Bekrivelse	Ansvar	Tidsestimat	Aktivitet
TCP. Forbedring av basic kommunikasjon. Research mot struct	Svein	16	A6-002
Videoserver feilsøking software, kontakte auresius og prøve å finne best løsning	Alexander	16	A10-005
Spille av video fra videoserver i Qt	Carl Peder	16	A10-003

4.3 BØR-MÅL

Tabell 6 - Administrative bør-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet

Tabell 7 - Tekniske bør-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet

5 AKTIVITETER

Tabell 8 - Aktiviteter

#	Aktivitet	Ansvarlig	Kort beskrivelse	Tidsestimat
A1-015	Skrive iterasjonsplan C1	Carl Peder	Skrive iterasjonsplan 6	3
A1-014	Skrive iterasjonsrapport E3	Alexander	Skrive iterasjonsrapport 5	3
A2-002	Veiledermøte intern	Alle	Møte med intern veileder på Krona onsdag 03.03.16, Svein skriver møtereferat	4
A2-003	Veiledermøte ekstern	Alle	Møte med teknisk veileder på Krona torsdag 04.03.16, Svein skriver møtereferat	8
A1-002	Gantt	Vegard	Oppdatere Gantt og tidsplan	4
A60-001	Grayhill	Vegard	Kontakte Grayhill ang driver og kode	2
A1-031	Designdokument	Vegard	Skrive første utkast til design dokument som skal inneholde class, component, GUI og sekvens diagram. (Få med tekstbeskrivelse i sekvensdiagrammet)	12
A1-004	Use Case	Carl Peder	Use case. Oppdatering av sekvensbeskrivelser	4
A7-005	Hjemmeside	Svein	Oppdatere hjemmeside	2
A6-002	TCP	Svein	TCP. Forbedring av basic kommunikasjon. Research mot struct	16
A1-034	Videostrøm	Alexander	Analysedokument mot videostrøm	8
A10-005	Videoserver	Alexander	Videoserver feilsøking software, kontakte auresius og prøve å finne best løsning	16
A10-003	Videoserver	Carl Peder	Spille av video fra videoserver i Qt	16

6 RESSURSER

6.1 PERSONER

- Alexander Schinnes
- Svein Bekkevold
- Carl Peder Jansen
- Vegard Smedsvik

6.2 UTSTYR FRA KDA

- Stasjonær pc med utstyr til hver av medlemmene
- 3D Mus
- Tobii EyeTracker
- Multi Touch Control Wheel

6.3 ANNET UTSTYR

- Private bærbare

6.4 LOKALER

Vi har fått tildelt Utviklings lab – 1204 av Høgskolen i Sørøst-Norge avd. Kongsberg

REFERANSER

Ingen

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

ITERASJONSPLAN – C1

07.03.2016 – 13.03.2016

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENTOMFANG 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIE 3
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 INNLEDNING 5
4 ITERASJON 5
4.1 HENSIKTEN 5
4.2 MÅ-MÅL 5
4.3 BØR-MÅL 6
5 AKTIVITETER 6
6 RESSURSER 7
6.1 PERSONER 7
6.2 UTSTYR FRA KDA 7
6.3 ANNET UTSTYR 7
6.4 LOKALER 7
REFERANSER 7

1 DOKUMENTOMFANG

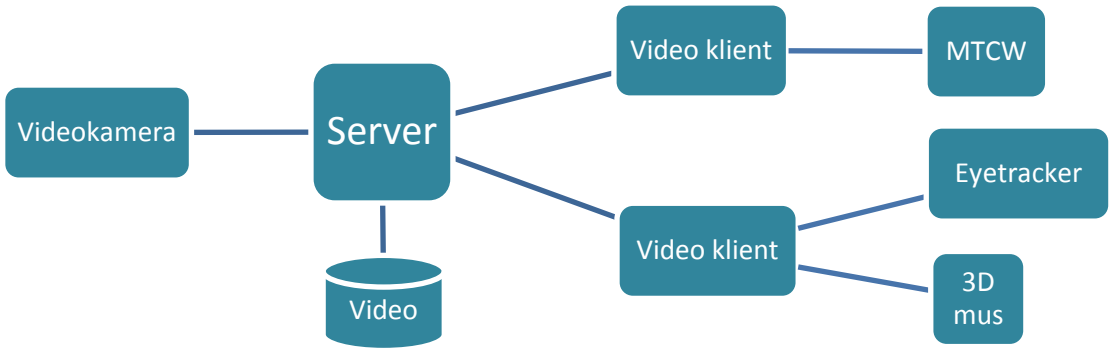
1.1 OVERSIKT

Dette dokumentet inneholder samtlige «må» og «bør» mål. Hvor alle «må» målene er førsteprioritet og «bør» målene vil bli påbegynt etter hvert som «må» målene er oppnådd og ferdigstilt. Dokumentet inneholder også hvilke ressurser, aktiviteter som er relevant for å oppnå målene og hvem som er ansvarlig. Samt en tidsfrist.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 - Systemoversikt

1.3 HISTORIE

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
1.0	04.03.2016	Carl Peder	Første utkast

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 3
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4
Tabell 3 - Administrative må-mål 5
Tabell 4 - Tekniske må-mål 5
Tabell 5 - Administrative bør-mål 6
Tabell 6 - Tekniske bør-mål 6
Tabell 7 - Aktiviteter 6

2.4 FIGURER OG BILDER

Figur 1 - Systemoversikt 3

3 INNLEDNING

Hensikten med å ha en iterasjonsplan er å sette mål for neste iterasjon. Denne planen inneholder hvilke ressurser og aktiviteter man trenger for å oppnå målene. Det er også en oversikt over hvilke mål man «må» og «bør» gjennomføre, og hvilke aktiviteter som er relevant for å nå disse. Hvem som er ansvarlig og ressurser som blir tatt i bruk står i aktivitetsoversikten.

4 ITERASJON

4.1 HENSIKTEN

Hovedfokuset i denne iterasjonen er å oppdatere og gjennomgå alt av dokumentasjon i prosjektet før den skal leveres inn til andre presentasjon. For å ha kontroll på revisjoner og versjoner av koden skal vi sette opp prosjektets struktur i revisjonskontroll programmet Tortoise Hg. Dette gir oss en ekstra sikkerhet man kan støtte seg på underveis i utviklingen.

4.2 MÅ-MÅL

Tabell 4 - Administrative må-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A1-016	Skrive iterasjonsplan C2	2	Vegard
A1-015	Skrive iterasjonsrapport C1	2	CP
A2-002	Møte med intern veileder på Krona onsdag 09.03.16, Svein skriver møtereferat	4	Alle
A2-003	Møte med teknisk veileder på Krona torsdag 10.03.16, Svein skriver møtereferat	8	Alle
A1-035	Gjennomgå all dokumentasjon i henhold til Karolines tilbakemeldinger -skrivefeil -referanser -gruppe 2 på alle forsider -riktig format -kjører alle dokumenter i minst versjon 1.0	20	Alle
A5-004	Sende innkallinger til presentasjon 2	0,5	Vegard
A1-035	Oppdatere systembeskrivelse i produserte dokumenter (husk prosjektplan)	2	Alexander
A2-006	Koordinere med Argos ang presentasjonen	0,5	Alexander

Tabell 5 - Tekniske må-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A7-004	Sette opp prosjektets struktur i Tortoise Hg	4	Vegard

4.3 BØR-MÅL

Tabell 6 - Administrative bør-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar

Tabell 7 - Tekniske bør-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A6-002	Kunne sende en struct/class fra server til klient ved hjelp av ha TCP.	26	Svein & Vegard
A30-001	Lage en qt applikasjon med enkel web browser uten GUI slik at vi kan styre kamera (visualisering av illustrasjon) + videoavspiller	30	Alexander og CP

5 AKTIVITETER

Tabell 8 - Aktiviteter

Aktivitet ID	Beskrivelse av aktivitet	Tidsestimat	Ansvar
A1-016	Skrive iterasjonsplan C2	2	Vegard
A1-015	Skrive iterasjonsrapport C1	2	CP
A2-002	Møte med intern veileder på Krona onsdag 09.03.16, Svein skriver møtereferat	4	Alle
A2-003	Møte med teknisk veileder på Krona torsdag 10.03.16, Svein skriver møtereferat	8	Alle
A1-035	Gjennomgå all dokumentasjon i henhold til Karolines tilbakemeldinger -skrivefeil -referanser -gruppe 2 på alle forsider -riktig format -kjører alle dokumenter i minst versjon 1.0	20	Alle
A5-004	Sende innkallinger	0,5	Vegard
A1-035	Oppdatere systembeskrivelse i produserte dokumenter (husk prosjektplan)	2	Alexander
A2-006	Koordinere med Argos ang presentasjonen	0,5	Alexander
A6-002	Det skal jobbes videre med TCP Vi skal kunne sende en struct/class fra server til klient	26	Svein & Vegard
A30-001	Lage en qt applikasjon med enkel web browser uten GUI slik at vi kan styre kamera (visualisering av illustrasjon) + videoavspiller	30	Alexander og CP
A7-004	Sette opp prosjektets struktur i Tortoise Hg	4	Vegard

6 RESSURSER

6.1 PERSONER

- Alexander Schinnes
- Svein Bekkevold
- Carl Peder Jansen
- Vegard Smedsvik

6.2 UTSTYR FRA KDA

- Stasjonær pc med utstyr til hver av medlemmene
- 3D Mus
- Tobii EyeTracker
- Multi Touch Control Wheel

6.3 ANNET UTSTYR

- Private bærbare

6.4 LOKALER

Vi har fått tildelt Utviklings lab – 1204 av Høgskolen i Sørøst-Norge avd. Kongsberg

REFERANSER

Ingen

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

ITERASJONPLAN – C2

14.03.2016 – 20.02.2016

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENTOMFANG 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIE 3
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 INNLEDNING 5
4 ITRASJON 5
4.1 HENSIKTEN 5
4.2 MÅ-MÅL 5
4.3 BØR-MÅL 5
5 AKTIVITETER 7
6 RESSURSER 7
6.1 PERSONER 7
6.2 UTSTYR FRA KDA 7
6.3 ANNET UTSTYR 7
6.4 LOKALER 7
REFERANSER 8

1 DOKUMENTOMFANG

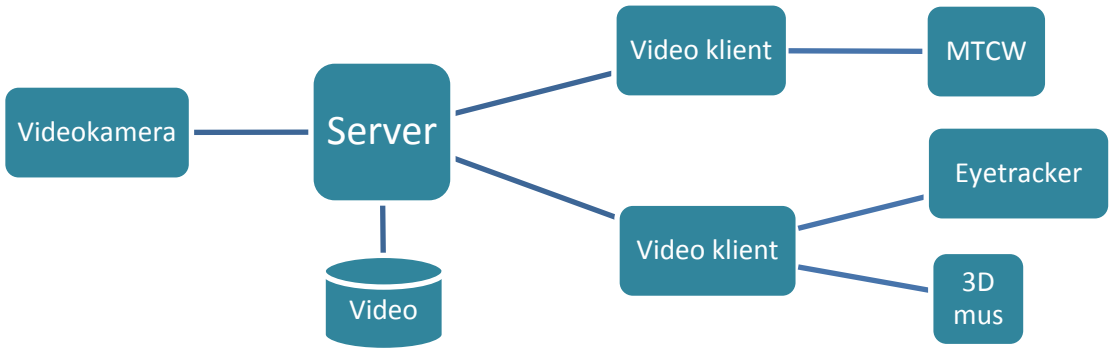
1.1 OVERSIKT

Dette dokumentet inneholder samtlige «må» og «bør» mål. Hvor alle «må» målene er førsteprioritet og «bør» målene vil bli påbegynt etter hvert som «må» målene er oppnådd og ferdigstilt. Dokumentet inneholder også hvilke ressurser, aktiviteter som er relevant for å oppnå målene og hvem som er ansvarlig. Samt en tidsfrist.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 - Systemoversikt

1.3 HISTORIE

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
1.0	11.03.2016	Vegard	Første utkast

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 3
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4

2.4 FIGURER OG BILDER

Figur 1 - Systemoversikt 3

3 INNLEDNING

Hensikten med å ha en iterasjonsplan er å sette mål for neste iterasjon. Denne planen inneholder hvilke ressurser og aktiviteter som trenger for å oppnå målene. Det er også en oversikt over hvilke mål man «må» og «bør» gjennomføre, og hvilke aktiviteter som er relevant for å nå disse. Hvem som er ansvarlig og ressurser som blir tatt i bruk står i aktivitetsoversikten.

4 ITRASJON

4.1 HENSIKTEN

Hovedfokuset med denne iterasjonen er å forberede seg til presentasjon nummer 2. Dette er en presentasjon som skal ha fokus på det tekniske. I den anledning skal også all relevant dokumentasjon bli skrevet ut og levert til Karoline før onsdag 16.03 kl. 12:00. Hvis vi får tid skal vi fortsette med å forbedre TCP kommunikasjonen.

4.2 MÅ-MÅL

Tabell 4 - Administrative må-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A1-017	Skrive iterasjonsplan C3	2	Svein
A1-016	Skrive iterasjonsrapport C2	2	Vegard
A2-002	Møte med intern veileder på Krona onsdag 17.03.16, Svein skriver møtereferat	4	Alle
A2-006	Presentasjon 2	64	Alle

Tabell 5 - Tekniske må-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar

4.3 BØR-MÅL

Tabell 6 - Administrative bør-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A2-003	Møte med teknisk veileder på Krona torsdag 17.03.16, Svein skriver møtereferat	4	Alle

Tabell 7 - Tekniske bør-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A6-002	Jobbe videre med TCP kommunikasjon	8	Svei n og Vegard

5 AKTIVITETER

Tabell 8 - Aktiviteter

Aktivitet ID	Beskrivelse av aktivitet	Tidsestimat	Ansvar
A1-017	Skrive iterasjonsplan C3	2	Svein
A1-016	Skrive iterasjonsrapport C2	2	Vegard
A2-002	Møte med intern veileder på Krona onsdag 17.03.16, Svein skriver møtereferat	4	Alle
A2-003	Møte med teknisk veileder på Krona torsdag 17.03.16, Svein skriver møtereferat	4	Alle
A2-006	Presentasjon 2	64	Alle
A6-002	Det skal jobbes videre med TCP	8	Svein & Vegard

6 RESSURSER

6.1 PERSONER

- Alexander Schinnes
- Svein Bekkevold
- Carl Peder Jansen
- Vegard Smedsvik

6.2 UTSTYR FRA KDA

- Stasjonær pc med utstyr til hver av medlemmene
- 3D Mus
- Tobii EyeTracker
- Multi Touch Control Wheel

6.3 ANNET UTSTYR

- Private bærbare

6.4 LOKALER

Vi har fått tildelt Utviklings lab – 1204 av Høgskolen i Sørøst-Norge avd. Kongsberg

REFERANSER

Ingen

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

ITERASJONPLAN – C3

04.04.2016 – 17.04.2016

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENTOMFANG 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIE 3
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 INNLEDNING 5
4 ITRASJON 5
4.1 HENSIKTEN 5
4.2 MÅ-MÅL 5
4.3 BØR-MÅL 6
5 AKTIVITETER 7
6 RESSURSER 7
6.1 PERSONER 7
6.2 UTSTYR FRA KDA 7
6.3 ANNET UTSTYR 8
6.4 LOKALER 8
REFERANSER 8

1 DOKUMENTOMFANG

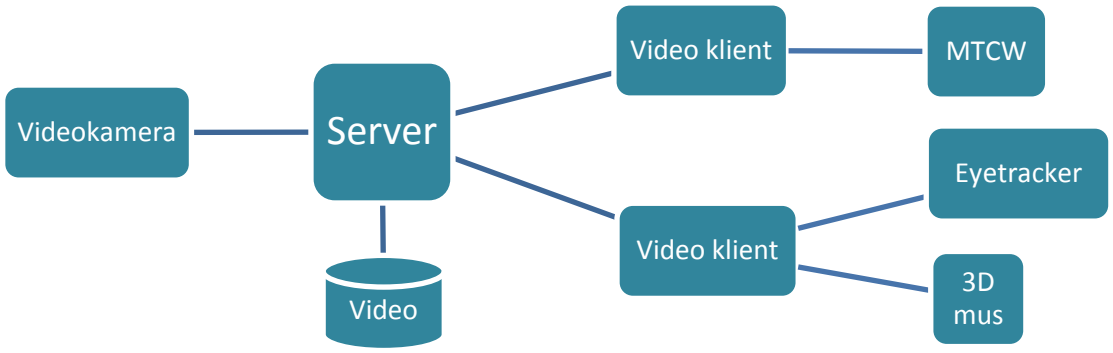
1.1 OVERSIKT

Dette dokumentet inneholder samtlige «må» og «bør» mål. Hvor alle «må» målene er førsteprioritet og «bør» målene vil bli påbegynt etter hvert som «må» målene er oppnådd og ferdigstilt. Dokumentet inneholder også hvilke ressurser, aktiviteter som er relevant for å oppnå målene og hvem som er ansvarlig. Samt en tidsfrist.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 - Systemoversikt

1.3 HISTORIE

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
0.1			Første utkast

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 3
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4

2.4 FIGURER OG BILDER

Figur 1 - Systemoversikt 3

3 INNLEDNING

Hensikten med å ha en iterasjonsplan er å sette mål for neste iterasjon. Denne planen inneholder hvilke ressurser og aktiviteter man trenger for å oppnå målene. Det er også en oversikt over hvilke mål man «må» og «bør» gjennomføre, og hvilke aktiviteter som er relevant for å nå disse. Hvem som er ansvarlig og ressurser som blir tatt i bruk står i aktivitetsoversikten.

4 ITRASJON

4.1 HENSIKTEN

Hovedfokuset i denne iterasjonen blir å utvikle API mot kamera og enheter. Vi har også fått en SDK til MTCW driveren fra ekstern veileder som skal tas i bruk. Det kommer til å bli noen oppdateringer på dokumenter, møter med veiledere og videre utvikling av det interne kommunikasjonssystemet. Endringer fra forrige plan var at presentasjon nr. 2 ble flyttet til over påske.

4.2 MÅ-MÅL

Tabell 4 - Administrative må-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A1-017	Skrive iterasjonsplan C3	2	Svein
A1-016	Skrive iterasjonsrapport C2	2	Vegard
A2-002	Møte med intern veileder på Krona onsdag 06.04, Svein skriver møtereferat	4	Alle
A2-003	Møte med ekstern veileder på Krona torsdag 07.04, Svein skriver møtereferat	4	Alle
A1-002	Oppdatere tidsplan	2	Vegard
A1-008	Oppdatering av risikodokumentet	1	Alex
A1-003	Oppdatere aktivitetsliste	3	CP
A7-005	Oppdatere websiden	1	Svein

Tabell 5 - Tekniske må-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A30-001	Komme videre på GUI	56	CP
A60-001	Grayhill Driver	44	Vegard
A6-002	Kommunikasjon -Styre kamera fra klienten gjennom server	72	Vegard Svein
A70-003 A40-001	Videreutvikle API -kamera -device	56	Alex

4.3 BØR-MÅL

Tabell 6 - Administrative bør-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A5-005	Føre timer fra presentasjonsøving	1	Alle

Tabell 7 - Tekniske bør-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A70-003 A40-001	Videreutvikle API -kamera -device	16	Alex
A30-001	Videreutvikling av GUI	16	CP
A60-001	Grayhill Driver	4	Vegard
A6-002	Kommunikasjon -Styre kamera fra klienten gjennom server	32	Vegard Svein

5 AKTIVITETER

Tabell 8 - Aktiviteter

Aktivitet ID	Beskrivelse av aktivitet	Tidsestimat	Ansvar
A1-017	Skrive iterasjonsplan C3	2	Svein
A1-016	Skrive iterasjonsrapport C2	2	Vegard
A2-002	Møte med intern veileder på Krona onsdag 06.04, Svein skriver møtereferat	4	Alle
A2-003	Møte med ekstern veileder på Krona torsdag 07.04, Svein skriver møtereferat	4	Alle
A1-002	Oppdatere tidsplan	2	Vegard
A1-008	Oppdatering av risikodokumentet	1	Alex
A1-003	Oppdatere aktivitetsliste	3	CP
A7-005	Oppdatere websiden	1	Svein
A30-001	Videre utvikling av GUI	56	CP
A60-001	Grayhill Driver	44	Vegard
A6-002	Kommunikasjon -Styre kamera fra klienten gjennom server	72	Vegard Svein
A70-003 A40-001	Videreutvikle API -kamera -device	56	Alex
A5-005	Føre timer fra presentasjonsøving	1	Alle

6 RESSURSER

6.1 PERSONER

- Alexander Schinnes
- Svein Bekkevold
- Carl Peder Jansen
- Vegard Smedsvik

6.2 UTSTYR FRA KDA

- Stasjonær pc med utstyr til hver av medlemmene
- 3D Mus
- Tobii EyeTracker
- Multi Touch Control Wheel

6.3 ANNET UTSTYR

- Private bærbare

6.4 LOKALER

Vi har fått tildelt Utviklings lab – 1204 av Høgskolen i Sørøst-Norge avd. Kongsberg

REFERANSER

Ingen

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

ITERASJONPLAN – C4

18.04.2016 – 29.04.2016

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENTOMFANG 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIE 3
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 INNLEDNING 5
4 ITRASJON 5
4.1 HENSIKTEN 5
4.2 MÅ-MÅL 5
4.3 BØR-MÅL 6
5 AKTIVITETER 7
6 RESSURSER 7
6.1 PERSONER 7
6.2 UTSTYR FRA KDA 8
6.3 ANNET UTSTYR 8
6.4 LOKALER 8
REFERANSER 8

1 DOKUMENTOMFANG

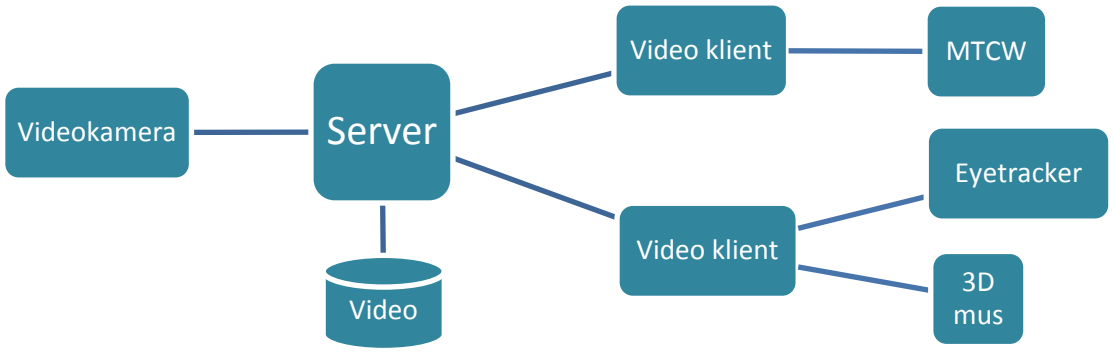
1.1 OVERSIKT

Dette dokumentet inneholder samtlige «må» og «bør» mål. Hvor alle «må» målene er førsteprioritet og «bør» målene vil bli påbegynt etter hvert som «må» målene er oppnådd og ferdigstilt. Dokumentet inneholder også hvilke ressurser, aktiviteter som er relevant for å oppnå målene og hvem som er ansvarlig. Samt en tidsfrist.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 - Systemoversikt

1.3 HISTORIE

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
0.1	15.04.2016	Carl Peder Jansen	Første utkast

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 3
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4

2.4 FIGURER OG BILDER

Figur 1 - Systemoversikt 3

3 INNLEDNING

Hensikten med å ha en iterasjonsplan er å sette mål for neste iterasjon. Denne planen inneholder, hvilke ressurser og aktiviteter man trenger for å oppnå målene. Det er også en oversikt over hvilke mål man «må» og «bør» gjennomføre, og hvilke aktiviteter som er relevant for å nå disse. Hvem som er ansvarlig og ressurser som blir tatt i bruk står i aktivitetsoversikten.

4 ITRASJON

4.1 HENSIKTEN

Hovedmålet i denne iterasjonen blir å oppfylle alle A kravene. Det som gjenstår for å nå dette er fullverdig integrering av inputenhetene og muligheten for lagring og avspilling av video. I den første uken kommer hovedfokuset til å være på inputenhetene, når dette kommer på plass er det videolagringen som er neste på listen. Det vil også bli jobbet med videre utvikling av det interne kommunikasjonssystemet (TCP) og funksjonalitet i GUI .

4.2 MÅ-MÅL

Tabell 4 - Administrative må-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A2-002	Møte med intern veileder 06.04	4	Alle
A2-003	Møte med ekstern veileder 07.04	8	Alle
A5-003	Skrive møtereferat	1	Svein
A1-036	Rapport	4	Vegard
A1-036	Plan	2	Svein

Tabell 5 - Tekniske må-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A30-004	Funksjonalitet i GUI	24	CP
A6-010	Testing	32	Alle
A80-001	Lage Device API (RTO) - Factory - API for hver device(klasse)	24	Alex
A10-006	Lage CameraView	32	Svein Vegard
A10-006	Backend TCP	16	Svein Vegard
A60-004	Integrere MTCW	8	Vegard
A50-004	Integrere 3D Mus	8	CP
A10-003	Lagre video på server	50	Alle
A30-005	Spille av fra server	50	Alle

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A50-004	Integrere 3D Mus	8	CP
A60-004	Integrere MTCW	8	Vegard
A40-004	Integrere Eye Tracker	8	Alex
A6-010	Testing	20	Alle

4.3 BØR-MÅL

Tabell 6 - Administrative bør-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A7-005	Oppdatere websiden	0,5	Svein

Tabell 7 - Tekniske bør-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A30-004	Overlay som rører på seg	4	CP

5 AKTIVITETER

Tabell 8 - Aktiviteter

Aktivitet ID	Beskrivelse av aktivitet	Tidsestimat	Ansvar
A2-002	Møte med intern veileder 06.04	4	Alle
A2-003	Møte med ekstern veileder 07.04	8	Alle
A7-005	Oppdatere websiden	0,5	Svein
A30-004	Funksjonalitet i GUI	24	CP
A6-010	Testing	32	Alle
A80-001	Lage Device API (RTO) - Factory - API for hver device(klasse)	24	Alex
A10-006	Lage CameraView	32	Svein Vegard
A10-006	Backend TCP	16	Svein Vegard
A60-004	Integrere MTCW	8	Vegard
A50-004	Integrere 3D Mus	8	CP
A30-004	Overlay som rører på seg	4	CP
A5-003	Skrive møtereferat	1	Svein
A10-003	Lagre video på server	50	Alle
A30-005	Spille av fra server	50	Alle
A50-004	Integrere 3D Mus	8	CP
A60-004	Integrere MTCW	8	Vegard
A40-004	Integrere Eye Tracker	8	Alex
A6-010	Testing	20	Alle
A1-036	Rapport	4	Vegard
A1-036	Plan	2	Svein

6 RESSURSER

6.1 PERSONER

- Alexander Schinnes
- Svein Bekkevold
- Carl Peder Jansen
- Vegard Smedsvik

6.2 UTSTYR FRA KDA

- Stasjonær pc med utstyr til hver av medlemmene
- 3D Mus
- Tobii EyeTracker
- Multi Touch Control Wheel

6.3 ANNET UTSTYR

- Private bærbare

6.4 LOKALER

Vi har fått tildelt Utviklings lab – 1204 av Høgskolen i Sørøst-Norge avd. Kongsberg

REFERANSER

Ingen

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

ITERASJONPLAN – C5

29.04.2016 – 13.05.2016

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENTOMFANG 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIE 3
2 FIGURER OG TABELLER 3
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 INNLEDNING 5
4 ITRASJON 5
4.1 HENSIKTEN 5
4.2 MÅ-MÅL 5
4.3 BØR-MÅL 6
5 AKTIVITETER 7
6 RESSURSER 7
6.1 PERSONER 7
6.2 UTSTYR FRA KDA 7
6.3 ANNET UTSTYR 7
6.4 LOKALER 8
REFERANSER 8

1 DOKUMENTOMFANG

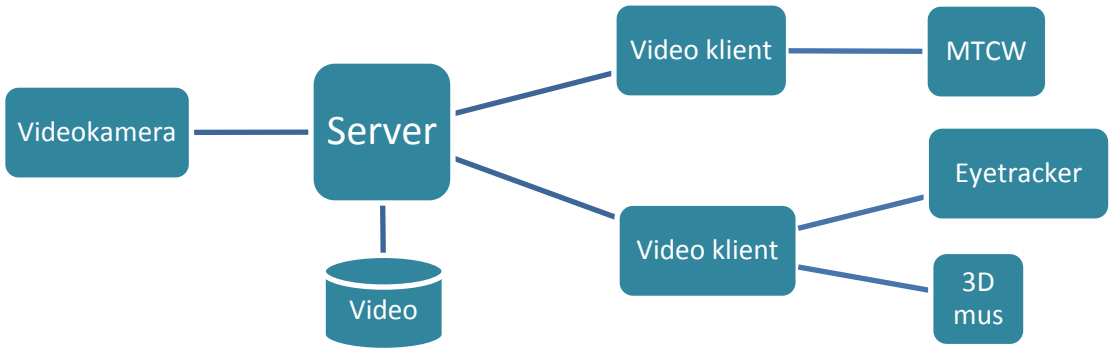
1.1 OVERSIKT

Dette dokumentet inneholder samtlige «må» og «bør» mål. Hvor alle «må» målene er førsteprioritet og «bør» målene vil bli påbegynt etter hvert som «må» målene er oppnådd og ferdigstilt. Dokumentet inneholder også hvilke ressurser, aktiviteter som er relevant for å oppnå målene og hvem som er ansvarlig. Samt en tidsfrist.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 - Systemoversikt

1.3 HISTORIE

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
0.1	03.05.2016	Svein Bekkevold	Første utkast

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 3
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4

2.4 FIGURER OG BILDER

Figur 1 - Systemoversikt 3

3 INNLEDNING

Hensikten med å ha en iterasjonsplan er å sette mål for neste iterasjon. Denne planen inneholder, hvilke ressurser og aktiviteter man trenger for å oppnå målene. Det er også en oversikt over hvilke mål man «må» og «bør» gjennomføre, og hvilke aktiviteter som er relevant for å nå disse. Hvem som er ansvarlig og ressurser som blir tatt i bruk står i aktivitetsoversikten.

4 ITRASJON

4.1 HENSIKTEN

Hovedmålet i denne iterasjonen er å optimalisere videoavspiller og MTCW, implementere funksjoner innenfor kommunikasjon, oppdatering av nettsiden, utføre testing og gjennomgå møter med intern og ekstern sensor.

4.2 MÅ-MÅL

Tabell 4 - Administrative må-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A2-002	Møte med intern veileder 04.05 og 11.05	8	Alle
A2-003	Møte med ekstern veileder 04.05	8	Alle
A5-003	Skrive møtereferat	3	Svein
A7-005	Nettside	16	Svein
A1-035	Dokumentasjon - Analysedokument - Oppdater klassediagram - Kommunikasjon	24	Alle
A1-035	Nye Dokumenter - Brukermanual - Etteranalyse - Systembeskrivelse - Vedlikehold	40	Alle
A1-001	Oppdatere Gantt	4	Vegard
A1-036	Iterasjonsrapport	4	CP

Tabell 5 - Tekniske må-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A10-002	Optimalisering av video på server	32	Alex
A60-003	Optimaliser MTCW	8	Vegard
A80-002	Endre mode på kamera - API - GUI	8	Svein

A30-004	GUI	48	CP
A20-003	Kommunikasjon - Error - Feilmelding - Bruke kamera knapp - Avkobling - Antall klienter som er koblet til server	16	Vegard
A1-035	Dokumentasjon - Analysedokument - Oppdater klassediagram - Kommunikasjon - Systembeskrivelse	16	Alle
A80-002	Lage en Swipe funksjon	8	Svein
A6-010	Testing/Debugging	52	Alle
A7-008	Generell optimalisering	20	Alle

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A10-002	Optimalisering av video på server	32	Alex
A60-003	Optimaliser MTCW	8	Vegard
A80-002	Endre mode på kamera	8	Svein
A30-004	GUI	48	CP
A20-003	Kommunikasjon	16	Vegard
A1-035	Dokumentasjon + Nye dokumenter	80	Alle
A80-002	Lage en Swipe funksjon	8	Svein
A7-005	Nettside	16	Svein
A6-010	Testing/Debugging	52	Alle
A2-002	Møte Intern	4	Alle
A2-003	Møte ekstern	8	Alle
A1-036	Iterasjonsrapport	4	CP
A7-008	Generell optimalisering	20	Alle

4.3 BØR-MÅL

Tabell 6 - Administrative bør-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A7-005	Oppdatere websiden	1	Svein
A5-003	Skrive møtereferat	3	Svein

Tabell 7 - Tekniske bør-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar

5 AKTIVITETER

Tabell 8 - Aktiviteter

Aktivitet ID	Beskrivelse av aktivitet	Tidsestimat	Ansvar
A10-002	Optimalisering av video på server	32	Alex
A60-003	Optimaliser MTCW	8	Vegard
A80-002	Endre mode på kamera	8	Svein
A30-004	GUI	48	CP
A20-003	Kommunikasjon	16	Vegard
A1-034	Dokumentasjon + Nye dokumenter	80	Alle
A80-002	Lage en Swipe funksjon	8	Svein
A7-005	Nettside	16	Svein
A6-010	Testing/Debugging	52	Alle
A2-002	Møte Intern	4	Alle
A2-003	Møte ekstern	8	Alle
A1-036	Iterasjonsrapport	4	CP
A7-008	Generell optimalisering	20	Alle
A10-002	Optimalisering av video på server	32	Alex

6 RESSURSER

6.1 PERSONER

- Alexander Schinnes
- Svein Bekkevold
- Carl Peder Jansen
- Vegard Smedsvik

6.2 UTSTYR FRA KDA

- Stasjonær pc med utstyr til hver av medlemmene
- 3D Mus
- Tobii EyeTracker
- Multi Touch Control Wheel

6.3 ANNET UTSTYR

- Private bærbare

6.4 LOKALER

Vi har fått tildelt Utviklings lab – 1204 av Høgskolen i Sørøst-Norge avd. Kongsberg

REFERANSER

Ingen

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

ITERASJONPLAN – T1

16.05.2016 – 02.06.2016

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENTOMFANG 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIE 3
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 INNLEDNING 5
4 ITERASJON 5
4.1 HENSIKTEN 5
4.2 MÅ-MÅL 5
4.3 BØR-MÅL 6
5 AKTIVITETER 6
6 RESSURSER 6
6.1 PERSONER 6
6.2 UTSTYR FRA KDA 7
6.3 ANNET UTSTYR 7
6.4 LOKALER 7
REFERANSER 7

1 DOKUMENTOMFANG

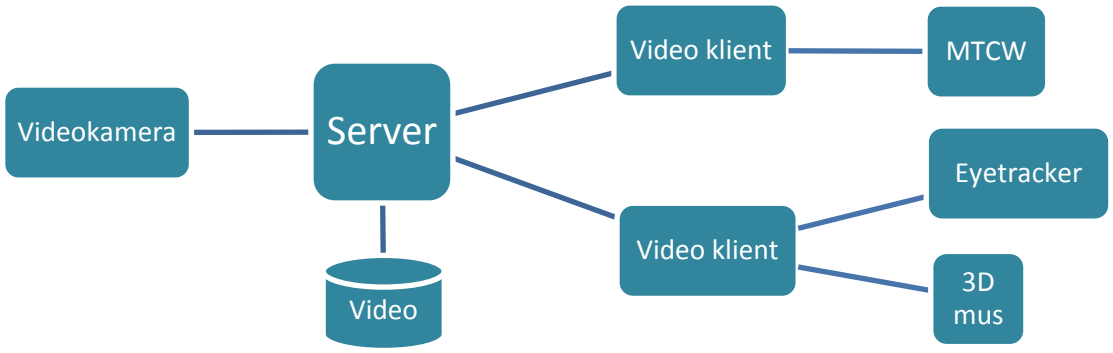
1.1 OVERSIKT

Dette dokumentet inneholder samtlige «må» og «bør» mål. Hvor alle «må» målene er førsteprioritet og «bør» målene vil bli påbegynt etter hvert som «må» målene er oppnådd og ferdigstilt. Dokumentet inneholder også hvilke ressurser, aktiviteter som er relevant for å oppnå målene og hvem som er ansvarlig. Samt en tidsfrist.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 - Systemoversikt

1.3 HISTORIE

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
1.0	18.05.2016	Alexander Schinnes	Første utkast

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 3
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4

2.4 FIGURER OG BILDER

Figur 1 - Systemoversikt 3

3 INNLEDNING

Hensikten med å ha en iterasjonsplan er å sette mål for neste iterasjon. Denne planen inneholder hvordan, hvilke ressurser og aktiviteter som man trenger for å oppnå målene. Det er også en oversikt over hvilke mål man «må» og «bør» gjennomføre, og hvilke aktiviteter som er relevant for å nå disse. Hvem som er ansvarlig og ressurser som blir tatt i bruk står i aktivitetsoversikten.

4 ITERASJON

4.1 HENSIKTEN

Vi skal nå over i «transition» fasen og har i hovedsak to punkter som det er fokus på i denne iterasjonen. Den første uken vil vi fokusere på å få alle dokumenter ferdige til innlevering den 23. mai. Uken deretter vil vi ha fokus på å lage og forberede alt rundt presentasjon tre.

4.2 MÅ-MÅL

Tabell 4 - Administrative må-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A7-005	Nettside	18	Svein
A2-003	Møte eksternt, walk through av koden + dokumentasjon	12	Alle
A2-002	Møte internt, gjennomgang av dokumentasjon	8	Alle
A1-035	Dokumentasjon - Ferdigstilling av dokumentasjon	34	Alle
A1-036	Iterasjonsplan	2	Alex
A1-036	Iterasjonsrapport	2	CP
A2-007	Forberedelser til presentasjon 3	144	Alle

Tabell 5 - Tekniske må-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A5-008	Kode - Alle skriver forklaring på de funksjoner de har laget - Generell opprydding i koden	32	Alle
A6-010	Testing/Debugging	4	CP

4.3 BØR-MÅL

Tabell 6 - Administrative bør-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar

Tabell 7 - Tekniske bør-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar

5 AKTIVITETER

Tabell 8 - Aktiviteter

Aktivitet ID	Beskrivelse av aktivitet	Tidsestimat	Ansvar
A1-035	Dokumentasjon - Ferdigstilling av dokumentasjon	34	Alle
A1-036	Iterasjonsplan	2	Alex
A1-036	Iterasjonsrapport	2	CP
A2-002	Møte internt, gjennomgang av dokumentasjon	8	Alle
A2-003	Møte eksternt, walk through av koden + dokumentasjon	12	Alle
A2-007	Forberedelser til presentasjon 3	144	Alle
A5-008	Kode - Alle skriver forklaring på de funksjoner de har laget - Generell opprydding i koden	32	Alle
A6-010	Testing/Debugging	4	CP
A7-005	Nettside	18	Svein

6 RESSURSER

6.1 PERSONER

- Alexander Schinnes
- Svein Bekkevold
- Carl Peder Jansen
- Vegard Smedsvik

6.2 UTSTYR FRA KDA

- Stasjonær pc med utstyr til hver av medlemmene
- 3D Mus
- Tobii EyeTracker
- Multi Touch Control Wheel

6.3 ANNET UTSTYR

- Private bærbare

6.4 LOKALER

Vi har fått tildelt Utviklings lab – 1204 av Høgskolen i Sørøst-Norge avd. Kongsberg

REFERANSER

Ingen

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

ITERASJONSRAPPORT – I1

01.01.2016 – 02.02.2016

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENT 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIE 3
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 STATUS 5
3.1 MÅ-MÅL 5
3.2 BØR-MÅL 5
4 AVGJØRELSER 6
4.1 TIDSPLAN 6
4.2 MØTESTRUKTUR 6
4.3 PROSJEKTMODELL 6
5 KONKLUSJON 6
6 TIDSBRUK 7
6.1 PLANLAGT 7
6.2 FAKTISK 8
6.3 KONKLUSJON 8
REFERANSER 8

1 DOKUMENT

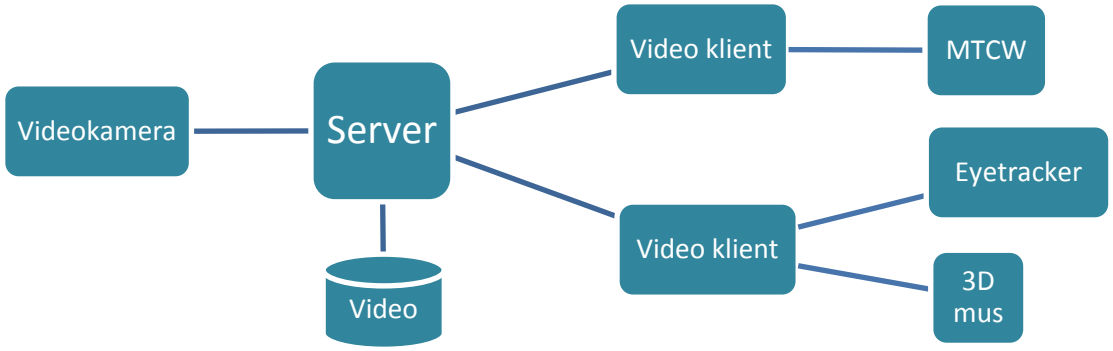
1.1 OVERSIKT

Dette er et dokument som inneholder en iterasjonsrapport for den første iterasjonen i prosjektet. Iterasjon nummer en gikk fra 13.01.2016 til 31.01.2016. Dette dokumentet inneholder status, konklusjoner og avgjørelser som ble tatt i denne perioden.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 – Systemoversikt

1.3 HISTORIE

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
1.0	05.02.2016	Vegard Smedsvik	Første versjon

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Mmachine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 3
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4
Tabell 4 - Administrative må-mål 5
Tabell 5 - Tekniske må-mål 5
Tabell 6 - Administrative bør-mål 5
Tabell 7 - Tekniske bør-mål 6
Tabell 8 - Aktiviteter 7

2.4 FIGURER OG BILDER

Figur 1 – Systemoversikt 3

3 STATUS

3.1 MÅ-MÅL

Tabell 3 og 4 viser alle målene vi har satt oss i denne perioden samt status. Vi har oppnådd alle «må-målene» vi har satt oss, bortsett fra å ferdigstille kontrakten. Alt fra vår side når det gjelder kontrakten er gjort, så vi flytter aktiviteter en iterasjon fram.

Tabell 4 - Administrative må-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet	Status
Lage prosjektplan	Alexsander	20	A1-001	
Lage kravspesifikasjon	Alexsander, Vegard	20	A1-005	
Lage testspesifikasjon	Svein, Carl Peder	30	A1-006	
Første utkast av tidsplan	Vegard	20	A1-002	
Lag generelle Use-Caser til systemet	Carl Peder Jansen	30	A1-004	
Møter med KDA	Alle	20	A2-001	
Møter med intern veileder	Alle	8	A2-002	
Valg av prosjektmodell	Gruppen	10	A1-001	
Sette av dato for 1 presentasjon	Vegard	0.5	A2-005	
Ferdigstille kontrakt	Svein	4	A1-007	
Lage risikovurdering	Vegard, Alexander	10	A1-008	
Lage iterasjonsplan I1	Vegard	6	A1-010	
Lage iterasjonsplan I2	Vegard	4	A1-011	
Lage oppfølgingsdokument	Svein	3	A5-002	
Lage aktivitetsplan	Svein	10	A1-003	

Tabell 5 - Tekniske må-mål

Bekrivelse	Ansvar	Tidsestimat	Aktivitet	Status
Kurs i QT	Vegard, Alexander	2	A4-001	
Kurs i C++	Vegard, Alexander	4	A4-002	
Få og hente utstyr fra KDA	Alle	8	A5-001	

3.2 BØR-MÅL

Tabell 5 og 6 viser alle målene vi bør gjennomføre. Vi fikk ikke gjort alt vi burde. I denne perioden greide ikke vi å finne et passende navn til prosjektet. Dette medførte at vi kan lage en logo og standard malen ikke kan ferdigstilles 100%.

Tabell 6 - Administrative bør-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet	Status
Lage standard mal	Alexsander	4	A1-009	
Finne på navn	Gruppen	2	A5-005	

Tabell 7 - Tekniske bør-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet	Status
Prototype av systemet	Vegard	4	A6-009	
Lage en logo til prosjektet	Vegard	4	A5-005	

4 AVGJØRELSER

4.1 TIDSPLAN

Utvikling av tidsplan skal skje i Excel. Dette ble anbefalt av flere, blant annet KDA som mente at det var en grei måte å gjøre det på. Vi kom med forslaget om å bruke Microsoft Project, men det krever at en bruker litt tid på å sette seg inn i programmet.

4.2 MØTESTRUKTUR

Vi ble enige om å ha ukentlige møter med intern veileder hver onsdag kl 11:00.
Vi ble enige om å ha ukentlige møter med ekstern veileder torsdager kl 12:30.

4.3 PROSJEKTMODELL

Etter en del research og samtaler med intern og eksterne veiledere har vi kommet fram til at Unified Processes er en modell som passer vårt prosjekt. Samtidig vil vi ha ta et element fra Scrum modellen, hvor vi har et scrum møte hver arbeidsdag kl 09:00. På dette møte svarer alle medlemmene på: Hva de har gjort siden forrige gang, hva de skal fortsette med og om de har noen problemer. Dette mener vi er en god måte å sørge for en jevn utvikling, siden vi kan raskt oppfatte om noen står fast.

5 KONKLUSJON

Denne iterasjonen har hatt som hovedfokus å samle mest mulig informasjon rundt prosjektet, hvilken modell vi skal bruke og administrative dokumenter. Vi har satt oss noen mål hvor grunnleggende dokumenter skal være utviklet i første utkast. Ukentlige møter med ekstern og intern veileder vil hjelpe oss å holde et jevnt tempo, samt at vi kan få hjelp om vi begynner å utvikle dokumenter, funksjoner eller Software som ikke er relevant.

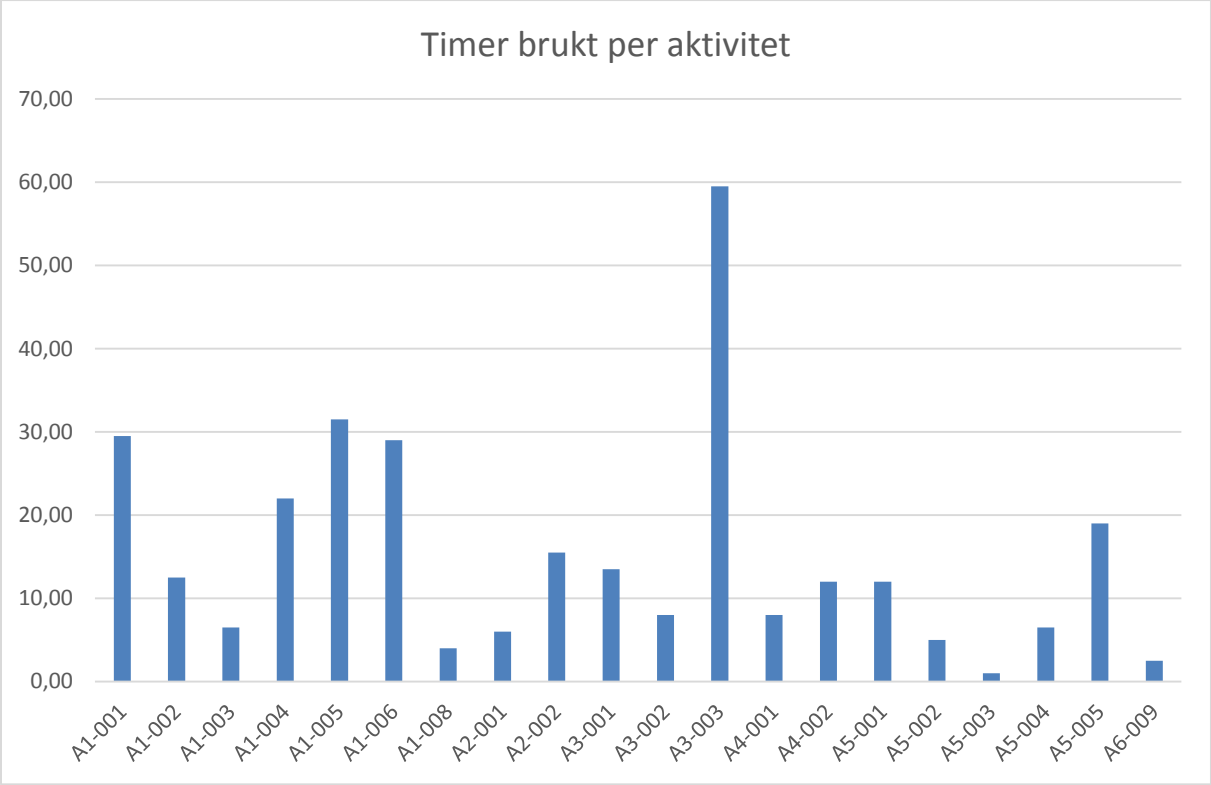
6 TIDSBRUK

6.1 PLANLAGT

Tabell 8 - Aktiviteter

#	Aktivitet	Ansvarlig	Kort beskrivelse	Tidsestimat
A1-001	Prosjektplan	Svein	Første utkast	20
A1-002	Tidsplan	Alexander	Første utkast	30
A1-003	Aktivitetsplan	Svein	Lage	10
A1-004	Use Case	Carl Peder	Lage overordnet system	30
A1-005	Kravspesifikasjon	Alexander, Vegard	Første utkast	20
A1-006	Testspesifikasjon	Carl Peder, Svein	Første utkast	30
A1-007	Kontrakter	Svein	Sørge for at de oppdatert	20
A1-008	Risikodokument	Alexander	Lage	10
A1-009	Lage dokumentmaler	Alexander	Lage en standard mal	4
A1-010	Iterasjons plan	Vegard	Lage mal og lage plan for første iterasjon	6
A1-011	Iterasjons plan	Vegard	Lage iterasjons plan for andre iterasjon	4
A2-001	Kongsberg Gruppen	Alle	Møte med KDA	14
A2-002	Veiledermøte internt	Alle	Møte med intern veileder	8
A3-001	Software	Alle	Research på Software	10
A3-002	Hardware	Alle	Research på Hardware	10
A3-003	Prosjektstyring	Alle	Research på prosjektstyring	50
A4-001	Kurs i QT	Vegard, Alexander	Kurs for resten av gruppen	2
A4-002	Kurs i C++	Vegard, Alexander	Kurs for resten av gruppen	4
A5-001	Hente og installere utstyr	Alle	Hente utstyr hos KDA, skaffe rom samt installere oss der.	8
A5-002	Oppfølgnings-dokument	Svein	Lage mal og oppdatere oppfølgningsdokument	4
A5-003	Skrive møtereferat	Svein	Skrive møtereferat etter alle møter	8
A5-004	Forberedelser til møter	Vegard	Sende innkalling til alle møter	4
A5-005	Annet	Alle	Diskutere navn og logo	3
A6-009	Prototype	Vegard	Lage en enkel prototype	4

6.2 FAKTISK



Figur 2- Tidsbruk

6.3 KONKLUSJON

Denne perioden hadde vi fokus på å gjøre en del research i forhold til prosjektmodell, software, hardware og dokumentasjon. I tillegg flyttet vi inn i en lab på skolen hvor vi kan arbeide. Vi har blitt enige om å jobbe strukturert hver onsdag, torsdag og fredag fra 08:00 – 16:00.

REFERANSER

Ingen

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

ITERASJONSRAPPORT – I2

01.02.2016 – 02.02.2016

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENTOMFANG 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIE 3
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 STATUS 5
3.1 MÅ-MÅL 5
3.2 BØR-MÅL 5
4 AVGJØRELSER 6
4.1 NAVN 6
4.2 LOGO 6
5 KONKLUSJON 6
6 TIDSBRUK 6
6.1 PLANLAGT 6
6.2 FAKTISK 7
6.3 KONKLUSJON 8
REFERANSER 8

1 DOKUMENTOMFANG

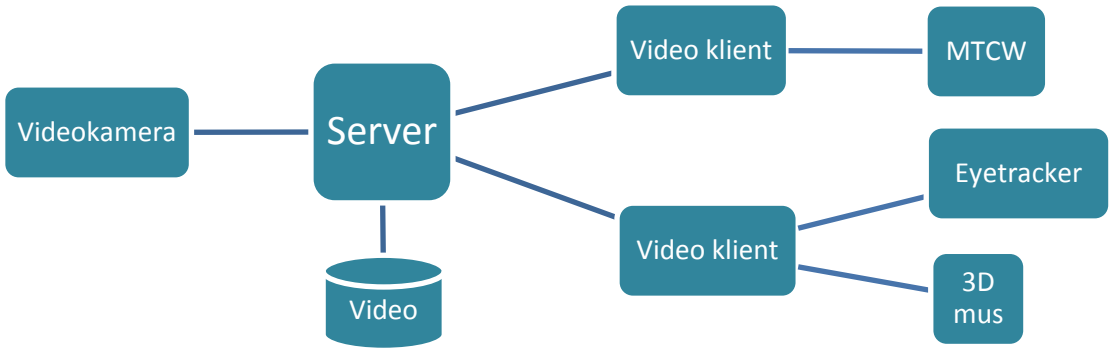
1.1 OVERSIKT

Dette er et dokument som inneholder en iterasjonsrapport for den første iterasjonen i prosjektet. Iterasjon nummer en gikk fra 01.02.2016 til 02.02.2016. Dette dokumentet inneholder status, konklusjoner og avgjørelser som ble tatt i denne perioden.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 - Systemoversikt

1.3 HISTORIE

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
0.1			Første utkast

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Mmachine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 3
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4
Tabell 4 - Administrative må-mål 5
Tabell 5 - Tekniske må-mål 5
Tabell 6 - Administrative bør-mål 5
Tabell 7 - Tekniske bør-mål 5
Tabell 8 - Aktiviteter 6
Tabell 9 - Godkjent estimering 7

2.4 FIGURER OG BILDER

Figur 1 - Systemoversikt 3

3 STATUS

3.1 MÅ-MÅL

Tabell 4 og 5 viser alle målene vi har satt oss i denne perioden samt status. Vi har oppnådd alle «må-målene» vi har satt oss, bortsett fra å lage en iterasjonsplan nummer 3. Vi har ukes iterasjoner noe som medfører at oppfølgningsdokument samt iterasjonsplan inneholder mye av det samme. Dermed så er det ikke kritisk om vi ikke har skrevet denne. Utsetter den en iterasjon.

Tabell 4 - Administrative må-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet	Status
Oppdatere prosjektplan	Alexander	20	A1-001	
Oppdatere kravspesifikasjon	Alexander	20	A1-005	
Oppdatere testspesifikasjon	Svein, Carl Peder	30	A1-006	
Oppdatere tidsplan	Vegard	20	A1-002	
Møter med KDA	Alle	20	A2-001	
Møter med intern veileder	Alle	8	A2-002	
Ferdigstille kontrakt	Svein	4	A1-007	
Oppdatere risikovurdering	Alexander	10	A1-008	
Lage iterasjonsplan I3	Vegard	3	A1-012	
Lage oppfølgingsdokument	Svein	3	A5-002	
Oppdatere aktivitetsliste	Alexander	10	A1-003	
Lage presentasjon	Vegard	5	A2-006	
Øve til presentasjon	Alle	40	A2-006	
Lage visjonsdokument	Svein	8	A1-030	

Tabell 5 - Tekniske må-mål

Bekrivelse	Ansvar	Tidsestimat	Aktivitet	Status

3.2 BØR-MÅL

Tabell 5 og 6 viser alle målene vi bør gjennomføre. Vi fikk gjort alt vi burde gjøre.

Tabell 6 - Administrative bør-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet	Status
Finne på navn	Alle	2	A5-005	

Tabell 7 - Tekniske bør-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet	Status
Lage en logo	Vegard	2	A5-005	

4 AVGJØRELSER

4.1 NAVN

Vi bestemte navn i denne perioden. Alternativene stod mellom «Every Day Albatros» og «Real Time Observation». Hvor vi gikk for «Real Time Observation».

4.2 LOGO

Logoen ble utviklet kort tid etter navnevalget. Den ser du på forsiden.

5 KONKLUSJON

Siden vi har strukturert prosjektet med ukes iterasjoner, så vil oppfølgningsdokumentet og iterasjonsplanen inneholde mye av den samme informasjonen. Dette betyr at det ikke er kritisk at iterasjonsplanen for iterasjon nummer 3 er laget. Det skal sies at det ikke er bra å havne bak med slik dokumentasjon, fordi dette stjeler mye unødvendig tid.

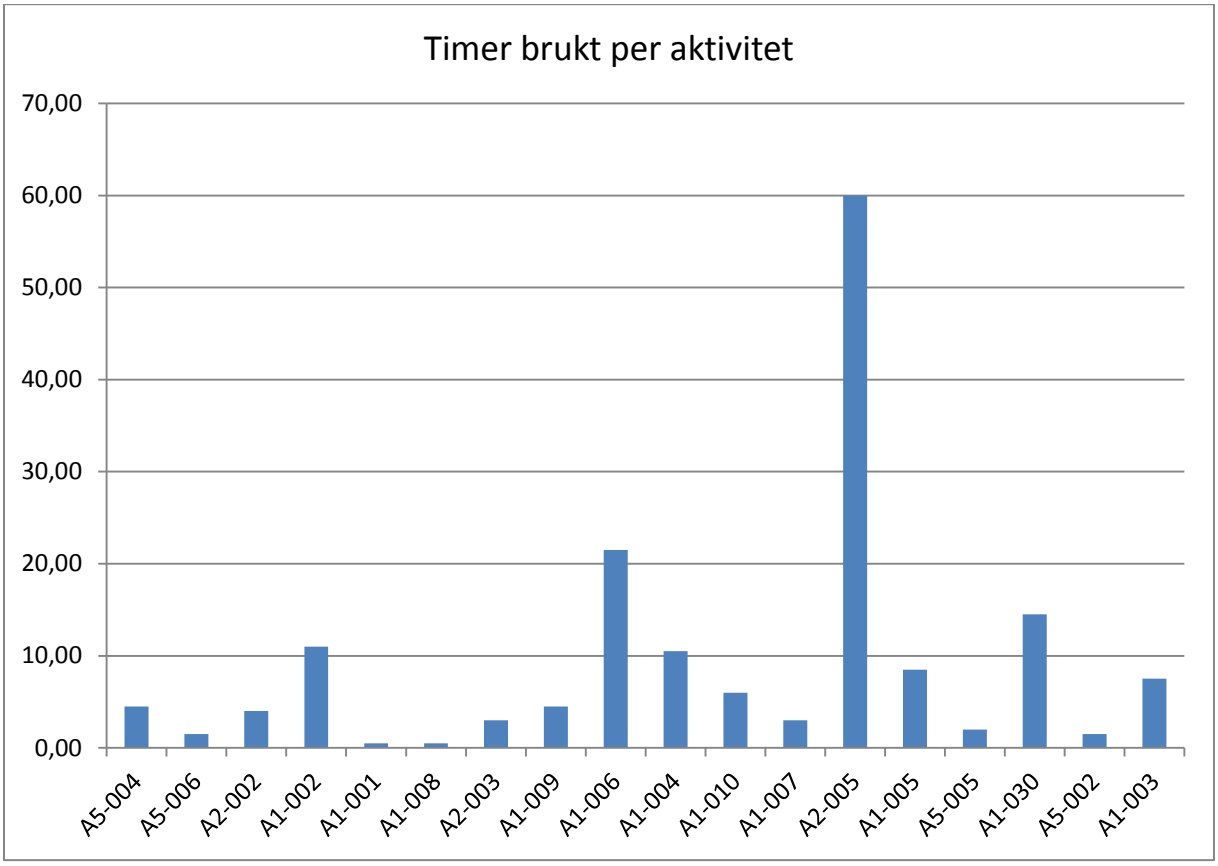
6 TIDSBRUK

6.1 PLANLAGT

Tabell 8 - Aktiviteter

#	Aktivitet	Ansvarlig	Kort beskrivelse	Tidsestimat
A1-001	Prosjektplan	Alexander	Oppdatere prosjektplan	20
A1-002	Tidsplan	Vegard	Oppdatere tidsplan	20
A1-003	Aktivitetsliste	Alexander	Oppdatere aktivitetsliste	10
A1-005	Kravspesifikasjon	Alexander	Oppdatere kravspesifikasjon	20
A1-006	Testspesifikasjon	Svein, Carl Peder	Oppdatere testspesifikasjon	30
A1-007	Kontrakt	Svein	Ferdigstille kontrakt	4
A1-008	Risikodokument	Alexander	Oppdatere risikodokument	10
A1-012	Lage iterasjonsplan 3	Vegard	Lage oppfølgingsdokument	3
A1-030	Visjonsdokument	Svein	Lage visjonsdokument	8
A2-001	Møte med KDA	Alle	Møte med KDA	8
A2-002	Møte intern	Alle	Møte med intern veileder	4
A2-005	Presentasjon	Vegard, Alle	Lage og øve til presentasjon	45
A5-002	Oppfølgnings-dokument	Svein	Lage oppfølgningsdokument for uke 7	3

6.2 FAKTISK



Figur 2 - Tidsbruk

Tabell 9 - Godkjent estimering

#	Forutsatt	Faktisk timer	Estimert timer	Godkjent
A1-001		0,5	20	
A1-002		11	20	
A1-003		7,50	10	
A1-004		10,50		
A1-005		8,50	20	
A1-006		21,50	30	
A1-007		3	4	
A1-008		0,5	10	
A1-009		4,50		
A1-010		6		
A1-012		0	3	
A1-030		14,50	8	
A2-001		0	8	
A2-002		4	4	
A2-005		60	45	
A5-002		1,50	3	
A5-004		4,50		
A5-005		2		
A5-006		1,5		

Tabell 9 sin «Godkjent» kolonne er basert på:

$$\frac{Estimert}{2} \leq Godkjent \leq Estimert \ x \ 2$$

6.3 KONKLUSJON

Vi har mye rødt denne perioden. Dette er ikke fordi vi ikke har oppnådd målene vi har satt oss. Men fordi vi planla for få aktiviteter, samt for mye tid på dem. Det røde kommer av dårlig planlegging, og er en viktig lærdom i dette prosjektet. Vi må sette enda mer presise aktiviteter som oppfyller målene vi har satt oss. Denne perioden ble nok for mye «generelt».

REFERANSER

Ingen

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

ITERASJONSRAPPORT – E1

13.02.2016 – 21.02.2016

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENTOMFANG 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIE 3
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 STATUS 5
3.1 MÅ-MÅL 5
3.2 BØR-MÅL 6
4 AVGJØRELSER 7
5 KONKLUSJON 7
6 TIDSBRUK 7
6.1 PLANLAGT 7
6.2 FAKTISK 8
6.3 KONKLUSJON 9
REFERANSER 9

1 DOKUMENTOMFANG

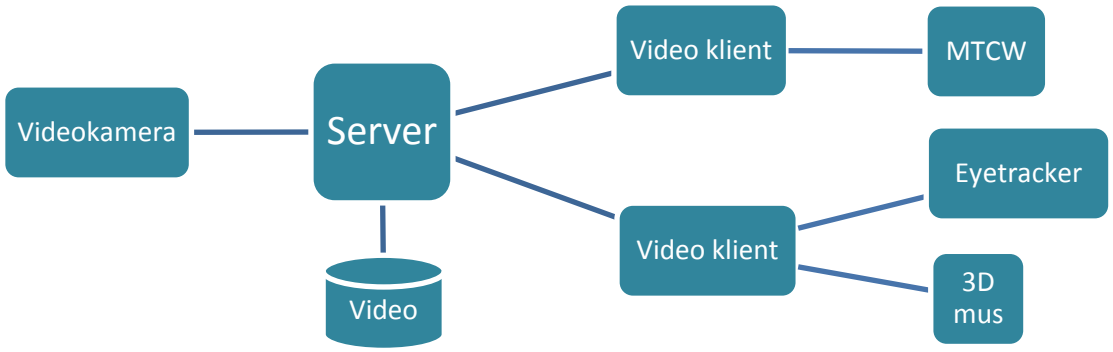
1.1 OVERSIKT

Dette er et dokument som inneholder en iterasjonsrapport for den tredje iterasjonen i prosjektet. Iterasjon nummer tre gikk fra 13.02.16 til 21.02.16 Dette dokumentet inneholder status, konklusjoner og avgjørelser som ble tatt i denne perioden.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 - Systemoversikt

1.3 HISTORIE

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
1.0	19.02.2016	Vegard	Første utkast

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 3
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4
Tabell 4 - Administrative må-mål 5
Tabell 5 - Tekniske må-mål 5
Tabell 6 - Administrative bør-mål 6
Tabell 7 - Tekniske bør-mål 6
Tabell 8 - Aktiviteter 7
Tabell 9 - Godkjent estimering 8

2.4 FIGURER OG BILDER

Figur 1 - Systemoversikt 3
Figur 2 - Tidsbruk 8

3 STATUS

3.1 MÅ-MÅL

Tabell 4 og 5 viser alle målene vi har satt oss i denne perioden samt status. I denne perioden hadde vi noen fokusområder:

1. Vi skulle tilrettelegge neste iterasjon som har hovedfokus innen design og arkitektur.
2. Få opp kommunikasjon via TCP og UDP mellom server og klient.
3. Gjennomgå kravspesifikasjonen og oppdatere administrative dokumenter.

Vi har nådd de fleste målene vi har satt oss, all oppdatering av dokumentasjon er ferdig.
Enkel TCP kommunikasjon er på plass. Mens UDP kommunikasjonen er ikke helt på plass, vi har video fra kamera til server, men får ikke sendt videostrømmen videre til klienter.

Innenfor tilrettelegging av design og arkitektur, så har vi begynt med design.
Implementasjonsdokumentet har vi utsatt til «Construction» fasen, siden det er for tidlig å begynne med den.

Tabell 4 - Administrative må-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet	Status
Skrive iterasjonsplan 3	Vegard	2	A1-012	
Skrive iterasjonsrapport 2	Vegard	4	A1-011	
Skrive iterasjonsplan 4	Vegard	2	A1-013	
Ferdigstille kravspesifikasjon	Alexander	4	A1-005	
Arkitektur	Svein	8	A7-001	
Endre systembeskrivelse	Alexander	4	A1-001	
Designdokument	Carl Peder	8	A1-031	
Implementasjonsdokument	Carl Peder	8	A1-032	

Tabell 5 - Tekniske må-mål

Bekrivelse	Ansvar	Tidsestimat	Aktivitet	Status
Kommunikasjon med TCP	Svein	16	A6-002	
Kommunikasjon med UDP	Alexander	16	A6-003	
Reseach innen det å lage driver	Vegard	16	A60-001	

3.2 BØR-MÅL

Vi har gjennomgått møte med ekstern veileder hvor vi gjennomgikk kravspesifikasjon.
Møte med intern veileder utgikk.

Tabell 6 - Administrative bør-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet	Status
Møte med intern veileder på Krona onsdag 17.02.16, Svein skriver møtereferat	Alle	4	A2-002	
Møte med ekstern veileder på Krona torsdag 18.02.16, Svein skriver møtereferat	Alle	4	A2-003	

Tabell 7 - Tekniske bør-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet	Status

4 AVGJØRELSER

Ingen avgjørelser ble tatt.

5 KONKLUSJON

Vi har fått gjort en del «research» rundt TCP og UDP kommunikasjon, samt fått på plass enkel TCP kommunikasjon mellom server og klient. UDP er litt mer problematisk, det er blitt overført videostrøm fra server til klient, men det ble for mye forsinkelse og det ble ikke brukt videokort. Vi har endret systembeskrivelse etter ønske fra arbeidsgiver og oppdatert kravspesifikasjonen basert på møtet vi hadde med ekstern veileder.

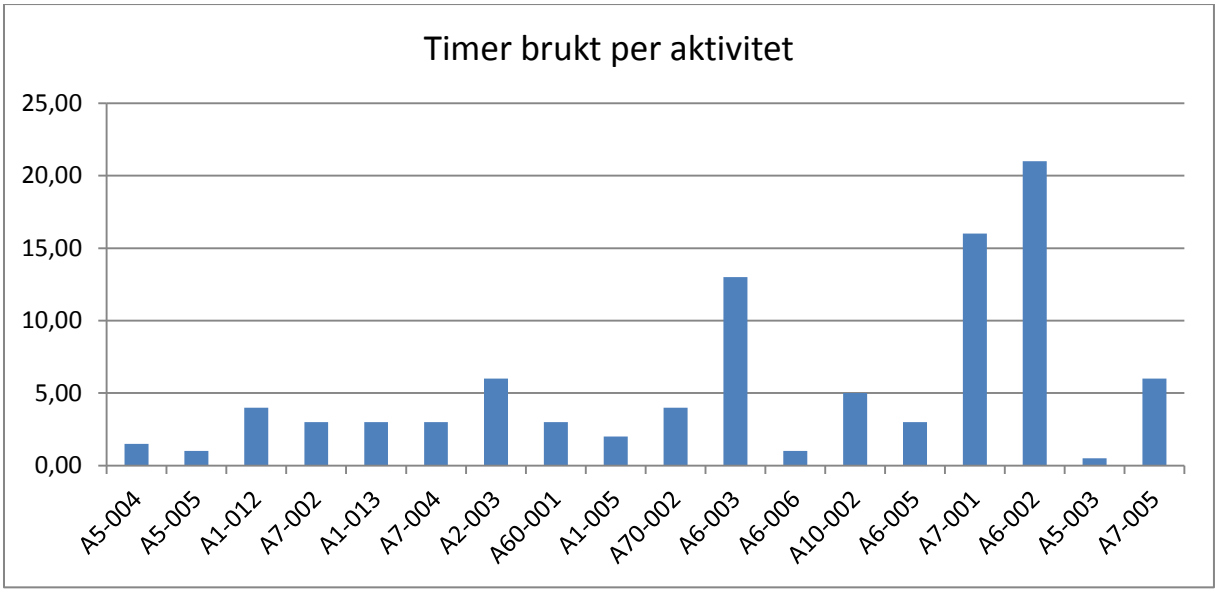
6 TIDSBRUK

6.1 PLANLAGT

Tabell 8 - Aktiviteter

#	Aktivitet	Ansvarlig	Kort beskrivelse	Tidsestimat
A1-001	Prosjektplan	Alexander	Endre systembeskrivelse	4
A1-005	Ferdigstille kravspesifikasjon	Alexander	Ferdigstille kravspesifikasjon	4
A1-011	iterasjonsrapport 2	Vegard	Skrive iterasjonsrapport 2	4
A1-012	iterasjonsplan 3	Vegard	Skrive iterasjonsplan 3	2
A1-013	Skrive iterasjonsplan 4	Vegard	Skrive iterasjonsplan 4	2
A1-031	Designdokument	Carl Peder	Designdokument	8
A1-032	Implementasjon dokument	Carl Peder	Implementasjonsdokument	8
A2-002	Møte intern	Alle	Møte med intern veileder på Krona onsdag 17.02.16, Svein skriver møtereferat	4
A2-003	Møte ekstern	Alle	Møte med ekstern veileder på Krona torsdag 18.02.16, Svein skriver møtereferat	4
A60-001	Driver	Vegard	Research innen det å lage driver	16
A6-002	TCP	Svein	Kommunikasjon med TCP	16
A6-003	UDP	Alexander	Kommunikasjon med UDP	16
A7-001	Arkitektur	Svein	Arkitektur	8

6.2 FAKTISK



Figur 2 - Tidsbruk

Tabell 9 - Godkjent estimering

#	Forutsatt	Faktisk timer	Estimert timer	Godkjent
A1-001			4	
A1-005		2	4	
A1-011			4	
A1-012		4	2	
A1-013		3	2	
A1-031				
A1-032			8	
A2-002			4	
A2-003		6	4	
A5-003		0,5		
A5-004		1,5		
A5-005		1		
A6-002		21	16	
A6-003		13	16	
A6-005		3		
A6-006		1		
A7-001			8	
A7-002		3		
A7-004		3		
A7-005		6		
A10-001		5		
A60-001		3	16	
A70-002		4		

Tabell 9 sin «Godkjent» kolonne er basert på:

$$\frac{Estimert}{2} \leq Godkjent \leq Estimert \ x \ 2$$

6.3 KONKLUSJON

Vi har nok en gang mange aktiviteter som er forutsett og ikke dokumentert i timelistene. Samt en del aktiviteter vi ikke har forutsett, men har jobbet med. Det å planlegge hvilke del-aktiviteter som er nødvendig å gjøre for å fullføre en hovedaktivitet er vanskelig. Vi har greid å gjennomføre hva vi har planlagt.

REFERANSER

Ingen

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

ITERASJONSRAPPORT – E2

21.02.2016 – 28.02.2016

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENTOMFANG 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIE 3
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 STATUS 5
3.1 MÅ-MÅL 5
3.2 BØR-MÅL 6
4 AVGJØRELSER 6
4.1 AVGJØRELSE 6
5 KONKLUSJON 6
6 TIDSBRUK 7
6.1 PLANLAGT 7
6.2 FAKTISK 7
6.3 KONKLUSJON 8
REFERANSER 8

1 DOKUMENTOMFANG

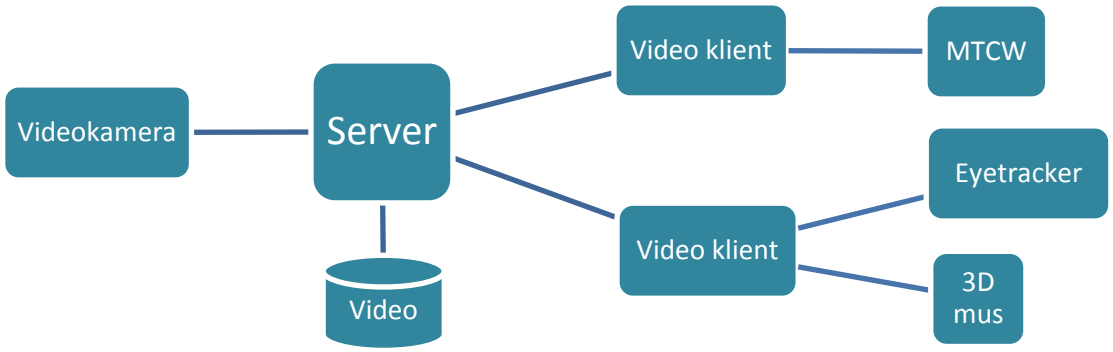
1.1 OVERSIKT

Dette er et dokument som inneholder en iterasjonsrapport for den fjerde iterasjonen i prosjektet. Iterasjon nummer en gikk fra 21.02.16 til 28.02.16. Dette dokumentet inneholder status, konklusjoner og avgjørelser som ble tatt i denne perioden.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 - Systemoversikt

1.3 HISTORIE

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
1.0	26.02.2016	Vegard	Første utkast

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 3
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4
Tabell 4 - Administrative må-mål 5
Tabell 5 - Tekniske må-mål 5
Tabell 6 - Administrative bør-mål 6
Tabell 7 - Tekniske bør-mål 6
Tabell 8 - Aktiviteter 7
Tabell 9 - Godkjent estimering 8

2.4 FIGURER OG BILDER

Figur 1 - Systemoversikt 3
Figur 2 - Tidsbruk 7

3 STATUS

3.1 MÅ-MÅL

Tabell 4 og 5 viser alle målene vi har satt oss i denne perioden samt status. I denne perioden hadde vi noen fokusområder.

1. Vi skulle komme i gang med å legge en grunnleggende arkitektur av hvordan systemet skal bygges opp i form av komponent og klassediagram.
2. Sende videostrøm internt til klienter og motta en videostrøm fra serveren.
3. Gjennomgang av kravspesifikasjon med ekstern veileder og gjøre endringer i forhold til tilbakemelding fra sensor.
4. Lage et utkast til design av software og GUI

Vi har nådd de mål som er satt for fjerde iterasjon. Grunnen til at Design og Arkitektur er satt som gul status er fordi beskrivelsen av oppgavene er for diffuse. Vi har laget første utkast til komponent og klassediagram under arkitektur og første utkast til GUI under design. Men dette er egentlig aktiviteter som kommer til å bli endret underveis, så neste gang skal vi formulere bedre hva som ligger i oppgaven.

Det har blitt oppnådd en kommunikasjon mellom klienter og server ved å sende videostrøm. Foreløpig er det VLC vi har brukt som verktøy til dette. Vi har også fått rettet opp småfeil i krav og utbedret det vi fikk som tilbakemelding av sensor etter første presentasjon.

Tabell 4 - Administrative må-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet	Status
Design	CP	32	A7-003	
Skrive iterasjonsplan E3	Svein	2	A1-014	
Skrive iterasjonsrapport E1	Vegard	3	A1-012	
Skrive iterasjonsrapport E2	Svein	3	A1-013	
Arkitektur	Svein & Vegard	32	A7-001	

Tabell 5 - Tekniske må-mål

Bekrivelse	Ansvar	Tidsestimat	Aktivitet	Status

3.2 BØR-MÅL

Tabell 6 - Administrative bør-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet	Status
Møte med intern veileder på Krona onsdag 24.02.16, Svein skriver møtereferat	Alle	4	A2-002	
Møte med ekstern veileder og teknisk veileder på Krona torsdag 23.02.16, Svein skriver møtereferat	Alle	8	A2-003	

Tabell 7 - Tekniske bør-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet	Status
Sende videostrøm til klienter og motta videostrøm fra server	Alexander	16	A10-002, A20-002	

4 AVGJØRELSER

4.1 AVGJØRELSE

Latency: Vi har etter analyser mot latency funnet ut at videokortet som vi har fått utlevert ikke leverer innenfor kravet om 200ms responstid til real time streaming. Basert på analysen har vi i samsvar med oppdragsgiver bestemt at vi går videre i prosjektet og godtar at vi ikke har oppfylt det kravet. Vi skal da dokumentere for hvorfor kravet ikke har blitt møtt.

Dokumentasjon: Oppdragsgiver har bestemt at de ikke vil sette noen regler på hva de krever av dokumentasjon, men at vi skal følge prosjektmodell og forhøre oss med intern veileder ang hva dokumentasjonen skal inneholde.

5 KONKLUSJON

Det har blitt tatt en viktig avgjørelse ang latency som gjør at vi fortsetter prosjektet uten å møte kravet om 200ms latency. Vi har fått til å sende videostrøm til klienter og motta videostrøm fra server. Vi har gjort en del research innen UML og laget førsteutkast til component og class-diagram. Etter møtet med intern veileder har vi fått et par punkt fra sensor som vi har rettet på, samt rettet opp noen krav fra møtet med oppdragsgiver og ekstern veileder.

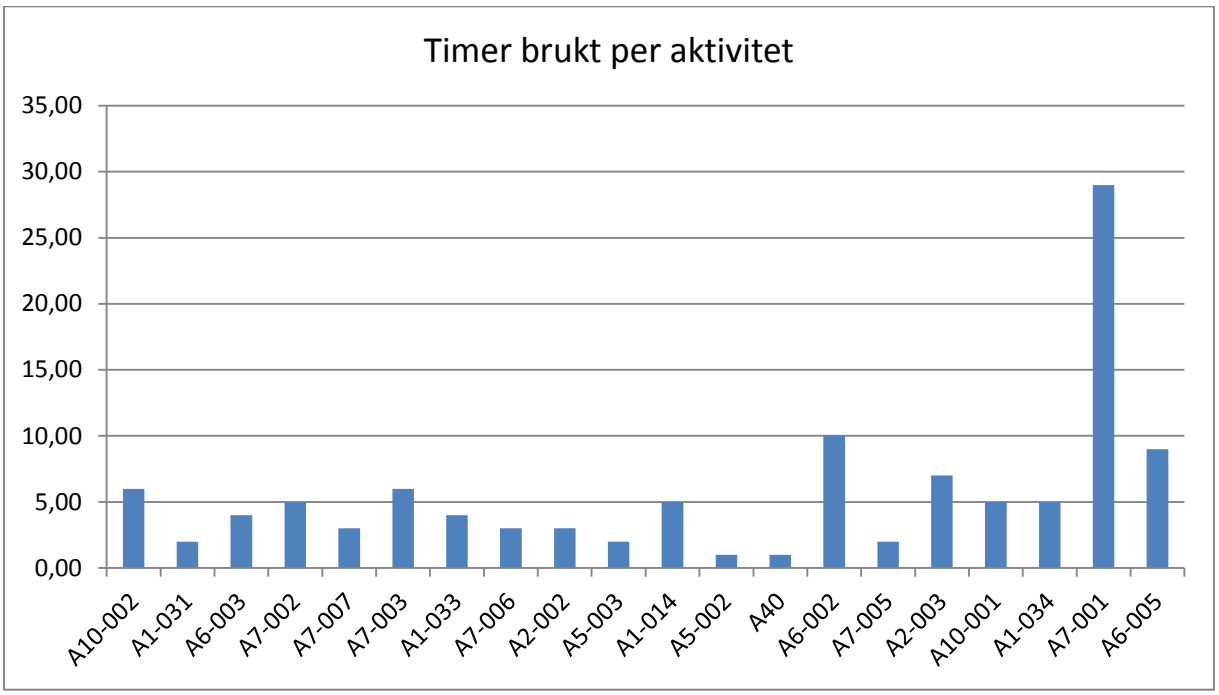
6 TIDSBRUK

6.1 PLANLAGT

Tabell 8 - Aktiviteter

#	Aktivitet	Ansvarlig	Kort beskrivelse	Tidsestimat
A1-012	iterasjonsrapport E1	Vegard	Skrive iterasjonsrapport 3	3
A1-013	iterasjonsrapport E2	Svein	Skrive iterasjonsrapport 4	3
A1-014	iterasjonsplan E3	Svein	Skrive iterasjonsplan 5	2
A2-002	Møte intern	Alle	Møte med intern veileder på Krona onsdag 24.02.16, Svein skriver møtereferat	4
A2-003	Møte ekstern	Alle	Møte med ekstern veileder og teknisk veileder på Krona torsdag 23.02.16, Svein skriver møtereferat	8
A7-003	Design	Carl Peder	Lage første utkast til design av GUI og bistå arkitektur av software	32
A7-001	Arkitektur	Svein & Vegard	Lage første utkast av diagrammer innen UML	32
A10-002, A20-002	Videostrøm	Alexander	Sende videostrøm til klienter og motta videostrøm fra server	16

6.2 FAKTISK



Figur 2 - Tidsbruk

Tabell 9 - Godkjent estimering

#	Forutsatt	Faktisk timer	Estimert timer	Godkjent
A1-012		0	3	
A1-013		0	3	
A1-014		5	2	
A1-031		2		
A1-033		4		
A1-034		5		
A2-002		3	4	
A2-003		7	8	
A5-002		1		
A5-003		2		
A6-002		10		
A6-003		4		
A6-005		9		
A7-001		29	32	
A7-002		5		
A7-003		32	6	
A7-005		2		
A7-006		3		
A7-007		3		
A10-001		5		
A10-002		6	8	
A20-002		0	8	
A40		1		

Tabell 9 sin «Godkjent» kolonne er basert på:

$$\frac{Estimert}{2} \leq Godkjent \leq Estimert \ x \ 2$$

6.3 KONKLUSJON

Vi har nok en gang mange aktiviteter som er forutsatt og ikke dokumentert i timelistene. Samt en del aktiviteter vi ikke har forutsatt, men har jobbet med. Det å planlegge hvilke del-aktiviteter som er nødvendig å gjøre for å fullføre en hoved aktivitet er vanskelig. Vi har greid å gjennomføre hva vi har planlagt.

REFERANSER

ingen

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

ITERASJONSRAPPORT – E3

28.02.2016 – 06.03.2016

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENTOMFANG 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIE 3
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 STATUS 4
3.1 MÅ-MÅL 4
3.2 BØR-MÅL 6
4 AVGJØRELSER 6
4.1 AVGJØRELSE 6
5 KONKLUSJON 6
6 TIDSBRUK 7
6.1 PLANLAGT 7
6.2 FAKTISK 8
6.3 KONKLUSJON 9
REFERANSER 9

1 DOKUMENTOMFANG

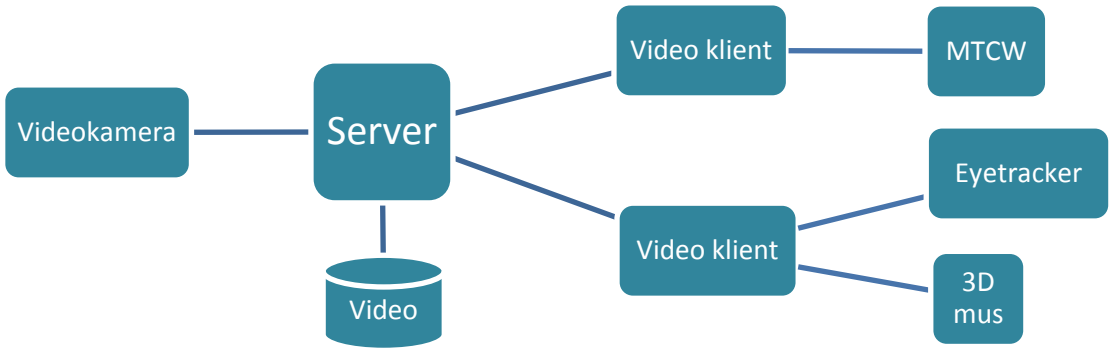
1.1 OVERSIKT

Dette er et dokument som inneholder en iterasjonsrapport for den femte iterasjonen i prosjektet. Iterasjon nummer en gikk fra 28.02.2016 til 06.03.2016. Dette dokumentet inneholder status, konklusjoner og avgjørelser som ble tatt i denne perioden.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 - Systemoversikt

1.3 HISTORIE

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
1.0	04.03.2016	Alexander Schinnes	Første utkast

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 3
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4
Tabell 4 - Administrative må-mål 5
Tabell 5 - Tekniske må-mål 6
Tabell 6 - Administrative bør-mål 6
Tabell 7 - Tekniske bør-mål 6
Tabell 8 - Aktiviteter 7
Tabell 9 - Godkjent estimering 8

2.4 FIGURER OG BILDER

Figur 1 - Systemoversikt 3
Figur 2 - Tidsbruk 8

3 STATUS

3.1 MÅ-MÅL

Tabell 4 og 5 viser alle målene vi har satt oss i denne perioden samt status. I denne perioden hadde vi følgende fokusområder.

1. Ferdigstille førsteutkast av alle dokumenter som er beregnet å være ferdig i løpet av Elaboration fasen, samt oppdatere tidligere dokumenter som f.eks. tidsplan.
2. Kunne sende direkte videostrøm fra server og kunne motta denne videostrømmen i en Qt applikasjon.
3. Forbedring av TCP kommunikasjon.

Vi har nådd stort sett de fleste målene vi har satt oss. Vi har laget førsteutkast av designdokument som inneholder diagrammer for klasse, komponenter, sekvenser og lignende. Kravspesifikasjon og use case er også oppdatert. Grunnlaget i form av mye design og analyse er nå gjennomført slik at vi er klar til å begynne på Construction fasen.

Det har også blitt jobbet en del med å få på plass en hel kommunikasjonslinje på videostrømmen fra server til klient. Dette har vi nå fått, selv om vi fremdeles har noen utfordringer knyttet til stor forsinkelse på ca 3 sekunder. Vår foreløpige linje går ut på at videokortet mottar video fra kameraet. OBS (Open Broadcast Software) plukker opp denne videostrømmen og sender den videre til Unreal Media Server (RTMP-protokoll) for at videostrømmen skal kunne transporteres ut på nettverket. Klientene mottar så denne nettverksstrømmen i Qt og avspiller den ved hjelp av Vlclib. Forbedringer vil foregå i neste iterasjon.

Det jobbes fremdeles med å forbedre TCP kommunikasjonen for overføring av datameldinger. Vi vil fortsette med å forsøke å sende og motta flere variabler ved hjelp av struct eller klasse.

Tabell 4 - Administrative må-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet	Status
Skrive iterasjonsplan C1	Carl Peder	3	A1-015	
Skrive iterasjonsrapport E3	Alexander	3	A1-014	
Møte med intern veileder på Krona onsdag 03.03.16, Svein skriver møtereferat	Alle	4	A2-002	
Møte med teknisk veileder på Krona torsdag 04.03.16, Svein skriver møtereferat	Alle	4	A2-003	Møtet utgikk
Oppdatere Gantt og tidsplan	Vegard	4	A1-002	
Kontakte Grayhill ang driver og kode	Vegard	2	A60-001	
Skrive første utkast til designdokument som skal inneholde class, component, GUI og sekvens diagram. (Få med tekstbeskrivelse i sekvensdiagrammet)	Vegard	12	A1-031	
Use case. Oppdatering av sekvensbeskrivelser	Carl Peder	4	A1-004	
Oppdatere nettside	Svein	2	A7-005	
Analysedokument mot videostrøm	Alexander	8	A1-034	

Tabell 5 - Tekniske må-mål

Bekrivelse	Ansvar	Tidsestimat	Aktivitet	Status
TCP. Forbedring av basic kommunikasjon. Research mot struct	Svein	16	A6-002	
+	Alexander	16	A10-005	
Spille av video fra videoserver i Qt	Carl Peder	16	A10-003	

3.2 BØR-MÅL

Tabell 6 - Administrative bør-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet	Status

Tabell 7 - Tekniske bør-mål

Beskrivelse	Ansvar	Tidsestimat	Aktivitet	Status

4 AVGJØRELSER

4.1 AVGJØRELSE

Vi har satt en dato for når 2. presentasjon skal avholdes. Dette blir 18. mars kl. 13:30. Formøte vil bli holdt kl. 13:00, mens ettermøte vil bli holdt kl. 14:00.

5 KONKLUSJON

Konklusjonen er at vi så langt ligger bra an i forhold til tidsplanen og det vi har beregnet å gjøre. Vi har gjort alt vi skal gjøre på dokumentasjon og planleggings siden. Vi har gjort mye design, analyse og research på forskjellige løsninger, hardware og software. I tillegg har vi analysert og begynt med noen hovedelementer på software, som blant annet videokommunikasjon. Videre kommer vi til å fortsette med den planen som er lagt.

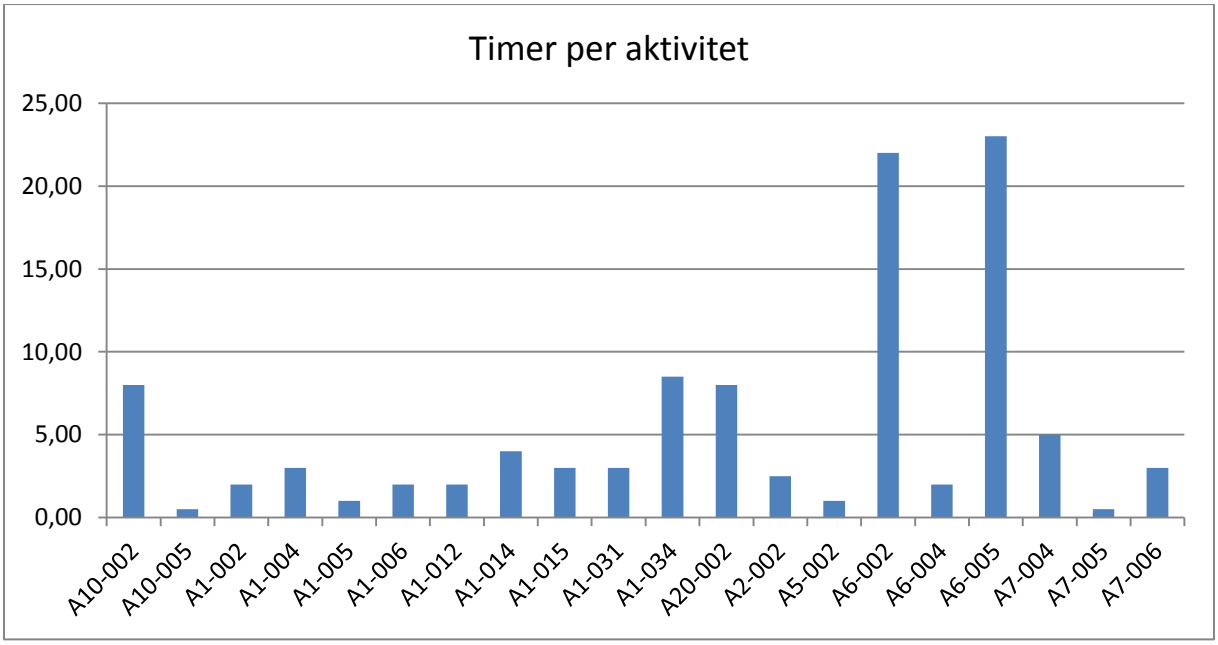
6 TIDSBRUK

6.1 PLANLAGT

Tabell 8 - Aktiviteter

#	Aktivitet	Ansvarlig	Kort beskrivelse	Tidsestimat
A10-003	Videoserver	Carl Peder	Spille av video fra videoserver i Qt	16
A10-005	Videoserver	Alexander	Videoserver feilsøking software, kontakte auresius og prøve å finne best løsning	16
A1-002	Gantt	Vegard	Oppdatere Gantt og tidsplan	4
A1-004	Use Case	Carl Peder	Use case. Oppdatering av sekvensbeskrivelser	4
A1-014	Skrive iterasjonsrapport E3	Alexander	Skrive iterasjonsrapport 5	3
A1-015	Skrive iterasjonsplan C1	Carl Peder	Skrive iterasjonsplan 6	3
A1-031	Designdokument	Vegard	Skrive første utkast til design dokument som skal inneholde class, component, GUI og sekvens diagram. (Få med tekstbeskrivelse i sekvensdiagrammet)	12
A1-034	Videostrøm	Alexander	Analysedokument mot videostrøm	8
A2-002	Veiledermøte intern	Alle	Møte med intern veileder på Krona onsdag 03.03.16, Svein skriver møtereferat	4
A2-003	Veiledermøte ekstern	Alle	Møte med teknisk veileder på Krona torsdag 04.03.16, Svein skriver møtereferat	8
A60-001	Grayhill	Vegard	Kontakte Grayhill ang driver og kode	2
A6-002	TCP	Svein	TCP. Forbedring av basic kommunikasjon. Research mot struct	16
A7-005	Hjemmeside	Svein	Oppdatere hjemmeside	2

6.2 FAKTISK



Figur 2 - Tidsbruk

Tabell 9 - Godkjent estimering

#	Forutsatt	Faktisk timer	Estimert timer	Godkjent
A1-002		2	4	
A1-004		4	4	
A1-005		1		
A1-006		2		
A1-012		2		
A1-014		4	3	
A1-015		3	3	
A1-031		3	12	
A1-034		8,5	8	
A2-002		2,5	4	
A2-003		0	8	
A5-002		1		
A6-002		22	16	
A6-004		2		
A6-005		23		
A7-004		5		
A7-005		0,5	2	
A7-006		3		
A10-002		8		
A10-003		0	16	
A10-005		0,5	16	
A20-002		8		
A60-001		0	2	

Tabell 9 sin «Godkjent» kolonne er basert på:

$$\frac{Estimert}{2} \leq Godkjent \leq Estimert\ x\ 2$$

6.3 KONKLUSJON

Ut ifra figur 2 kan vi se at vi har brukt mest tid på å få på plass TCP kommunikasjon for overføring av data mellom klient og server, også har det gått med mye tid til å integrere VLC-lib i Qt applikasjonen. I tabell 9 kan vi se en del røde felt som stort sett kommer av planlagt arbeid på feil aktiviteter. Her har vi planlagt riktige arbeidsoppgaver, men de har blitt noe feilplassert med tanke på hvilken aktivitet de hører under. Dette er også grunnen til at det er noen røde felt på estimeringen.

Vi skal forbedre dette til neste iterasjon, men kan samtidig konkludere med at vi har gjort alle planlagte arbeidsoppgaver og ligger fremdeles godt an i forhold til tidsplanen.

REFERANSER

Ingen

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

ITERASJON RAPPORT – C1

07.03.2016 – 13.03.2016

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENTOMFANG 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIE 3
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 STATUS 4
3.1 MÅ-MÅL 4
3.2 BØR-MÅL 5
4 AVGJØRELSER 6
4.1 AVGJØRELSE 6
5 KONKLUSJON 6
6 TIDSBRUK 6
6.1 PLANLAGT 6
6.2 FAKTISK 7
6.3 KONKLUSJON 8
REFERANSER 8

1 DOKUMENTOMFANG

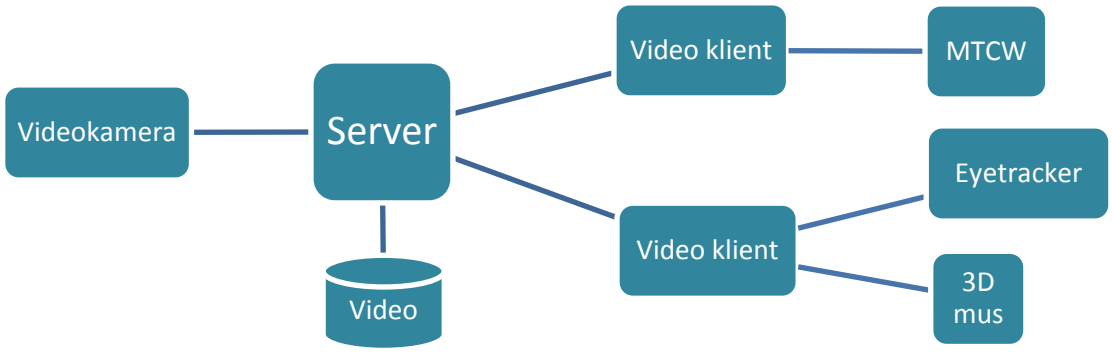
1.1 OVERSIKT

Dette er et dokument som inneholder en iterasjonsrapport for den første iterasjonen i prosjektets construction fase. Iterasjonen gikk fra 07.03.16 til 13.03.16. Dette dokumentet inneholder status, konklusjoner og avgjørelser som ble tatt i denne perioden.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 - Systemoversikt

1.3 HISTORIE

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
1.0	11.03.2016	Carl Peder Jansen	Første utkast

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 3
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4
Tabell 4 - Administrative må-mål 5
Tabell 5 - Tekniske må-mål 5
Tabell 6 - Administrative bør-mål 5
Tabell 7 - Tekniske bør-mål 5
Tabell 8 - Aktiviteter 6
Tabell 9 - Godkjent estimering 7

2.4 FIGURER OG BILDER

Figur 1 - Systemoversikt 3
Figur 2 – Tidsbruk 7

3 STATUS

3.1 MÅ-MÅL

Tabell 4 og 5 viser alle målene vi har satt oss i denne perioden samt nåværende status. I denne perioden har fokusområdene vært:

1. Gjennomgang av alle dokumenter og ferdigstilling av disse.
2. Forberedelse til andre presentasjon.
3. TCP kommunikasjon mellom klient og server.

Tabell 4 - Administrative må-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar	Status
A1-016	Skrive iterasjonsplan C2	2	Vegard	
A1-015	Skrive iterasjonsrapport C1	2	CP	
A2-002	Møte med intern veileder på Krona onsdag 09.03.16, Svein skriver møtereferat	4	Alle	Møtet utgikk
A2-003	Møte med teknisk veileder på Krona torsdag 10.03.16, Svein skriver møtereferat	8	Alle	Møtet utgikk
A1-035	Gjennomgå all dokumentasjon i henhold til Karolines tilbakemeldinger -skrivefeil -referanser -gruppe 2 på alle forsider -riktig format -kjører alle dokumenter i minst versjon 1.0	20	Alle	
A5-004	Sende innkallinger til presentasjon 2	0,5	Vegard	
A1-035	Oppdatere systembeskrivelse i produserte dokumenter (husk prosjektplan)	2	Alexander	
A2-006	Koordinere med Argos ang presentasjonen	0,5	Alexander	

Tabell 5 - Tekniske må-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar	Status
A7-004	Sette opp prosjektets struktur i TortoiseHg	4	Vegard	

3.2 BØR-MÅL

Tabell 6 - Administrative bør-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar	Status

Tabell 7 - Tekniske bør-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar	Status
A6-002	Kunne sende en struct/class fra server til klient ved hjelp av ha TCP.	26	Svein & Vegard	
A30-001	Lage en qt applikasjon med enkel web browser uten GUI slik at vi kan styre kamera (visualisering av illustrasjon) + videoavspiller	30	Alexander og CP	

4 AVGJØRELSER

4.1 AVGJØRELSE

Da det viste seg at ekstern sensor ikke hadde mulighet til å møte opp til andre presentasjonen tok vi kontakt med intern sensor (Karoline). Her ble vi enige om at dette ville gå fint så lenge vi filmer en gjennomgang av presentasjonen. Vi har også bestemt oss for å endre lengden på iterasjonene i construction fasen til 2 uker. Årsaken til dette er at vi ser at 1 ukes iterasjoner blir litt liten tid per iterasjon og at tiden vi bruker på å skrive iterasjonsplaner og rapporter ble noe høyere enn først antatt.

5 KONKLUSJON

Vi ligger fortsatt godt an i forhold til tidsplan og det vi har beregnet å gjøre. Det er nå blitt gjort en nøye gjennomgang av alle dokumenter slik at alt er oppdatert og klart til presentasjon. Vi har begynt å forberede oss til selve presentasjonen. Vi har også kommet litt lengere i TCP kommunikasjon

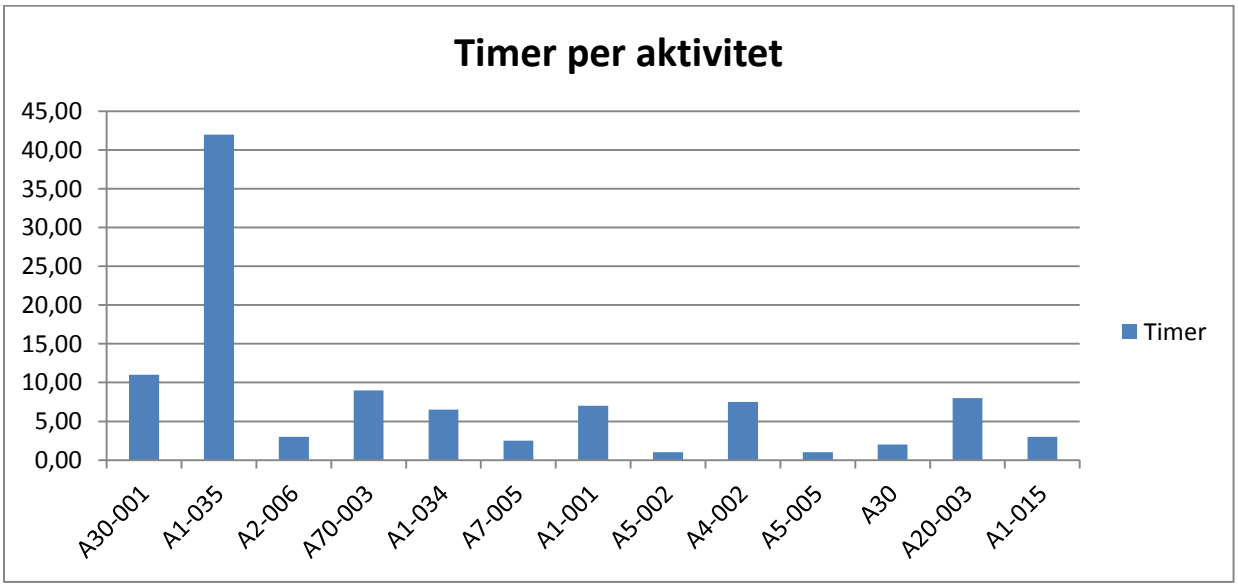
6 TIDSBRUK

6.1 PLANLAGT

Tabell 8 - Aktiviteter

Aktivitet ID	Beskrivelse av aktivitet	Tidsestimat	Ansvar
Aktivitet ID	Beskrivelse av aktivitet	Tidsestimat	Ansvar
A1-016	Skrive iterasjonsplan C2	2	Vegard
A1-015	Skrive iterasjonsrapport C1	2	CP
A2-002	Møte med intern veileder på Krona onsdag 09.03.16, Svein skriver møtereferat	4	Alle
A2-003	Møte med teknisk veileder på Krona torsdag 10.03.16, Svein skriver møtereferat	8	Alle
A1-035	Gjennomgå all dokumentasjon i henhold til Karolines tilbakemeldinger -skrivefeil -referanser -gruppe 2 på alle forsider -riktig format -kjører alle dokumenter i minst versjon 1.0	20	Alle
A5-004	Sende innkallinger	0,5	Vegard
A1-035	Oppdatere systembeskrivelse i produserte dokumenter (husk prosjektplan)	2	Alexander
A2-006	Koordinere med Argos ang presentasjonen	0,5	Alexander
A6-002	Det skal jobbes videre med TCP Vi skal kunne sende en struct/class fra server til klient	26	Svein & Vegard
A30-001	Lage en qt applikasjon med enkel web browser uten GUI slik at vi kan styre kamera (visualisering av illustrasjon) + videoavspiller	30	Alexander og CP
A7-004	Sette opp prosjektets struktur i TortoiseHg	4	Vegard

6.2 FAKTISK



Figur 2 – Tidsbruk

GRAF OVER TIMER

Tabell 9 - Godkjent estimering

#	Forutsatt	Faktisk timer	Estimert timer	Godkjent
A1-015		3	2	
A1-016			2	
A1-035		50	20	
A2-002			4	Møte

				Utgikk
A2-003			8	Møte utgikk
A2-006		3	0,5	
A30-001		11	30	
A5-004			0,5	
A6-002			26	
A7-004			4	
A70-003		9		
A1-034		6,5		
A7-005		2,5		
A1-001		7		
A5-002		1		
A4-002		7,5		
A5-005		1		
A20-003		8		

Tabell 9 sin «Godkjent» kolonne er basert på:

$$\frac{Estimert}{2} \leq Godkjent \leq Estimert \ x \ 2$$

6.3 KONKLUSJON

Vi ser vi bommet en god del på tidsestimatet på gjennomgangen av dokumentene. Vi føler likevel at det var viktig å gjøre en nøye gjennomgang av disse. Det er mange røde aktiviteter i tabell 9, årsaken til dette er en kombinasjon av feilestimeringen av dokumentgjennomgangen og hvordan hver enkelt gruppe medlem har registret timene sine. Det er spesielt mange av de røde aktivitetene som burde vært registret under aktivitet A1-035 (oppdatering av dokumenter). Dette er noe vi må bli mye flinkere på.

Når det er sagt føler vi likevel vi har fått gjort det meste vi satt oss som mål denne uken.

REFERANSER

Ingen

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

ITERASJONRAPPORT – C2

14.03.2016 – 20.03.2016

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENTOMFANG 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIE 3
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 STATUS 4
3.1 MÅ-MÅL 4
3.2 BØR-MÅL 5
4 AVGJØRELSER 6
5 KONKLUSJON 6
6 TIDSBRUK 7
6.1 PLANLAGT 7
6.2 FAKTISK 7
6.3 KONKLUSJON 8
REFERANSER 8

1 DOKUMENTOMFANG

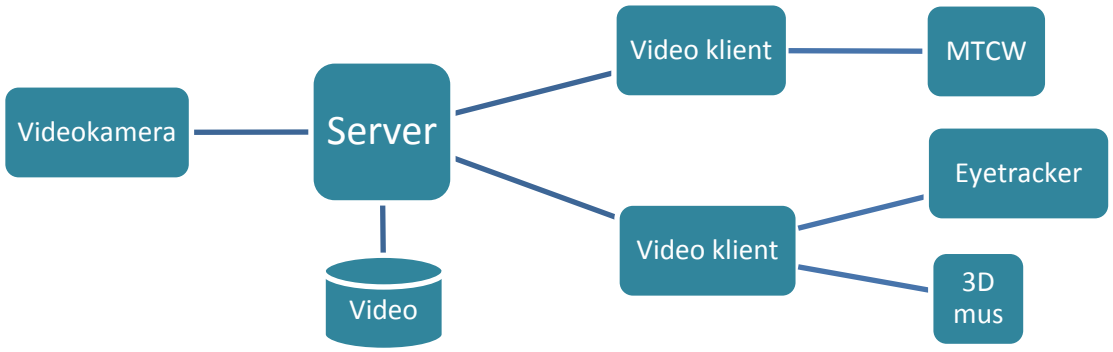
1.1 OVERSIKT

Dette er et dokument som inneholder en iterasjonsrapport for den syvende iterasjonen i prosjektet. Iterasjon nummer en gikk fra 14.03.2016 til 20.02.2016 Dette dokumentet inneholder status, konklusjoner og avgjørelser som ble tatt i denne perioden.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 - Systemoversikt

1.3 HISTORIE

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
1.0	18.03.2016	Vegard	Første utkast

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 3
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4
Tabell 4 - Administrative må-mål 5
Tabell 5 - Tekniske må-mål 5
Tabell 6 - Administrative bør-mål 5
Tabell 7 - Tekniske bør-mål 5
Tabell 8 - Aktiviteter 7
Tabell 9 - Godkjent estimering 7

2.4 FIGURER OG BILDER

Figur 1 - Systemoversikt 3
Figur 2 - Tidsbruk 7

3 STATUS

3.1 MÅ-MÅL

Tabell 4 - Administrative må-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A1-017	Skrive iterasjonsplan C3	2	Svein
A1-016	Skrive iterasjonsrapport C2	2	Vegard
A2-002	Møte med intern veileder på Krona onsdag 17.03.16, Svein skriver møtereferat	4	Alle
A2-006	Presentasjon 2	64	Alle

Tabell 5 - Tekniske må-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar

3.2 BØR-MÅL

Tabell 6 - Administrative bør-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A2-003	Møte med teknisk veileder på Krona torsdag 17.03.16, Svein skriver møtereferat	4	Alle

Tabell 7 - Tekniske bør-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A6-002	Jobbe videre med TCP kommunikasjon	8	Svei n og Vegard

4 AVGJØRELSER

Ingen avgjørelser ble tatt denne iterasjonen.

5 KONKLUSJON

Det ble litt endinger i planen, pga sykdom. Vi skulle egentlig ha vår andre presentasjon fredag 18.03.2016. Denne utgikk fordi Rune(Ekstern veileder) ble syk.

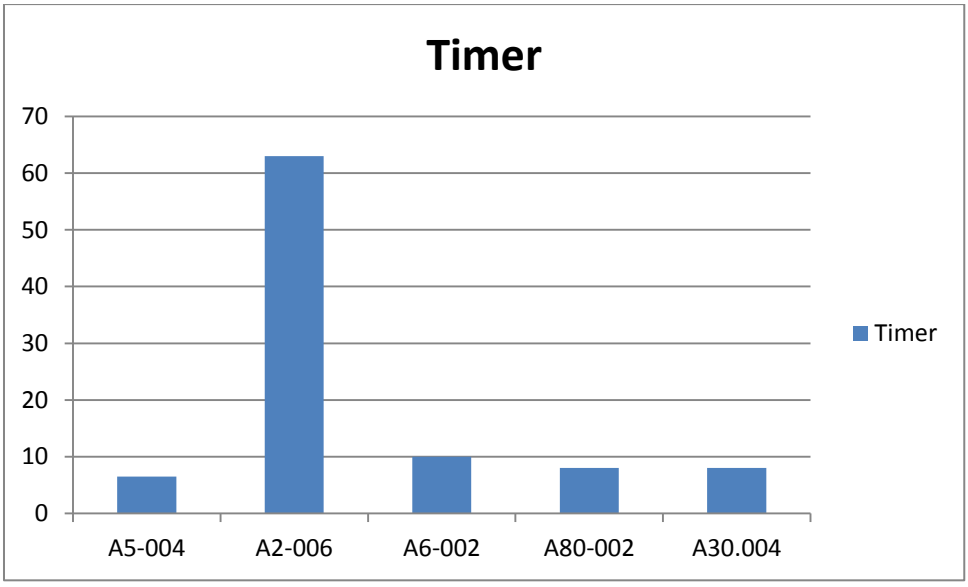
6 TIDSBRUK

6.1 PLANLAGT

Tabell 8 - Aktiviteter

Aktivitet ID	Beskrivelse av aktivitet	Tidsestimat	Ansvar
A1-017	Skrive iterasjonsplan C3	2	Svein
A1-016	Skrive iterasjonsrapport C2	2	Vegard
A2-002	Møte med intern veileder på Krona onsdag 17.03.16, Svein skriver møtereferat	4	Alle
A2-003	Møte med teknisk veileder på Krona torsdag 17.03.16, Svein skriver møtereferat	4	Alle
A2-006	Presentasjon 2	64	Alle
A6-002	Det skal jobbes videre med TCP	8	Svein & Vegard

6.2 FAKTISK



Figur 2 - Tidsbruk

Tabell 9 - Godkjent estimering

#	Forutsatt	Faktisk timer	Estimert timer	Godkjent
A1-016			2	
A1-017			2	
A2-002			4	
A2-003			4	
A2-006		63	64	
A5-004		6,5		
A6-002		10	8	
A30-004		8		
A80-002		8		

Tabell 9 sin «Godkjent» kolonne er basert på:

$$\frac{Estimert}{2} \leq Godkjent \leq Estimert \ x \ 2$$

6.3 KONKLUSJON

Vi har truffet greit med timene denne gangen. Vi har forberedt oss mye til presentasjonene. Siden vi ikke fikk presentert på fredagen, så har det oppstått noen andre aktiviteter som ikke var planlagt. Dette er helt greit, siden det er vanskelig å forutsi sykdom.

REFERANSER

Ingen

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

ITERASJONRAPPORT – C3

04.04.2016 – 17.04.2016

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENTOMFANG 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.3 HISTORIE 3
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 STATUS 4
3.1 MÅ-MÅL 4
3.2 BØR-MÅL 5
4 AVGJØRELSER 6
4.1 AVGJØRELSE 6
5 KONKLUSJON 6
6 TIDSBRUK 6
6.1 PLANLAGT 6
6.2 FAKTISK 7
6.3 KONKLUSJON 8
REFERANSER 8

1 DOKUMENTOMFANG

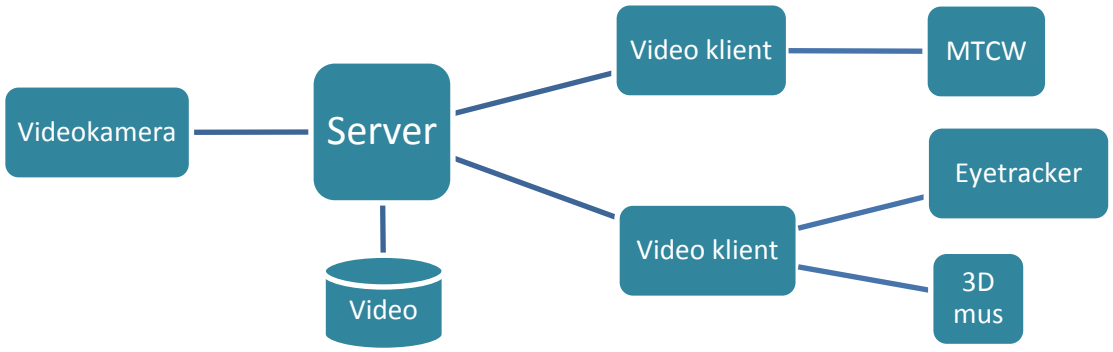
1.1 OVERSIKT

Dette er et dokument som inneholder en iterasjonsrapport for den åttende iterasjonen i prosjektet. Iterasjonen gikk fra 04.04.2016 – 17.04.2016. Dette dokumentet inneholder status, konklusjoner og avgjørelser som ble tatt i denne perioden.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 - Systemoversikt

1.3 HISTORIE

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
0.1	21.04.2016	Alexander Schinnes	Første utkast

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 3
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4
Tabell 4 - Administrative må-mål 5
Tabell 5 - Tekniske må-mål 5
Tabell 6 - Administrative bør-mål 5
Tabell 7 - Tekniske bør-mål 5
Tabell 8 - Aktiviteter 6
Tabell 9 - Godkjent estimering 7

2.4 FIGURER OG BILDER

Figur 1 - Systemoversikt 3
Figur 2 - Tidsbruk 7

3 STATUS

3.1 MÅ-MÅL

Tabell 4 - Administrative må-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A1-017	Skrive iterasjonsplan C3	2	Svein
A1-016	Skrive iterasjonsrapport C2	2	Vegard
A2-002	Møte med intern veileder på Krona onsdag 06.04, Svein skriver møtereferat	4	Alle
A2-003	Møte med ekstern veileder på Krona torsdag 07.04, Svein skriver møtereferat	4	Alle
A1-002	Oppdatere tidsplan	2	Vegard
A1-008	Oppdatering av risikodokumentet	1	Alex
A1-003	Oppdatere aktivitetsliste	3	CP
A7-005	Oppdatere websiden	1	Svein

Tabell 5 - Tekniske må-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A30-001	Komme videre på GUI	56	CP
A60-001	Grayhill Driver	44	Vegard
A6-002	Kommunikasjon -Styre kamera fra klienten gjennom server	72	Vegard Svein
A70-003 A40-001	Videreutvikle API -kamera -device	56	Alex

3.2 BØR-MÅL

Tabell 6 - Administrative bør-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A5-005	Føre timer fra presentasjonsøving	1	Alle

Tabell 7 - Tekniske bør-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A70-003 A40-001	Videreutvikle API -kamera -device	16	Alex
A30-001	Videreutvikling av GUI	16	CP
A60-001	Grayhill Driver	4	Vegard
A6-002	Kommunikasjon -Styre kamera fra klienten gjennom server	32	Vegard Svein

4 AVGJØRELSER

4.1 AVGJØRELSE

Vi har tatt en avgjørelse som har innflytelse på hvordan brukergrensesnittet kan se ut og benyttes. I begynnelsen var planen å kun vise en videostrøm på en GUI, men vi har nå laget en løsning som gjør at vi kan spille av flere forskjellige kamerastrømmer. Det blir opprettet et camera view objekt som det kan settes individuell videostrøm på.

5 KONKLUSJON

Hovedfokuset denne perioden har vært å utvikle en API mot kamera og enheter, i tillegg til å få flest mulig av devicene til å fungere slik at de kan benyttes i applikasjonen. Vi har også utført noen administrative oppgaver som å oppdatere dokumenter. Kamera API er på plass og fungerer slik den skal. Device API er ikke startet på enda. Vi bestemte oss for å utvikle en skikkelig kamera API, slik at oppsettet kan «kopieres» til device API. Ellers ligger vi veldig bra an i forhold til tidsplan. Vi har fått alle devicene til å fungere, i tillegg til at eyetrackeren har blitt integrert i applikasjonen.

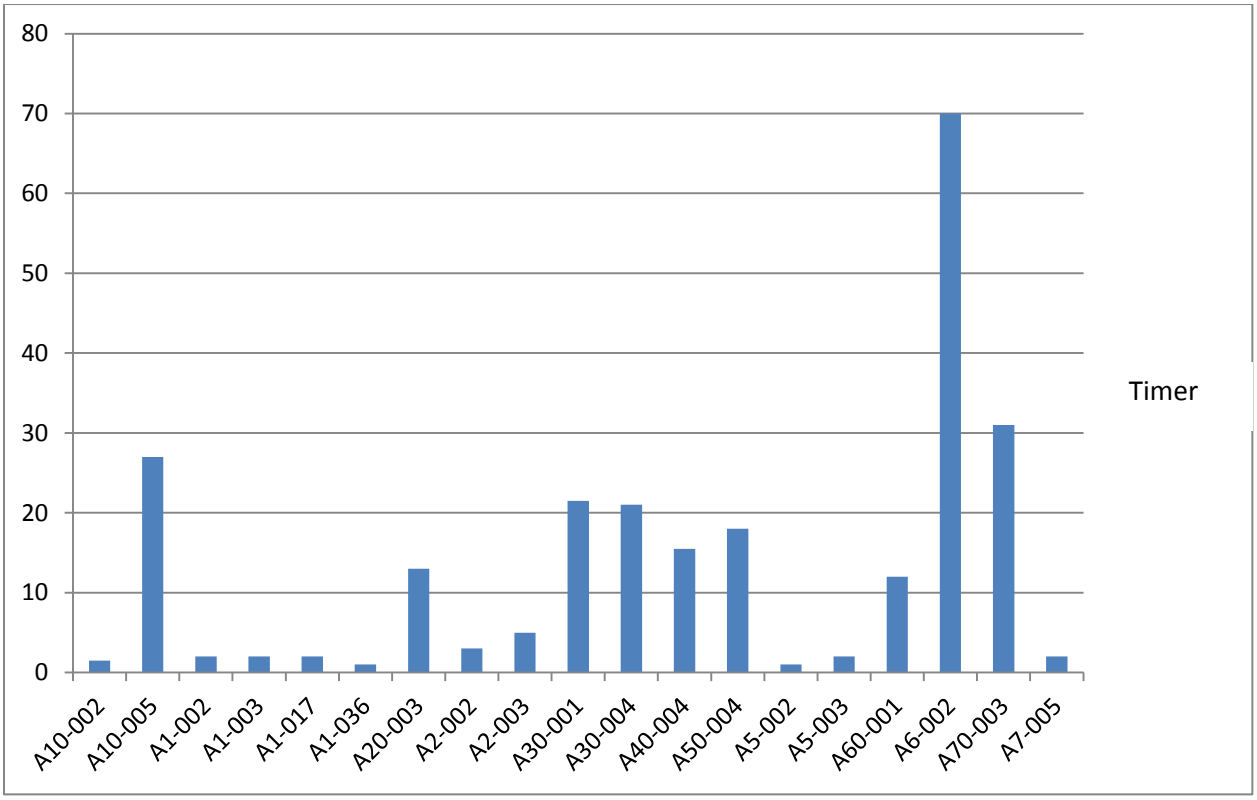
6 TIDSBRUK

6.1 PLANLAGT

Tabell 8 - Aktiviteter

Aktivitet ID	Beskrivelse av aktivitet	Tidsestimat	Ansvar
A1-002	Oppdatere tidsplan	2	Vegard
A1-003	Oppdatere aktivitetsliste	3	CP
A1-008	Oppdatering av risikodokumentet	1	Alex
A1-016	Skrive iterasjonsrapport C2	2	Vegard
A1-017	Skrive iterasjonsplan C3	2	Svein
A2-002	Møte med intern veileder på Krona onsdag 06.04, Svein skriver møtereferat	4	Alle
A2-003	Møte med ekstern veileder på Krona torsdag 07.04, Svein skriver møtereferat	4	Alle
A30-001	Komme videre på GUI	72	CP
A40-001	Videreutvikle device API	12	Alex
A5-005	Føre timer fra presentasjonsøving	1	Alle
A60-001	Grayhill Driver	48	Vegard
A6-002	Kommunikasjon -Styre kamera fra klienten gjennom server	104	Vegard Svein
A70-003	Videreutvikle kamera API	60	Alex
A7-005	Oppdatere websiden	1	Svein

6.2 FAKTISK



Figur 2 - Tidsbruk

GRAF OVER TIMER

Tabell 9 - Godkjent estimering

#	Forutsatt	Faktisk timer	Estimert timer	Godkjent
A10-002		1,5	0	
A10-005		27	0	
A1-002		2	2	
A1-003		2	3	
A1-008		0	1	
A1-016		0	2	
A1-017		2	2	
A1-036		1	0	
A20-003		13	0	
A2-002		3	4	
A2-003		5	4	
A30-001		21,5	72	
A30-004		21	0	
A40-001		0	12	
A40-004		15,5	0	
A50-004		18	0	
A5-002		1	0	
A5-003		2	0	
A5-005		0	1	

A60-001		12	48	
A6-002		70	104	
A70-003		31	60	
A7-005		2	1	

Tabell 9 sin «Godkjent» kolonne er basert på:

$$\frac{Estimert}{2} \leq Godkjent \leq Estimert \ x \ 2$$

6.3 KONKLUSJON

Vi kan se at vi har bommet mye på tidsbruk pr aktivitet for denne iterasjonen. En gjenganger som vi kan se er at vi har estimert for mye tid på en aktivitet, som senere har blitt splittet opp i flere små aktiviteter. Dermed vil vi bomme på for kort tidsbruk samt aktiviteter som ikke er planlagte. Her er det også viktig å nevne at alt som er planlagt av arbeid har blitt utført og vi ligger godt an i forhold til tidsplanen. Men vi ser at vi fremdeles har forbedringspotensialet når det kommer til å planlegge aktiviteter og tidsforbruk.

REFERANSER

Ingen

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

ITERASJONRAPPORT – C4

18.04.2016 – 01.05.2016

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENTOMFANG 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.1 HISTORIE 3
2 REFERANSER, FIGURER OG TABELLER 4
2.1 LISTE OVER REFERANSER 4
2.2 AKRONYMER OG FORKORTELSER 4
2.3 ORDBOK 4
2.4 TABELLER 4
2.5 FIGURER OG BILDER 4
3 STATUS 5
3.1 MÅ-MÅL 5
3.2 BØR-MÅL 5
4 AVGJØRELSER 6
4.1 Database eller GUI 6
4.2 API for enhetene 6
5 KONKLUSJON 6
6 TIDSBRUK 7
6.1 PLANLAGT 7
6.2 FAKTISK 8
6.3 KONKLUSJON 9

1 DOKUMENTOMFANG

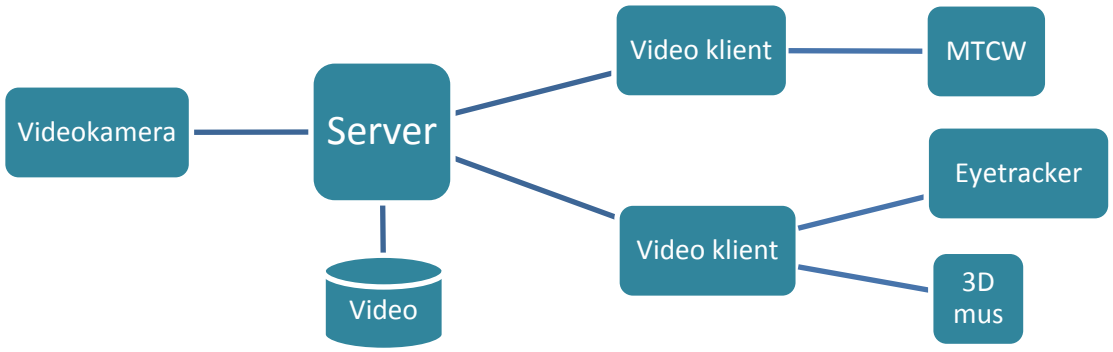
1.1 OVERSIKT

Dette er et dokument som inneholder en iterasjonsrapport for den fjerde iterasjonen i prosjektet. Iterasjon nummer en gikk fra 18.04 til 01.05. Dette dokumentet inneholder status, konklusjoner og avgjørelser som ble tatt i denne perioden.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 - Systemoversikt

1.3 HISTORIE

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
1.0	01.05.16	Vegard	Første utkast

2 REFERANSER, FIGURER OG TABELLER

2.1 LISTE OVER REFERANSER

Ingen

2.2 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.3 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.4 TABELLER

Tabell 1 - Dokumenthistorie 3
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4
Tabell 4 - Administrative må-mål 5
Tabell 5 - Tekniske må-mål 5
Tabell 6 - Administrative bør-mål 5
Tabell 7 - Tekniske bør-mål 5
Tabell 8 - Aktiviteter 7
Tabell 9 - Godkjent estimering 8

2.5 FIGURER OG BILDER

Figur 1 - Systemoversikt 3
Figur 2 - Tidsbruk 8

3 STATUS

3.1 MÅ-MÅL

Tabell 4 - Administrative må-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A2-002	Møte med intern veileder 06.04	4	Alle
A2-003	Møte med ekstern veileder 07.04	8	Alle
A5-003	Skrive møtereferat	1	Svein
A1-036	Rapport	4	Vegard
A1-036	Plan	2	Svein

Tabell 5 - Tekniske må-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A30-004	Funksjonalitet i GUI	24	CP
A6-010	Testing	32	Alle
A80-001	Lage Device API (RTO) - Factory - API for hver device(klasse)	24	Alex
A10-006	Lage CameraView	32	Svein Vegard
A10-006	Backend TCP	16	Svein Vegard
A60-004	Integrere MTCW	8	Vegard
A50-004	Integrere 3D Mus	8	CP
A10-003	Lagre video på server	50	Alle
A30-005	Spille av fra server	50	Alle

3.2 BØR-MÅL

Tabell 6 - Administrative bør-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A7-005	Oppdatere websiden	0,5	Svein

Tabell 7 - Tekniske bør-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A30-004	Overlay som rører på seg	4	CP

4 AVGJØRELSER

Vi har hatt noen diskusjoner denne iterasjonen, det har ført til noen avgjørelser.

4.1 Database eller GUI

Vi har hatt en diskusjon om å lage en database for å lagre tags eller å oppdatere GUIen. Det å lage en database vil gi mer funksjonalitet til applikasjonen, mens det å oppdatere GUIen vil gi brukeren en bedre visuell opplevelse og et inntrykk om at dette er en fullverdig applikasjon. Vi har valgt å oppdatere GUI basert på argumentene over og det å lage en database er et C krav.

4.2 API for enhetene

Vi hadde opprinnelig et krav og et mål om å lage en felles API for alle enhetene. Denne felles APIen skulle vi lage slik at det var lett å legge til en ny enhet for å styre systemet.
Etter å ha snakket med eksern veileder og fått hjelp av teknisk veileder har vi besluttet å ikke lage en slik API. Grunnet at Tobii Eyetracker ikke kulle bli lagt til som en enhet i APIen på lik linje med de andre enhetene. Istedenfor implementerer vi hver enkelt enhet i programmet.

5 KONKLUSJON

Hovedfokuset i denne perioden har vært å implementere alle enhetene i applikasjonen, samt legge til funksjonalitet for lagring og avspilling av opptak. Dette er funksjonalitet som vi har greid å legge til i løpet av denne iterasjonen. Vi har også gjennomført en del tester. Resultatet av testingen viser at vi ligger bra ann til å greie alle A og B som ikke er «fritatt» fra arbeidsgiver. Vi har også funnet og lagt til funksjonalitet for «network-chaching» på videoavspilleren, som resulterer i 700 ms mindre delay. I tillegg har vi lagt til funksjonalitet for å vise «frames per seconds»

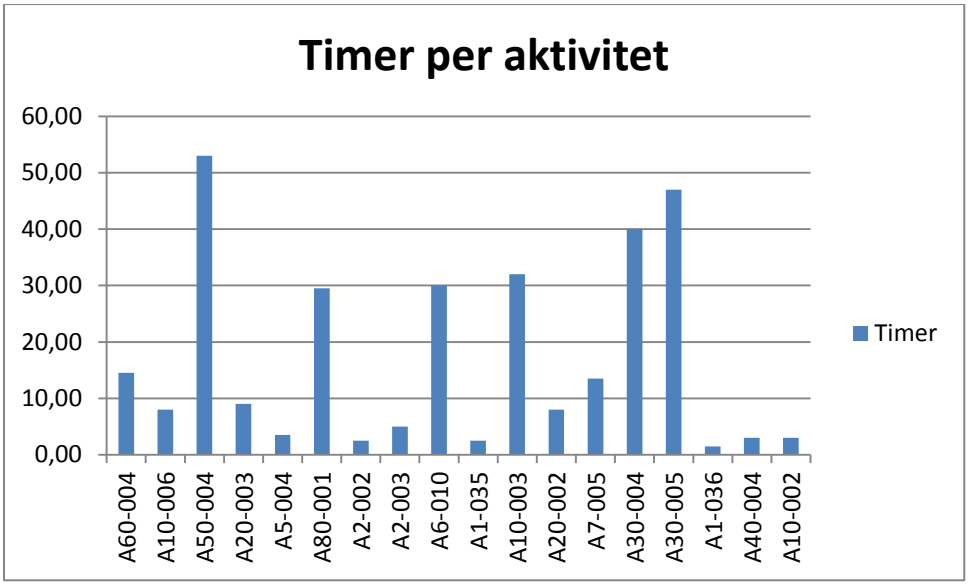
6 TIDSBRUK

6.1 PLANLAGT

Tabell 8 - Aktiviteter

Aktivitet ID	Beskrivelse av aktivitet	Tidsestimat	Ansvar
A2-002	Møte med intern veileder 06.04	4	Alle
A2-003	Møte med ekstern veileder 07.04	8	Alle
A7-005	Oppdatere websiden	0,5	Svein
A30-004	Funksjonalitet i GUI	24	CP
A6-010	Testing	32	Alle
A80-001	Lage Device API (RTO) - Factory - API for hver device(klasse)	24	Alex
A10-006	Lage CameraView	32	Svein Vegard
A10-006	Backend TCP	16	Svein Vegard
A60-004	Integrere MTCW	8	Vegard
A50-004	Integrere 3D Mus	8	CP
A30-004	Overlay som rører på seg	4	CP
A5-003	Skrive møtereferat	1	Svein
A10-003	Lagre video på server	50	Alle
A30-005	Spille av fra server	50	Alle
A50-004	Integrere 3D Mus	8	CP
A60-004	Integrere MTCW	8	Vegard
A40-004	Integrere Eye Tracker	8	Alex
A6-010	Testing	20	Alle
A1-036	Rapport	4	Vegard
A1-036	Plan	2	Svein

6.2 FAKTISK



Figur 2 - Tidsbruk

Tabell 9 - Godkjent estimering

#	Forutsatt	Faktisk timer	Estimert timer	Godkjent
A1-035		2,5		
A1-036		1,5	6	
A2-002		2,5	4	
A2-003		5	8	
A5-004		3,5		
A6-010		30	52	
A7-005		13,5	0,5	
A10-002		3		
A10-003		32	50	
A10-006		8	48	
A20-002		8		
A20-003		9		
A30-004		40	28	
A30-005		47	50	
A40-004		3		
A50-004		53	16	
A60-004		14,5	16	
A80-001		29,5	24	

Tabell 9 sin «Godkjent» kolonne er basert på:

$$\frac{Estimert}{2} \leq Godkjent \leq Estimert \ x \ 2$$

6.3 KONKLUSJON

Vi ser at vi har en forbedring i å planlegge aktiveteter. Det oppstår ende en del uforutsette aktiveteter. Dette er aktiveteter som er oppstått fordi vi har funnet andre måter å løse gamle problemer på. Da er det bedre å rette opp i dem fortløpende isteden for å vente til neste periode. Dette er fordi det vil ta lengre tid å sette seg inn i problemet igjen. Tids estimeringer av aktiviteter er gått opp i opp. Det er noen aktiviteter som vi har løst mye raskere enn det vi har antatt, mens andre aktiviteter har vi brukt en del mer tid på enn antatt.
Det vi kan ta med oss fra denne iterasjonene, er at vi har gjort så å si alle planlagte aktivitetene.

Høgskolen i Sørøst-Norge

KONGSBERG

REAL TIME OBSERVATION

GRUPPE 2

ITERASJONRAPPORT – C5

09.04.2016 – 13.05.2016

Oppdragsgiver

Kongsberg Defence & Aerospace AS

Utføres ved

Høgskolen i Sørøst-Norge, avd. Kongsberg

VEGARD SMEDSVIK
ALEXANDER SCHINNES

SVEIN BEKKEVOLD
CARL PEDER JANSEN

INNHOLDSFORTEGNELSE

1 DOKUMENTOMFANG 3
1.1 OVERSIKT 3
1.2 SYSTEMOVERSIKT 3
1.1 HISTORIE 3
2 FIGURER OG TABELLER 4
2.1 AKRONYMER OG FORKORTELSER 4
2.2 ORDBOK 4
2.3 TABELLER 4
2.4 FIGURER OG BILDER 4
3 STATUS 4
3.1 MÅ-MÅL 4
3.2 BØR-MÅL 6
4 AVGJØRELSER 7
4.1 AVGJØRELSE 7
5 KONKLUSJON 7
6 TIDSBRUK 7
6.1 PLANLAGT 7
6.2 FAKTISK 8
6.3 KONKLUSJON 9

1 DOKUMENTOMFANG

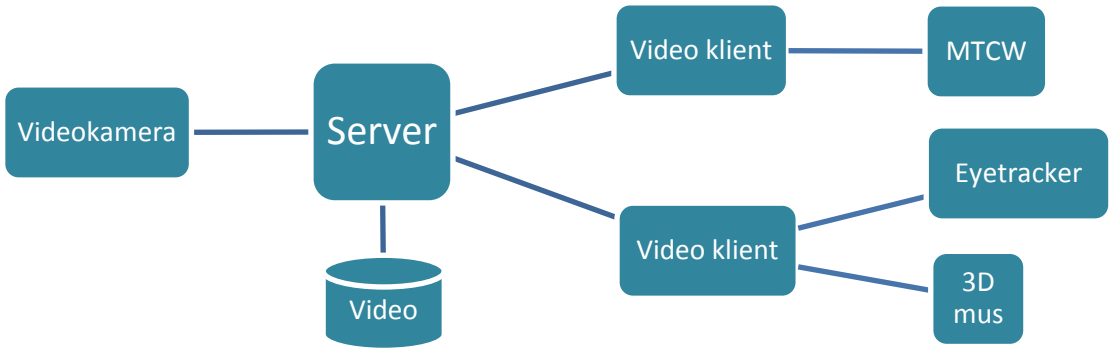
1.1 OVERSIKT

Dette er et dokument som inneholder en iterasjonsrapport for den femte iterasjonen i prosjektet. Iterasjon nummer en gikk fra 09.04 til 13.05. Dette dokumentet inneholder status, konklusjoner og avgjørelser som ble tatt i denne perioden.

1.2 SYSTEMOVERSIKT

Real Time Observation er et studentprosjekt som går ut på å se på nye og innovative løsninger for hvordan et motorisert videokamera skal kunne styres av en operatør med ulike MMI-enheter i en maritim applikasjon. Bilde og video fra videokameraet skal også kunne presenteres for brukeren i nåtid med ulike analyseresultater av videodata.

Systemet består av et kamera på som overfører videobildet via kabel til en videoserver. Denne videoserveren skal kunne distribuere videobildet i nåtid videre til ulike klienter hvor videoen skal vises. Videoen skal også kunne tas opp og lagres på serveren. Klientene skal kunne velge å hente opp videoen for å avspille den igjen på et senere tidspunkt. Det lages en egen applikasjon i Qt Quick for klientene. Her skal de kunne se videobildet, spille av tidligere opptak og styre kamera og applikasjon med ulike brukerenheter som «eyetracker», «Multi Touch Control Wheel» og 3D mus.



Figur 1 - Systemoversikt

1.3 HISTORIE

Tabell 1 - Dokumenthistorie

Versjon	Dato	Skrevet av	Beskrivelse
1.0	13.05.16	Carl Peder	

2 FIGURER OG TABELLER

2.1 AKRONYMER OG FORKORTELSER

Tabell 2 - Akronymer og forkortelser

Forkortelse	Beskrivelse
KDA	Kongsberg Defence & Aerospace AS
RTO	Real Time Observation
MMI	Man-Machine Interface
HSN	Høgskolen i Sørøst-Norge

2.2 ORDBOK

Tabell 3 - Ordbok

Ord	Beskrivelse
Oppdragsgiver	Kongsberg Defence & Aerospace AS (KDA) er en del av teknologibedriften Kongsberg Gruppen og fungerer som vår oppdragsgiver.

2.3 TABELLER

Tabell 1 - Dokumenthistorie 3
Tabell 2 - Akronymer og forkortelser 4
Tabell 3 - Ordbok 4
Tabell 4 - Administrative må-mål 5
Tabell 5 - Tekniske må-mål 5
Tabell 6 - Administrative bør-mål 6
Tabell 7 - Tekniske bør-mål 6
Tabell 8 - Aktiviteter 7
Tabell 9 - Godkjent estimering 8

2.4 FIGURER OG BILDER

Figur 1 - Systemoversikt 3
Figur 2 - Tidsbruk 8

3 STATUS

3.1 MÅ-MÅL

Tabell 4 - Administrative må-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A2-002	Møte med intern veileder 04.05 og 11.05	8	Alle
A2-003	Møte med ekstern veileder 04.05	8	Alle
A5-003	Skrive møtereferat	3	Svein
A7-005	Nettside	16	Svein
A1-034	Dokumentasjon - Analysedokument - Oppdater klassediagram - Kommunikasjon	24	Alle
A1-034	Nye Dokumenter - Brukermanual - Etteranalyse - Systembeskrivelse - Vedlikehold	40	Alle
A1-001	Oppdatere Gantt	4	Vegard
A1-036	Iterasjonsrapport	4	CP

Tabell 5 - Tekniske må-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A10-002	Optimalisering av video på server	32	Alex
A60-003	Optimaliser MTCW	8	Vegard
A80-002	Endre mode på kamera - API - GUI	8	Svein
A30-004	GUI	48	CP
A20-003	Kommunikasjon - Error - Feilmelding - Bruke kamera knapp - Avkobling - Antall klienter som er koblet til server	16	Vegard
A1-034	Dokumentasjon - Analysedokument - Oppdater klassediagram - Kommunikasjon - Systembeskrivelse	16	Alle
A80-002	Lage en Swipe funksjon	8	Svein
A6-010	Testing/Debugging	52	Alle
A7-008	Generell optimalisering	20	Alle

3.2 BØR-MÅL

Tabell 6 - Administrative bør-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar
A7-005	Oppdatere websiden	1	Svein
A5-003	Skrive møtereferat	3	Svein

Tabell 7 - Tekniske bør-mål

Aktivitet ID	Beskrivelse	Tidsestimat	Ansvar

4 AVGJØRELSER

4.1 AVGJØRELSE

Det er ikke blitt gjort noen spesielle avgjørelser som går mot hva som var planlagt i denne iterasjonen.

5 KONKLUSJON

Hovedfokuset i denne perioden har vært å sy hele systemet sammen til et komplett system. Vi er ferdig med en redesignet GUI som har all funksjonalitet som er planlagt. Det er utviklet swipe og nattmodus funksjoner. Det har vært spesielt fokus på testing og systemtesting av systemet. Systemtestene startet vi med i andre uke av perioden. I ettertid ser vi at dette er tester vi kunne startet med tidligere, da disse avdekket flere feil spesielt rettet mot kommunikasjon mellom server og klienter. Det ble totalt gjennomført tre fulle systemtester med før vi kunne anse testen som godkjent. Vi har også jobbet mye med dokumentasjon ved siden av det tekniske. Vi hadde planlagt mer tid til dette men da testing og bugfix tok lenger tid enn antatt ble det derfor mindre fokus på dokumentasjonen. Vi er fornøyd med hvordan iterasjonen har gått og mener vi ligger greit an i forhold til tidsplan.

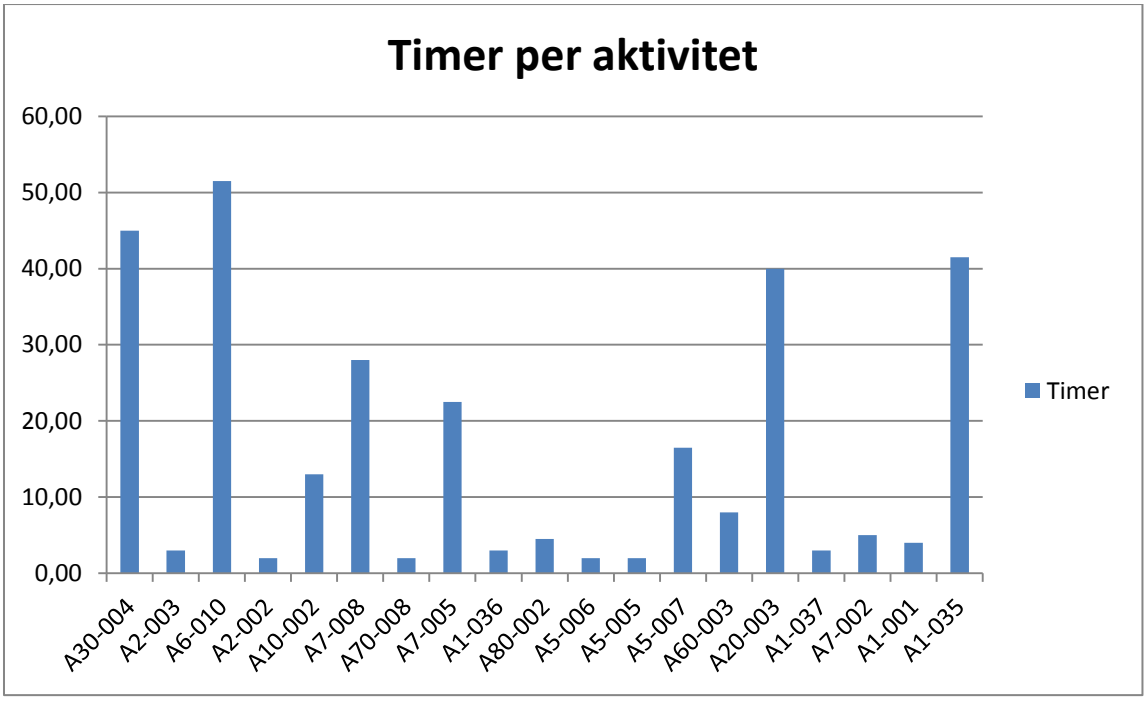
6 TIDSBRUK

6.1 PLANLAGT

Tabell 8 - Aktiviteter

Aktivitet ID	Beskrivelse av aktivitet	Tidsestimat	Ansvar
A10-002	Optimalisering av video på server	32	Alex
A60-003	Optimaliser MTCW	8	Vegard
A80-002	Endre mode på kamera	8	Svein
A30-004	GUI	48	CP
A20-003	Kommunikasjon	16	Vegard
A1-034	Dokumentasjon + Nye dokumenter	80	Alle
A80-002	Lage en Swipe funksjon	8	Svein
A7-005	Nettside	16	Svein
A6-010	Testing/Debugging	52	Alle
A2-002	Møte Intern	4	Alle
A2-003	Møte ekstern	8	Alle
A1-036	Iterasjonsrapport	4	CP
A7-008	Generell optimalisering	20	Alle
A10-002	Optimalisering av video på server	32	Alex

6.2 FAKTISK



Figur 2 – Tidsbruk

GRAF OVER TIMER

Tabell 9 - Godkjent estimering

#	Forutsatt	Faktisk timer	Estimert timer	Godkjent
A10-002		13	32	
A60-003		8	8	
A80-002		4,5	16	
A30-004		45	48	
A20-003		40	16	
A1-035		41,5	80	
A7-005		22,5	16	
A6-010		51,5	52	
A2-002		2	4	
A2-003		3	8	
A1-036		3	4	
A7-008		28	20	
A10-002		13	32	
A7-008		2	20	
A5-005		2		
A5-006		2		
A5-007		16,5		
A1-037		3		

Tabell 9 sin «Godkjent» kolonne er basert på:

$$\frac{Estimert}{2} \leq Godkjent \leq Estimert \ x \ 2$$

6.3 KONKLUSJON

Vi ser fortsatt at vi har et stort forbedringspotensial når det kommer til planlegging av aktiviteter og tidsestimat av disse. Det vi bommet på denne gangen var tiden brukt på testing og fiksing av bugs. Timene som ble brukt på debug ble skrevet på de aktuelle aktivitetene feilene var knyttet til. Tiden brukt på test og debug er altså fordelt på litt forskjellige aktiviteter, vi burde nok hatt en egen aktivitet for debug. Som nevnt tidligere ble det brukt mye tid på testing/debug, det har derfor blitt mindre tid til dokumentasjon. Utvikling av plakat til stand var dessverre en aktivitet vi glemte helt bort. Selv om det er mye som kan forbedres syntes vi at vi ligger godt an i forhold til tidsplan.