

BACHELOROPPGAVE:

**AKTIVE**

FORFATTER(E):

Lap To  
Tobias Moe Thorstensen  
Håvard Andreas Heggheim

Dato: 22/5-12



## SAMMENDRAG AV BACHELOROPPGAVEN

Tittel:	AKTIVe Tablet-basert kommunikasjonsløsning for hjemmetjenesten	Nr. : Dato : 22/5-12
Deltaker(e):	Lap To Tobias Moe Thorstensen Håvard Andreas Heggheim	
Veileder(e):	Frode Haug	
Oppdragsgiver:	Gjøvik kommune, Helse og Omsorg. Høgskolen i Gjøvik	
Kontaktperson:	Tommy Karlstad(GK), Tom Røise(HiG)	
Stikkord (4 stk)	Helse IKT, Android, Nettbrett, Flex	
Antall sider: 120 inkl. vedlegg	Antall bilag: 9	Tilgjengelighet: Åpen
<p>AKTIVe er et system for å forbedre kommunikasjonen mellom eldre og hjemmetjenesten i Gjøvik kommune. AKTIVe består av to systemer, et webpanel for administrering av brukere og en applikasjon for nettbrett og pc.</p> <p>AKTIVe applikasjonen er hovedbiten i dette systemet. I denne applikasjonen kan brukerne opprette og motta videosamtaler og få en oppdatert avtalekalender over ulike aktiviteter som skjer i fremtiden. Fra webpanelet kan administratorene av systemet opprette og endre brukere av systemet, oppdatere kontaktlisten og legge til aktiviteter i kalenderen til brukerne.</p> <p>Webpanelet er skrevet i HTML, CSS, PHP og JavaScript. jQuery rammeverket er også brukt i denne modulen. Nettsiden er optimalisert for nettleserne Firefox, Chrome, Opera og Internet Explorer 7.</p> <p>AKTIVe applikasjonen er utviklet med rammeverket Flex som er en kombinasjon med ActionScript 3.0 og MXML. PHP er også brukt for kommunikasjon med MySQL databasen.</p> <p>Kommunen vil tilby denne tjenesten for brukere med nettbrett kjørende Android. Brukere med pc vil også kunne kjøre denne tjenesten da applikasjonen er en kryssplattform.</p>		



## Forord:

Oppgaven ble presentert som en bacheloroppgave i 2011, da det innledende arbeidet ble gjort som en prosjektoppgave i faget Objekt orientert systemutvikling. Ingen valgte denne oppgaven i fjor, men gruppen fant denne oppgaven spennende og utfordrende. Forarbeidet fra faget objekt orientert systemutvikling har vært til stor hjelp da gruppen skulle presisere kravspesifikasjonen. Mye av funksjonaliteten som ble presentert i denne oppgaven ble ikke realisert, men de mest essensielle modulene har blitt implementert, etter interessentens ønsker.

Det innledende arbeidet med valg av utviklingspråk og teknologi var tidkrevende da applikasjonen måtte ha støtte for video- og samtalekommunikasjon. Gruppen har fra tidligere hobbyprosjekter jobbet med Android plattformen, men ikke med Flex rammeverket. Arbeidet med å lære seg dette rammeverket var til tider problematisk, men i løpet av kort tid fikk gruppen større forståelse og oppståtte problemer ble løst raskt.

Prosjektet har både vært lærerikt og spennende da programvaren som vi har utviklet vil bli benyttet til realistiske formål av en brukergruppe som vil dra nytte av denne teknologien.

Gruppen ønsker å rette en stor takk til oppdragsgiverne, Tommy Karlstad og Tom Røise for godt samarbeid og god kommunikasjon under prosjektet. Vi ønsker også å takke vår veileder, Frode Haug for gode konstruktive tilbakemeldinger under veiledningstimene ved Høgskolen i Gjøvik.

Videre ønsker vi å rette en takk til personene som har testet programvaren underveis i utviklingsfasen, dette har gitt oss en god pekepinn på funksjonalitet som kunne forbedres.

Gjøvik, 22. mai 2012

Lap To

Lap To

Håvard Heggheim

Håvard Heggheim

Tobias M. Thorstensen

Tobias Moe Thorstensen



# Innhold

<b>1 Innledning</b>	<b>1</b>
1.1 Problemområde og oppgavedefinisjon . . . . .	1
1.2 Målgruppe . . . . .	2
1.3 Gruppens bakgrunn og kompetanse . . . . .	2
1.4 Arbeidsmetodikk og gjennomføring . . . . .	2
1.4.1 Utviklingsmodell . . . . .	2
1.4.2 Roller og prosjektorganisering . . . . .	4
1.5 Organisering av rapporten . . . . .	5
1.5.1 Kapitteloppsummering . . . . .	5
1.5.2 Terminologi . . . . .	6
1.5.3 Layout . . . . .	6
<b>2 Kravspesifikasjon</b>	<b>7</b>
2.1 Introduksjon . . . . .	7
2.1.1 Krav til systemet . . . . .	7
2.1.2 Systemets omgivelser . . . . .	7
2.1.3 Systemets brukere . . . . .	7
2.2 Overordnede funksjonelle krav . . . . .	8
2.2.1 Use Case Diagram . . . . .	8
2.2.2 Overordnet Use Case beskrivelse . . . . .	9
2.2.3 Detaljert Use Case beskrivelse . . . . .	12
2.3 Klassediagram . . . . .	22
2.3.1 Forklaring til klassediagrammet . . . . .	22
2.4 Overordnede operasjonelle systemkrav . . . . .	23
2.4.1 Ytelse . . . . .	23
2.4.2 Sikkerhet . . . . .	24
2.4.3 Tilgjengelighet . . . . .	24
2.4.4 Utgivelser . . . . .	24
2.5 Begrensninger . . . . .	25
2.5.1 Software design begrensninger . . . . .	25
2.5.2 Hardware begrensninger . . . . .	25
2.6 Aspekter omkring livssyklus . . . . .	25
2.6.1 Dokumentasjon . . . . .	25

2.6.2	Krav til utvidelser . . . . .	25
<b>3</b>	<b>Programvarearkitektur og design</b>	<b>26</b>
3.1	AKTIVe applikasjonen . . . . .	26
3.2	Webpanel . . . . .	27
3.3	Deployment view . . . . .	28
3.4	Databasemodell . . . . .	29
3.5	Brukergrensesnitt i AKTIVe applikasjonen . . . . .	30
3.6	Faktisk brukergrensesnitt . . . . .	31
3.7	Designutfordringer . . . . .	32
3.7.1	Lagring av kontakter . . . . .	32
3.7.2	Ekko problematikk . . . . .	33
<b>4</b>	<b>Implementering</b>	<b>34</b>
4.1	Valg av utviklingsmiljø og språk . . . . .	34
4.2	Valg av tredjeparts tjenester . . . . .	34
4.3	Implementasjon av webpanel . . . . .	35
4.3.1	Registrer ny bruker . . . . .	35
4.3.2	Endre bruker . . . . .	36
4.3.3	Endre venner . . . . .	38
4.3.4	Avtalekalender . . . . .	40
4.3.5	Kvitterte avtaler . . . . .	42
4.4	Implementasjon av nettbrett applikasjon . . . . .	43
4.4.1	Innlogging . . . . .	43
4.4.2	Video- og samtalekommunikasjon . . . . .	47
4.4.3	Avtaler . . . . .	51
4.4.4	Rangering av brukere . . . . .	52
<b>5</b>	<b>Testing og kvalitetssikring</b>	<b>54</b>
5.1	Testgruppe . . . . .	54
5.2	Whitebox testing . . . . .	54
5.3	Blackbox testing . . . . .	54
5.4	Regresjonstesting . . . . .	54
5.5	Enheter brukt under testing . . . . .	56
5.5.1	HTC Flyer . . . . .	56
5.5.2	Samsung Galaxy Tab 1.0 . . . . .	57



<b>6</b>	<b>Evaluering og avslutning</b>	<b>58</b>
6.1	Resultater . . . . .	58
6.2	Videre arbeid . . . . .	58
6.3	Drøfting av gruppens arbeid . . . . .	59
6.4	Konklusjon . . . . .	60
<b>7</b>	<b>Vedlegg</b>	<b>62</b>
7.1	A - Ordliste . . . . .	63
7.2	B - Veiledningsdokumenter gitt til sluttbrukere . . . . .	66
7.3	C - Statusrapporter og møtelogger . . . . .	85
7.4	D - Faktisk gannt-skjema . . . . .	99
7.5	E - Forprosjekt . . . . .	101
7.6	F - Gruppereregler . . . . .	111
7.7	G - Prosjektavtale . . . . .	113
7.8	H - Dokumentasjon av kildekode . . . . .	116
7.9	I - Logg og timeføring . . . . .	118

## Figurer

1	Arbeidsmetodikken scrum . . . . .	3
2	Use Case Diagram . . . . .	8
3	Sekvensdiagram for 'Logg inn' . . . . .	18
4	Sekvensdiagram diagram for 'Opprette videosamtale' . . . . .	21
5	Klassediagram over systemet . . . . .	22
6	Arkitektur - AKTIVe applikasjonen . . . . .	26
7	Arkitektur - webpanel . . . . .	27
8	Deployment view av systemet . . . . .	28
9	Databasemodell over systemet . . . . .	29
10	Forslag til brukergrensesnitt . . . . .	30
11	Faktisk brukergrensesnitt i AKTIVe applikasjonen . . . . .	31
12	AEC i praksis . . . . .	33
13	Oppretting av brukere i webpanelet . . . . .	35
14	Endring av brukere i webpanelet - Nedtrekksmeny . . . . .	36
15	Endring av brukere i webpanelet . . . . .	37
16	Legge til venner i webpanelet - Nedtrekksmeny . . . . .	38
17	Legge til venner i webpanelet . . . . .	39
18	Oppretting av ny avtale . . . . .	40
19	Opprette ny avtale . . . . .	41
20	Kvitterte avtaler . . . . .	42
21	Nettbrettet HTC Flyer . . . . .	56
22	Nettbrettet Samsung Galaxy Tab . . . . .	57

## Listings

1	Logg inn fra applikasjonen . . . . .	43
2	HTTPService for innlogging . . . . .	43
3	Logg inn skriptet . . . . .	44
4	Logg inn skriptet . . . . .	45
5	Kodesnutt som lagrer dataene . . . . .	45
6	Kodesnutt for automatisk innlogging . . . . .	46
7	PHP skript for å opprette bruker i cirrus . . . . .	47
8	Henter mikrofon og kamera fra enheten . . . . .	48
9	Opprette videosamtale . . . . .	48
10	Søk basert på fornavn og etternavn. . . . .	48
11	Opprette videosamtale . . . . .	49
12	Funksjon for å lage en ringtone . . . . .	49
13	Funksjon for å godta en samtale . . . . .	50
14	Kodesnutt for å generere dato . . . . .	51
15	Kodesnutt for å formatere dato . . . . .	51
16	Kodesnutt som lytter til avkrysningsboksen . . . . .	51
17	Kode for å hente kontakter fra database . . . . .	52
18	Kode for å legge brukere i et array . . . . .	52
19	Kode for å rangere brukere . . . . .	53

# 1 Innledning

## 1.1 Problemområde og oppgavedefinisjon

Høgskolen har i samarbeid med Gjøvik kommune tidligere gjennomført forskningsprosjektet AKTIVe (Aktiverings- og KommunikasjonsTeknologi Innen Velferdstjenester for Eldre) med støtte fra Regionalt Forskningsfond Innlandet. Dette prosjektet skal benyttes som en veiledning for spesifikasjoner og krav underveis i bacheloroppgaven, men det vil være muligheter for å gjøre endringer i samtale med oppdragsgiver. I løpet av prosjektets levetid skal spesifiseringen av kravene fullføres, samt utviklingen av applikasjonen som forbedrer kommunikasjon mellom tjenestemottakerne og kommunens omsorgstjeneste.

Applikasjonen skal være egnet for eldre. Helsen blir svekket med årene og mange finner det vanskelig å komme seg ut av egen bolig for å sosialisere seg med andre. Dette medfører et stadig behov for besøk av omsorgstjenesten og familie. Ønsket er å tilby en teknologisk enkel og sosial løsning som letter arbeidet for hjemmetjenesten, og samtidig sprer glede hos eldre, der de får større muligheter til sosialisere seg i omgangskretsen sin. Implementert funksjonalitet er video- og samtalekommunikasjon og en påminnelsetavle inneholdene kommende avtaler eller gjøremål. Ved å lette kommunikasjonen mellom helsepersonell og tjenestemottaker tilknyttet hjemmetjenesten vil de ansatte kunne effektivisere sin jobbhverdag. Tjenestemottakerne vil også få større mulighet for å opprettholde daglig kontakt med denne tjenesten som kommunen tilbyr. Applikasjonen skal underveis testes blant ansatte og beboere i Gjøvik kommune og videreutvikles basert på tilbakemeldiger fra sluttbrukerne. Mye av utfordringen i prosjektet ligger ikke bare i utviklingen av kjernefunksjonaliteten, men å kombinere dette med et enkelt og forståelige brukergrensesnitt.

## 1.2 Målgruppe

Denne rapportens målgruppe er sensor, medstudenter, veileder, oppdragsgiver og andre interesserte ved utdanningsinstitusjonen, Høgskolen i Gjøvik. Det er blitt opprettet en testbruker slik at sensor har mulighet for å logge seg inn, og teste systemet i sin helhet. Brukerinformasjonen ligger i filen `readme.txt` som er lokalisert under mappen `AKTIVeTotalLøsning.zip`. Hovedmålgruppen for prosjektet er eldre hjemmeboende som har tilknytning til omsorgstjenesten i Gjøvik kommune. Målgruppen vil også være personer med tilknytning til den eldre, dette er personer som hjelpepleier, familie og venner.

## 1.3 Gruppens bakgrunn og kompetanse

Alle gruppe-medlemmene studerer dataingeniør, vår faglige bakgrunn ligger i utvikling av programvare, systemutvikling, realfag og noe elektronikk. Gjennom tre år på Høgskolen i Gjøvik har gruppe-medlemmene hovedsaklig erfaring fra programmeringsspråkene C++ og Java. Gruppen har også kompetanse på HTML, CSS, PHP og JavaScript. Utenom skolen har gruppe-medlemmene programmert aktivt i Java med Android SDK. Utviklingsrammeverket Flex som ble benyttet har ingen tidligere erfaringer med, derfor måtte dette læres fra bunnen av.

## 1.4 Arbeidsmetodikk og gjennomføring

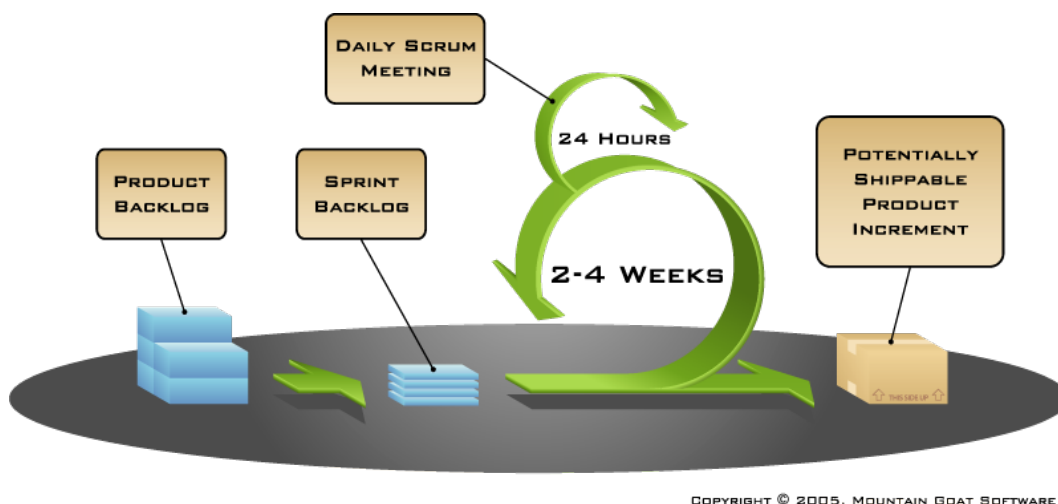
Etter hvert gjennomførte møte med oppdragsgiver ble det skrevet et referat, hvert gruppe-medlem har også skrevet daglige logger over arbeidet som ble lagt ned, samt en oversikt over timeantallet den dagen. Møte med oppdragsgiver ble holdt to ganger i måneden, disse møtene fant sted før hver sprint. Gruppens kjernetid er klokken 08:00-16:00, i dette tidsrommet er alle gruppe-medlemme tilstede på vårt avsatte grupperom, A032. For å holde en oppdatert logg over arbeidsoppgaver benyttet gruppen web-løsningen Trello, som tilbyr de samme funksjonalitetene som et *scrum board*. For å sikre at det alltid kodes på siste, oppdaterte versjon av programvaren ble versjonkontrolleringssystemet subclipse benyttet.

### 1.4.1 Utviklingsmodell

I starten av prosjektet hadde gruppen en diskusjon om hvilken utviklingsmodell som skulle følges under prosjektets levetid. Gruppen konkluderte med at fokuset ville bli rettet mot en smidig utviklingsmodell, da slike modeller legger til rette for at kravspesifikasjonen kunne bli endret underveis i prosjektet. De gjentakende prinsippene i slike modeller er iterasjoner, kontinuerlig forbedringer og hyppige utgivelser. Utviklingsmodellene som ble trukket frem var XP og Scrum, begge gode alternativer, med sine gode og dårlige sider.

eXtreme Programming (XP) har på like linje med Scrum hyppige utgivelser og iterasjoner, men forskjellig terminologi. eXtreme Programming skiller seg fra andre utviklingsmodeller ved å innføre prinsippet parprogrammering. Dette er en arbeidsmetodikk som legger til rette for at to personer skal kunne sitte sammen og utvikle kode. Dette er et prinsipp som fungerer meget bra, men da gruppen kun består av tre personer kan dette bli vanskelig å gjennomføre. Et annet aspekt ved parprogrammering er at tiden blir lite effektivt benyttet og med tanke på arbeidsmengden som gruppen skal utføre, viser dette seg å være et problem.

Som eneste gjenstående alternativ, falt valget på Scrum. Denne utviklingsmodellen er som kjent en smidig utviklingsmodell som legger til rette for endring i kravene fra interessenten. I startfasen av prosjektet kom oppdragsgiver med krav til programvaren, dette kalt Product Backlog. Grunnpilaret i Scrum er sprinter. I forkant av en sprint ble det holdt et møte med oppdragsgiver, kalt sprint planning meeting. Dette møtet førte til en Sprint Backlog som beskrev funksjonaliteten som skulle implementeres i applikasjonen i løpet av sprinten. Valgt funksjonalitet skal være i tråd med beskrevet funksjonalitet i product backlog. Etter møtet satt gruppen seg ned i 14 dager, skjermet fra interessenten og implementerte ønsket funksjonalitet som fremkom på møtet. Den eneste aktøren i gruppen som kunne kommunisere med oppdragsgiver under en sprint, var Scrum Master. Vedkommende som hadde denne rollen var prosjektlederen i gruppen. I slutfasen av sprinten holdt gruppen et sprint retrospectiv møte hvor sprinten ble vurdert innad i gruppen. Før neste sprint startet ble interessenten presentert for arbeidet som var blitt utført, et slikt møte kalles sprint review meeting.



Figur 1: Arbeidsmetodikken Scrum<sup>1</sup>

<sup>1</sup>Figur hentet fra: <http://www.mountaingoatsoftware.com/scrum/overview>

#### 1.4.2 Roller og prosjektorganisering

<b>Oppdragsgiver:</b>	Tommy Karlstad, Gjøvik kommune. Tom Røise, Høgskolen i Gjøvik.
<b>Veileder:</b>	Frode Haug, Høgskolen i Gjøvik.
<b>Prosjektleder:</b>	Håvard Andreas Heggheim
<b>Hjemmeside:</b>	Tobias Moe Thorstensen
<b>Rapportansvarlig:</b>	Tobias Moe Thorstensen
<b>Flex utvikler:</b>	Lap To
<b>Webpanel backend:</b>	Håvard Andreas Heggheim
<b>Webpanel frontend:</b>	Tobias Moe Thorstensen

## **1.5 Organisering av rapporten**

### **1.5.1 Kapitteloppsummering**

#### **Innledning**

I denne seksjonen beskrives formålene med utviklingsarbeidet, hvilken målgruppe som essent ønsker å treffe, hvilke avgrensninger som er satt, samt en diskusjon på hvilken utviklingsmodell som gruppen fulgte.

#### **Kravspesifikasjon**

Kravspesifikasjonen er dokumentet som beskriver funksjonaliteten i sluttproduktet. Kravene til systemet og hvilke omgivelser system befinner seg i, blir også beskrevet i denne seksjonen.

#### **Programvarearkitektur og design**

I dette kapitlet beskrives systemets oppbygning, modulene blir beskrevet hver for seg med forklarende illustrasjoner og tekst. Systemet i sin helhet blir også presentert i denne seksjonen.

#### **Implementering**

I denne modulen blir utviklingsverktøyene, utviklingspråk og fremgangsmåten gruppen har brukt for å implementere sluttproduktet presentert.

#### **Testing og kvalitetssikring**

Testing er en essensiell bit av prosjektet, da vi utvikler programvare for en krevende brukergruppe. Denne seksjonen forklarer litt om testgruppen og ulike testmetodikker.

#### **Evaluering og avslutning**

Evaluering og avslutnings kapitlet inneholder drøfting og diskusjoner rundt gjennomføringen av oppgaven. Forslag til videre arbeid og en konklusjon av utført arbeid.

#### **Vedlegg**

Her blir møtelogger, fremmedord og brukerveiledninger m.m presentert.



### 1.5.2 Terminologi

I rapporten brukes det ulik terminologi på aktører og komponenter i systemet. Listen under definerer disse uttrykkene.

- Pleier, hjelpepleier og administrator komplementerer hverandre.
- Eldre og tjenestemottaker komplementerer hverandre.
- Påminnelsestavle og avtalekalender komplementerer hverandre.
- Gjøre mål og avtale komplementerer hverandre.

### 1.5.3 Layout

Rapporten er utformet med utgangspunkt i rapportmalen gitt ved Høgskolen i Gjøvik, med noen modifikasjoner. Rapporten er inndelt i kapitler, med tilhørende underkapitler. Det refereres med vancouver referansestil, dette vil si punktnummerering og referanseliste i slutten av rapporten. Alle bilder og opplistninger er presentert i egen innholdsfortegnelse i starten av rapporten. Alle fremmedord som opptrer i rapporten er forklart i vedlegg A, disse ordene er markert med kursiv i rapporten.

## 2 Kravspesifikasjon

Dette kapitlet gir et overblikk over programvaren og en nærmere beskrivelse av funksjonaliteten som tilbys sluttbrukerne av systemet. Kravspesifikasjonen er et resultat av møter med oppdragsgiver, hvor vedkommende har fremmet sitt syn på funksjonaliteten, samt gruppens egne vurderinger basert på tilgjengelig skriftlig informasjon gitt av interessenten.

### 2.1 Introduksjon

#### 2.1.1 Krav til systemet

Den utviklede løsningen er en kommunikasjonsplattform mellom eldre tjenestemottakere og helse- og omsorgstjenesten i Gjøvik kommune. Programvaren er en plattform som skal virke avlastende og øke den sosiale interaksjonen mellom brukerne. Siden AKTIVe applikasjonen skal brukes av en generasjon som ikke er vant med informasjonsteknologi, er det helt essensielt at brukergrensesnittet er lett og forståelig. For å opprette videosamtaler og motta oppdateringer i sin avtalekalender, må enhver bruker i systemet registreres i en sentral database. For å opprette og administrere bruker, opprette gjøremål og lese kvitterte gjøremål må en aksessere webpanelet. Dette panelet er administrasjonsverktøyet i systemet, for å få tilgang til webpanelet må en logge inn med en brukerkonto som har administratorrettigheter.

#### 2.1.2 Systemets omgivelser

Applikasjonen kjører hovedsakelig på håndholdte, mobile enheter med *Android* som operativsystem og er optimalisert for nettbrett. Programvaren kan også kjøres på pc som en vanlig applikasjon. Enheten må ha støtte for frontkamera og mikrofon, samt nettverkskort som muliggjør overføring av data mellom enheten og serveren. Serveren kjører skripts som håndterer autentisering av brukere, oppsett av video- og lydkommunikasjon og annen funksjonalitet som trenger tilgang til databasen. Webpanelet er optimalisert for nettleserne IE7, Google Chrome og Firefox.

#### 2.1.3 Systemets brukere

- Administrator: Hjelpepleiere med tilgang til webpanelet.
- Standard brukere: Tjenestemottakere som har tilgang til AKTIVe applikasjonen.

## 2.2 Overordnede funksjonelle krav

### 2.2.1 Use Case Diagram



Figur 2: Use case diagram over systemet

## 2.2.2 Overordnet Use Case beskrivelse

Use Case	Logg inn på AKTIVe applikasjonen
Aktør	Tjenestemottaker og pleier
Hensikt	Aktør ønsker tilgang til AKTIVe applikasjonen.
Beskrivelse	Aktørene logger inn på applikasjonen ved hjelp av brukernavn og passord. Om autentiseringen ble godkjent vil vedkommende bli videre-sendt til hjemskjermen.

Use Case	Vis avtaler
Aktør	Tjenestemottaker
Hensikt	Vedkommende ønsker oversikt over kommende avtaler, arrangementer og daglige gjøremål.
Beskrivelse	Forutsett at brukeren er logget inn for å utføre denne handlingen. Hjemskjermen vil ha et felt hvor alle avtalene blir vist. Beskjedene i dette feltet vil inneholdet en dato når gjøremålet skal utføres og en beskrivende tekst.

Use Case	Administrer kontakter
Aktør	Pleier
Hensikt	Aktøren ønsker å legge til, endre eller slette venner fra kontaktlisten til en bruker.
Beskrivelse	Aktøren logger inn på webpanelet og velger 'endre venner' fra menyen. Herfra kan aktøren velge hvilke venner en bruker skal ha i sin kontaktliste

Use Case	Logg inn på webpanelet
Aktør	Pleier
Hensikt	Aktøren ønsker å logge inn på webpanelet for å administrere brukere
Beskrivelse	Aktøren taster brukernavn og passord for å logge inn på webpanelet, herfra kan aktøren administrere alle brukere av systemet. Aktøren kan registrere nye brukere, endre brukere, oppdatere kalender, vise kvitterte avtaler og legge til nye venner for brukere av systemet.

Use Case	Ajourføre kalender
Aktør	Pleier
Hensikt	Oppdatere kalenderen til tjenestemottaker
Beskrivelse	Pleier legger til avtaler og arrangementer til tjenestemottakerens kalender fra webpanelet.

Use Case	Motta videosamtale
Aktør	Pleier og tjenestemottaker
Hensikt	Opprette en videosamtale
Beskrivelse	Vedkommende som mottar en videosamtale vil høre en ringelyd fra enheten og skjermen vil vise hvem som ringer. Vedkommende har da mulighet for å godta eller avslå denne forespørselen.

Use Case	Opprette videosamtale
Aktør	Pleier og tjenestemottaker
Hensikt	Starte en videosamtale
Beskrivelse	Aktørene har mulighet for å opprette en videosamtale. Dette gjøres ved å trykke på profilbildet til en kontakt og bekrefte at samtalen skal opprettes.

Use Case	Kvitter gjøremål
Aktør	Tjenestemottaker
Hensikt	Kvittere for utført gjøremål.
Beskrivelse	Aktøren kvitterer for utført gjøremål som er oppført i avtalekalenderen.

Use Case	Administrer bruker
Aktør	Pleier
Hensikt	Opprette nye brukere, endre eller slette dem fra systemet
Beskrivelse	Aktøren har mulighet for å opprette nye, endre eller slette brukere

Use Case	Vis kvitterte gjøremål
Aktør	Pleier
Hensikt	Presentere utførte gjøremål
Beskrivelse	Aktøren ønsker å vite om tjenestemottakeren har kvittert gjøremålet.

Use Case	Opprett bruker
Aktør	Pleier
Hensikt	Aktør ønsker å opprette nye brukere.
Beskrivelse	Aktøren logger inn på webpanelet og velger 'Registrer ny bruker'. Fyller deretter inn påkrev informasjon.

### 2.2.3 Detaljert Use Case beskrivelse

I denne underseksjonen vil use case modellen bli beskrevet detaljert. Noen av casene er modellert med UML-diagrammet sekvensdiagram, dette er de mest komplekse og essensielle modulene i systemet.

Use Case	Motta videosamtale
Aktør	Pleier og tjenestemottaker
Mål	Kommunikasjon med video og lyd.
Beskrivelse	Aktøren ønsker å motta en innkommende videosamtale.
Type	Viktig
Pre-betingelser	Bruker er logget inn på tjenesten AKTIVe, peer ID er generert og noen prøver å kontakte vedkommende
Post-betingelser	
Spesielle krav	Ingen
Detaljert hendelsesforløp	
<b>Aktør</b>	<b>System</b>
2. Mottar godtar eller avslår forespørselen	1. Signaliserer at aktøren har en innkommende samtale 3. Samtale forbundet
Alternativ hendelsesforløp nr. 1	
	2.Mottaker avslår samtalen. 3.Vedkommende som prøvde å forbinde en samtale får beskjed om avslaget.

Use Case	Administrer brukere
Aktør	Pleier
Mål	Endre brukerinformasjon til en bruker
Beskrivelse	Aktøren ønsker å endre informasjonen til en bruker i databasen. Informasjon som kan endres er fornavn, etternavn, passord og profilbilde
Pre-betingelser	Aktøren er logget inn på webpanelet
Post-betingelser	Bruker finnes i databasen
Spesielle krav	Ingen
Detaljert hendelsesforløp	
<p><b>Aktør</b></p> <p>1. Velger fanen 'endre brukere' i webpanelet</p> <p>3. Aktøren velger bruker fra nedtrekkslisten.</p> <p>5. Aktøren endrer informasjonen.</p>	<p style="text-align: center;"><b>System</b></p> <p>2. Webpanelet viser en liste over registrerte brukere i databasen.</p> <p>4. Nåværende info om bruker blir vist.</p> <p>6. Gir beskjed om at info er endret.</p>
Alternativ hendelsesforløp nr. 1	
	6. Obligatoriske felter er ikke fylt inn riktig.



Use Case	Opprett bruker
Aktør	Pleier
Mål	Opprette en bruker
Beskrivelse	Aktøren ønsker å opprette en ny bruker av systemet
Pre-betingelser	Aktøren er logget inn på webpanelet
Post-betingelser	Ingen
Spesielle krav	Ingen
Detaljert hendelsesforløp	
<b>Aktør</b> 1. Velger fanen 'Registrer ny bruker' i webpanelet  3. Aktøren fyller inn informasjonen	<b>System</b> 2. Webpanelet presenterer informasjon som må fylles ut  4. Får beskjed om at bruker er registrert.
Alternativ hendelsesforløp nr. 1	
	4. Obligatoriske felter er ikke fylt inn korrekt.

Use Case	Vis Avtaler
Aktør	Tjenestemottaker
Mål	Vise registrerte avtaler
Beskrivelse	Aktøren ønsker å vite hvilke avtaler som er oppført.
Pre-betingelser	Bruker er logget inn på AKTIVe portalen og det eksisterer avtaler.
Post-betingelser	Ingen
Spesielle krav	Ingen

Detaljert hendelsesforløp

Aktør	System
1. Navigerer seg til hjemskjermen	2. Presenterer avtalene rangert etter satt utførelses dato.

Use Case	Ajourfør avtalekalender
Aktør	Pleier
Mål	Vedlikeholde kalender
Beskrivelse	Aktøren ønsker å oppdatere den tjenestemottakerens avtalekalender.
Pre-betingelser	Bruker er logget inn på AKTIVe webpanel
Post-betingelser	Bekreft innsending av data
Spesielle krav	Ingen

Detaljert hendelsesforløp

Aktør	System
1. Velger fanen 'Ny avtale' i webpanelet	2. Webpanelet viser en liste over registrerte brukere i databasen.
3. Aktøren velger bruker fra nedtrekkslisten.	4. Tekstfelt og kalender blir vist.
5. Velger dato, skriver beskjed, velger om kvittering trengs, avtalens viktighet og om avtalen skal repeteres, trykker 'Post avtale'.	6. Gir beskjed om at kalenderen er oppdatert.

Alternativ hendelsesforløp nr. 1

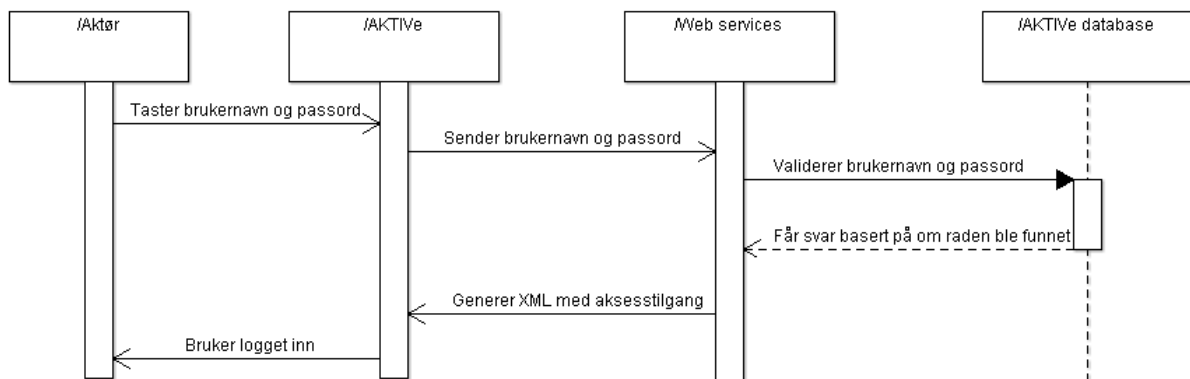
	6. Obligatoriske felter er ikke fylt inn, eller avtalen inneholder ulovlige tegn.
--	---

Use Case	Logg inn på webpanelet	
Aktør	Pleier	
Mål	Administrere systemet	
Beskrivelse	Aktøren ønsker tilgang til webpanelet hvor en kan administrere brukere av systemet, oppdatere kalender og legge til venner for brukere av panelet.	
Pre-betingelser	Aktøren har tilgang til internett og har en bruker med administrator rettigheter	
Post-betingelser	Ingen	
Spesielle krav	Ingen	
Detaljert hendelsesforløp		
<b>Aktør</b>	<b>System</b>	
1. Navigerer seg til nettsiden	2. Webpanelet viser innloggingsskjermen.	
3. Skriver inn brukernavn og passord.	4. Validerer brukernavn og passord mot databasen	
	5. Presenterer fremsiden av webpanelet.	
Alternativ hendelsesforløp nr. 1		
	4. Brukernavn og passord er ikke korrekt.	
	5. Bruker får beskjed om at brukernavn og passord ikke er korrekt	

Use Case	Administrer kontakter
Aktør	Pleier
Mål	Endre, legge til eller slette kontakter fra kontaktlisten
Beskrivelse	Aktøren ønsker å editere kontaktlisten slik at det er mulig å ringe andre brukere av systemet.
Type	Viktig
Pre-betingelser	Bruker er logget inn på webpanelet
Post-betingelser	
Spesielle krav	Ingen
Detaljert hendelsesforløp	
<b>Aktør</b> 1. Trykker på fanen 'endre venner'  3. Velger bruker fra listen  5. Aktør velger personene som skal legges til som venner	<b>System</b>  2. Viser nedtrekklisen over alle brukere  4. Presenterer alle bruker i systemet som kan legges til som venn for denne bruker  6. Kontaktene blir lagt til i kontaktlisten

Use Case	Kvitter gjøremål
Aktør	Tjenestemottaker
Mål	Kvittere for utførte gjøremål
Beskrivelse	Aktøren ønsker å kvittere for utførte gjøremål, slik at pleier får vite dette.
Type	Middels
Pre-betingelser	Det eksisterer gjøremål som krever kvittering etter utførelse
Post-betingelser	
Spesielle krav	Ingen
Detaljert hendelsesforløp	
<b>Aktør</b> 1. Navigerer seg til hjemskjermen  3. Huker av for utført gjøremål i avtalekalenderen	<b>System</b>  2. Viser avtalene for denne brukeren

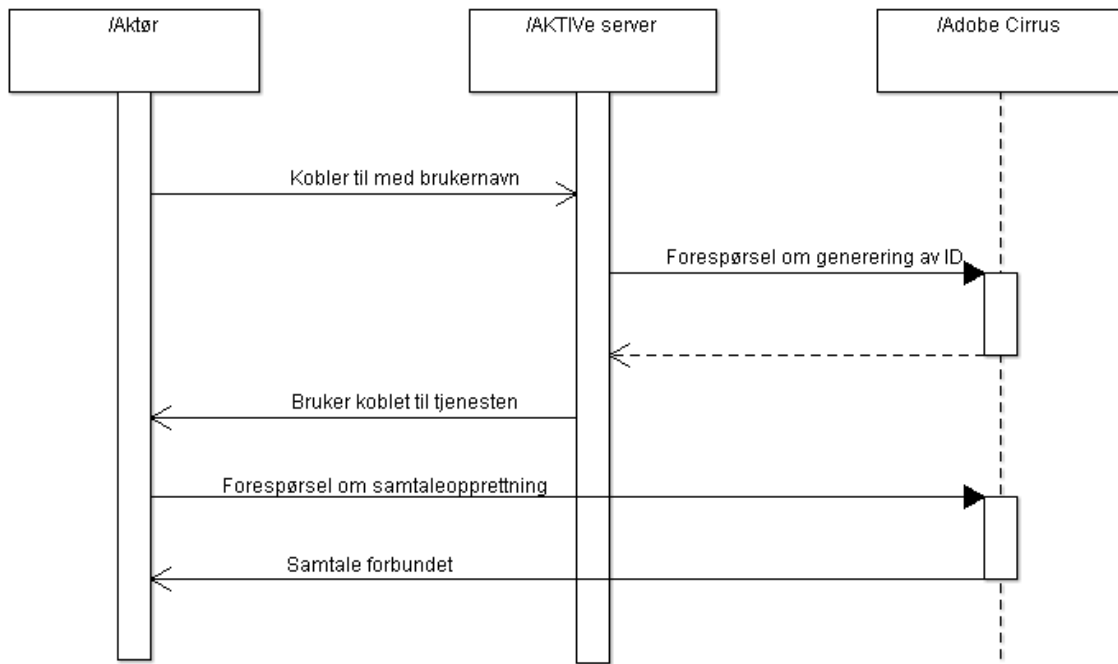
Use Case	Logg inn på AKTIVe applikasjonen
Aktør	Pleier og tjenestemottaker
Mål	Logge inn på plattformen
Beskrivelse	Brukeren logger inn på AKTIVe applikasjonen for å kunne benytte seg av tilgjengelige tjenester som lyd- og bildekommunikasjon, samt avtalekalender.
Type	Viktig
Pre-betingelser	Bruker har tilgang til nett, for å validere brukernavn og passord
Post-betingelser	Ingen
Spesielle krav	Ingen
<b>Detaljert hendelsesforløp</b>	
<b>Aktør</b> 1. Taster inn brukernavn og passord  3. Bruker logget inn	<b>System</b> 2. Korrekt brukernavn og passord  4. Presenterer hovedmenyen.
<b>Alternativ hendelsesforløp</b>	
	2. Feil brukernavn og passord  3. Viser bruker at det er feil kombinasjon mellom brukernavn og passord.



Figur 3: Sekvensdiagram for 'Logg inn'

Use Case	Vis kvitterte gjøremål
Aktør	Pleier
Mål	Vis kvitterte gjøremål fra tjenestemottakerne
Beskrivelse	Verifisere at tjenestemottakerne har utført hendelser som krever kvittering.
Type	Middels
Pre-betingelser	Bruker har tilgang til webpanelet og det eksisterer gjøremål som har blitt kvittert
Post-betingelser	Ingen
Spesielle krav	Ingen
Detaljert hendelsesforløp	
<b>Aktør</b> 1. Taster inn brukernavn og passord  3. Bruker velger lenken 'Kvitterte avtaler'	<b>System</b>  2. Presenterer hovedsiden  4. Presenterer alle avtaler som har blitt kvittert.
Alternativ hendelsesforløp nr. 1	
	2. Feil brukernavn og passord  3. Viser bruker at det er feil kombinasjon mellom brukernavn og passord.
Alternativ hendelsesforløp nr. 2	
	4. Ingen kvitterte avtaler eksisterer. Kun ukvitterte avtaler vises

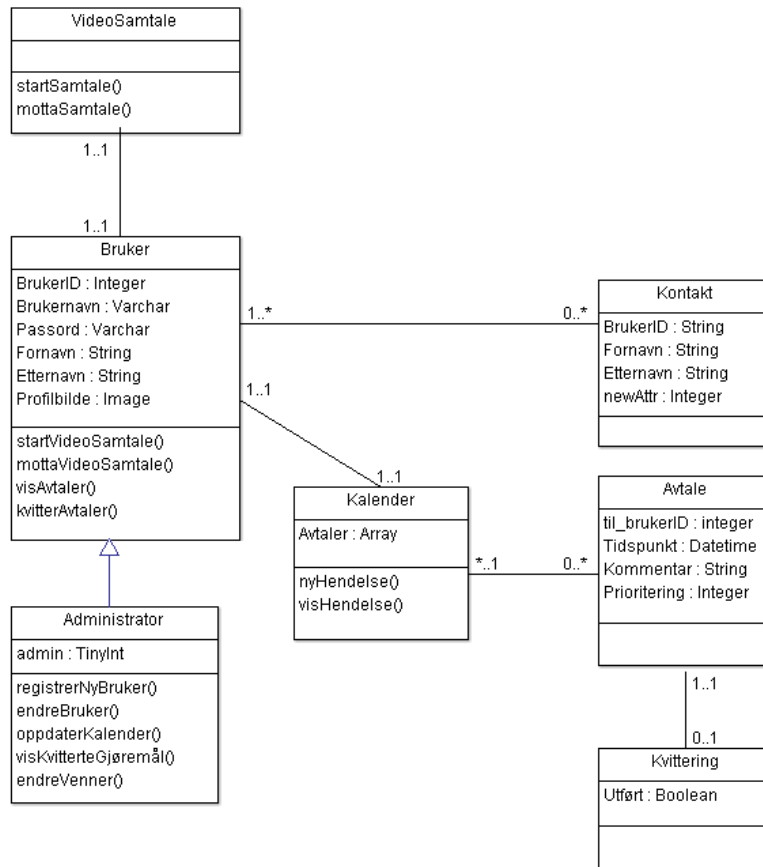
Use Case	Opprette videosamtale	
Aktør	Pleier og Tjenestemottaker	
Mål	Opprette kommunikasjonsforbindelse mellom to enheter	
Beskrivelse	Video- og samtalekommunikasjon er en essensiell bit i AKTIVe. Brukere kan ringe hverandre og hjemmetjenesten. Lyd og bilde blir overført i sanntid mellom enhetene. En bruker kan velge når samtalen skal avsluttes, vedkommende har også muligheten for å ikke godta en samtaleforespørsel.	
Type	Viktig	
Pre-betingelser	Bruker er logget inn på tjenesten AKTIVe.	
Post-betingelser	Ingen	
Spesielle krav	Ingen	
Detaljert hendelsesforløp		
<b>Aktør</b>	<b>System</b>	
1.Velger bruker som skal ringes	2.Kobler opp brukere vha av peer to peer.	
3.Mottaker godtar/avslår	4.Samtale forbundet	
Alternativ hendelsesforløp nr. 1		
	2.Bruker er ikke pålogget eller opptatt	
	4.Applikasjonen viser en feilmelding	
Alternativ hendelsesforløp nr. 2		
	3.Mottaker avslår, bruker får beskjed om dette	



Figur 4: Sekvensdiagram for 'Opprette videosamtale'



## 2.3 Klassediagram



Figur 5: Klassediagram over systemet

### 2.3.1 Forklaring til klassediagrammet

Klassediagrammet viser at en bruker har én, og bare én kalender som inneholder null eller flere avtaler. En avtale har null eller én kvittering, avhengig av om dettes kreves. Hver bruker i systemet har null eller flere kontakter i sin adressebok, mens en kontakt tilhører én eller flere brukere av systemet. Dette impliserer at flere brukere kan ha den samme kontakten i sin adressebok. Videosamtaler kan opprettes i applikasjonen, hvor en videosamtale kun tilhører én bruker. Arvestrukturen i UML-diagrammet viser at en administrator har samme attributtene og funksjonene som en bruker, men har også mulighet for å administrere systemet.

## 2.4 Overordnede operasjonelle systemkrav

### 2.4.1 Ytelse

Utvikling av applikasjoner til mobile plattformer krever at man må ta hensyn til begrenset prosesseringskraft og minne. Ved å unngå unødvendig bruk av systemressurser vil man holde liv i den bærbare enheten lenger og unngå at den går tom for batteri. Begrensning av antall kodelinjer og unødvendige funksjoner, vil holde størrelsen og minnebruken til applikasjonen nede på et akseptabelt nivå. Databaselaget består av en enkelt database som er 20MB stor. Systemet vil kunne ha svært mange brukere koblet på samtidig, uten at dette skal gå utover ytelsen. All video- og lydkommunikasjon blir strømmet ved hjelp av *P2P* teknologi, på denne måten unngår vi mellomlagring av større data.

Applikasjonen strømmer mye video- og lyddata, kvaliteten på nettverkstilkoblingen er essensiell for å oppnå en best mulig opplevelse av denne funksjonaliteten. Programvaren skal utvikles med tanke på at enheten er tilkoblet et *ADSL Wi-Fi* aksesspunkt, men video- og samtalekommunikasjon vil i praksis fungere over alternative nettverk som *3G* da med dårligere kvalitet. Det vil være mulig å begrense videokvaliteten på en svakere nettverkskobling, til fordel for bedre lyd kvalitet.

Navigering i grensesnitt skal gå umiddelbart, maksimalt et halvt sekund. Innlogging skal skje på under fem sekunder. Dette vil kunne skje raskere avhengig av hvor mange venner brukeren har i sin kontaktliste. Det blir tillatt en responstid på fire sekunder fordi innlastingen av profilbilder krever mye datatrafikk. Ved bruk av ringefunksjonen i videosamtale tjenesten, vil responstiden være begrenset av menneskelig respons. Det vil være en naturlig forsinkelse under overføring av data ved bruk av videosamtale, men dette er ikke noe brukerne vil oppleve i praksis, så lenge begge parter oppholder seg i forskjellige rom.

### 2.4.2 Sikkerhet

For å sikre at uvedkommende ikke tar i bruk applikasjonen ble det implementert en autentiseringsmekanisme hvor registrerte brukere av systemet kan logge seg inn. Brukernavn ligger lagret i klartekst i databasen, mens passord ligger som MD5-krypterte strenger. MD5-kryptering konverterer passordet til et 32 bytes hexadesimal streng som ikke har noen likheter med passordet som er kryptert. MD5 er også enveis kryptering, det vil si at man ikke får det samme resultatet ut, om man dekrypterer strengen. [1] Selvom MD5-krypterte strenger i teorien ikke skal være mulig å dekryptere, finnes det generatorer på internett som søker gjennom databaser etter lignende MD5-strenger. Disse databasene inneholder kun enkle tekststrenger, eksempler på dette kan være passord som pass123, password, admin osv.

Det er to autentiseringsnivåer, som defineres av et bit i databasen. Dette bittet er **admin**, som er av typen *tinyint*. Brukere som er registrert i databasen med bittet satt lik 1, vil ha rettigheter til å aksessere webpanelet, for å administrere systemet. Resterende brukere vil kun ha rettigheter til å logge seg inn på AKTIVe applikasjonen.

### 2.4.3 Tilgjengelighet

Under utviklingsperioden var backend delen av systemet lokalisert på høgskolen sin server, men etter endt prosjektfase overtok Gjøvik kommune driften. Arbeidstiden ved IT-tjenesten på høgskolen er mandag - fredag mellom 0800-1600. I denne tidsperioden vil det kunne oppnås en stabil oppetid, men utenom disse tidene vil det ikke være mulig å garantere dette. En viktig del av applikasjonen er avhengig av at Adobe sin Cirrus tjeneste er tilgjengelig. Cirrus er foreløpig en gratis beta tjeneste som ikke garanterer noen oppetid, men tjenesten har fungert stabilt gjennom hele utviklingsperioden. Etter utviklingsperioden på høgskolen er ferdig, vil kommunen kunne sette opp tjenesten på sin egen server. På nettbrettet vil det være mulig å spesifisere lokasjonen til skriptene som er nødvendig for at tjenesten skal fungere.

### 2.4.4 Utgivelser

Det ble utgitt en testversjon av applikasjonen etter hver sprint, hvor oppdragsgiver fikk en demonstrasjon på modulen som har blitt implementert. Det var planlagt tre testfaser hvor applikasjonen skulle blitt testet på brukergruppen. Disse testversjonene skulle ha fokus på brukergrensesnittet, da denne delen av utviklingen krever mye ekstern testing. Planlagt utgivelse av en beta versjon var planlagt i utgangen av april 2012, mens en endelig versjon ble utgitt i midten av mai 2012.

## **2.5 Begrensninger**

### **2.5.1 Software design begrensninger**

Applikasjonen er beregnet for eldre. Grensesnittet er enkelt, med få knapper og opsjoner. Benyttelsen av store knapper og tydelig farger samt en leselig skrifttype og skriftstørrelse er essensielt for vår brukergruppe. Programvaren støtter kun norsk som systemspråk. Brukerne av webpanelet er vant til like tjenester fra sin arbeidshverdag, dermed trenger ikke denne tjenesten noen form for forenkling. Det ble derfor ikke tatt noen spesielle forutsetninger med tanke på brukergrensesnittet i denne modulen.

### **2.5.2 Hardware begrensninger**

Applikasjonen optimaliseres for enheter som har Android Honeycomb som operativsystem. Enhetene må ha støtte for kamera i front for å realisere video- og samtalekommunikasjon. Enhetene som benyttes til testing har 1.5 GHz prosessor, 1 GB RAM, 1.3 Megapixler kamera i front og minst syv tommer skjerm. Dette er anbefalt spesifikasjoner, om en benytter seg av enheter med dårligere kapasitet vil en kunne merke dårligere kvalitet i video- og samtalekommunikasjonsmodulen.

## **2.6 Aspekter omkring livssyklus**

### **2.6.1 Dokumentasjon**

Dokumentasjonen opptrer i to versjoner, en teknisk dokumentasjon hvor funksjoner og kilde-koden blir beskrevet, og en brukerdokumentasjon som forklarer hvordan systemet tas i bruk. Det vil bli publisert to kopier av brukerdokumentasjonen, en skriftlig og en digital. Den digitale versjonen blir liggende ute på prosjektets nettside i to år.

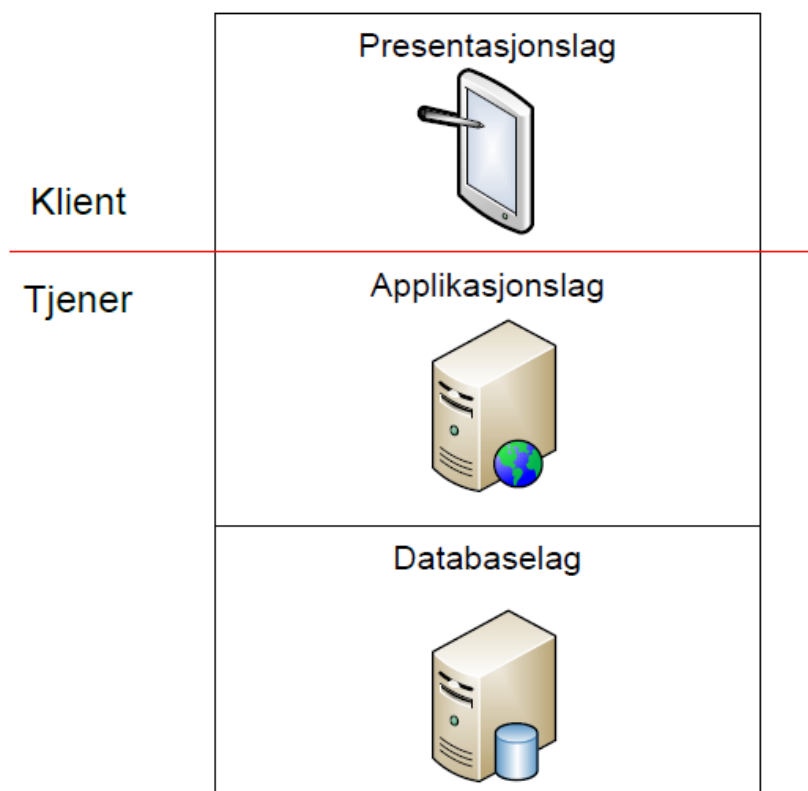
### **2.6.2 Krav til utvidelser**

AKTIVe vil kunne støtte utvidelser ved senere anledning. Kildekoden og dokumentasjon vil bli overlevert til interessenten når prosjektet er i slutfasen. Det vil bli kommunens jobb å utarbeide disse utgivelsene. Tenkte utvidelser kan være å integrere andre løsninger slik som spill eller andre nyttige applikasjoner som kommer brukergruppen til nytte.

### 3 Programvarearkitektur og design

AKTIVE består av to systemer, et webpanel og en applikasjon kjørende på nettbrett eller pc. AKTIVE applikasjonen benytter seg av tjenester på en ekstern server via internettilkobling, da er det naturlig å ha en klient/tjener arkitektur. Mekanismer for kalling på skript o.l finnes ikke på klientsiden, men slike tjenester er sentralisert på tjenersiden. Dette medfører at programvaren er delt inn i ulike lag.

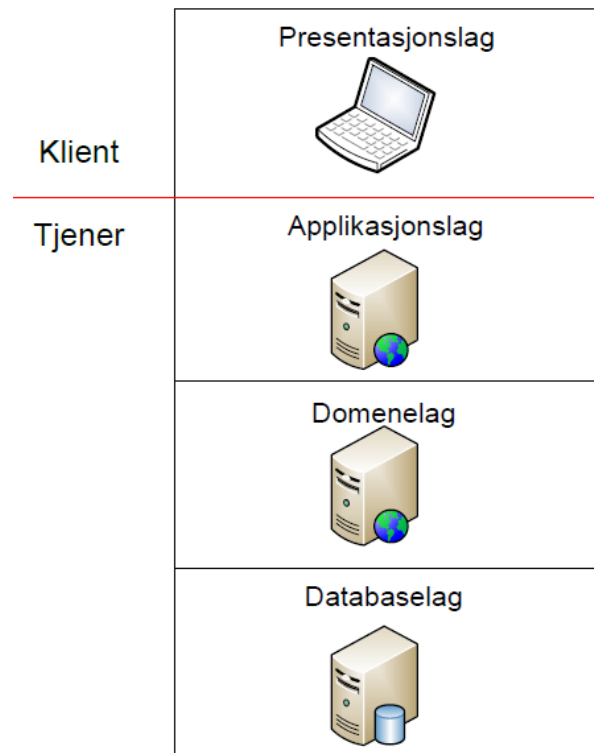
#### 3.1 AKTIVE applikasjonen



Figur 6: Arkitektur - AKTIVE applikasjonen

Illustrasjonen over viser hvordan lagdelingen av AKTIVE applikasjonen er utført. Presentasjonslaget sin oppgave er å fremvise brukergrensesnittet, i dette laget går også skillet mellom klient og tjener i systemet. Tjenestene i applikasjonslaget blir trigget av brukerinteraksjon i presentasjonslaget. Databaselaget består kun av en enkelt database som består av flere tabeller. Lese og skrive forespørsler som blir trigget i applikasjonslaget kommuniserer med dette laget for å hente ut eller skrive data fra databasen.

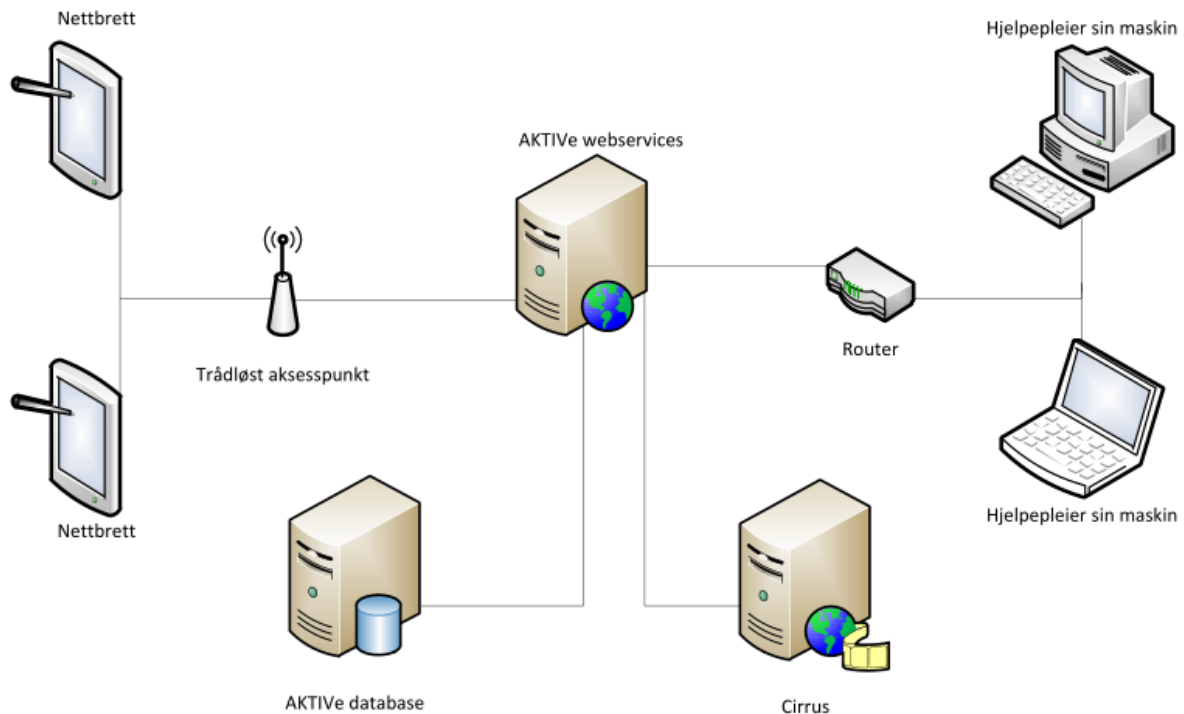
### 3.2 Webpanel



Figur 7: Arkitektur - webpanel

Arkitekturen for webpanelet er delt inn i fire lag. I presentasjonslaget ligger presentasjonskoden (HTML og CSS) for webpanelet. Oppgaven til dette laget er derfor å presentere det grafiske brukergrensesnittet. Det er naturlig å skille klient fra tjener i overgangen mellom presentasjonslaget og applikasjonslaget, da oppgaven til klienten kun er å tolke presentasjonskoden. Applikasjonslaget sin oppgave er å kontrollere at reglene som er blitt satt blir overholdt av brukeren. Input validering er en funksjonalitet som blir tilbudt i dette laget. Domenelaget sitt ansvar er å utføre handlinger mot databasen, dette kan være lesing eller skriving til databasen. Til slutt i denne lagdelingsmodellen finner vi databaselaget. I dette laget finnes det kun en database inneholdende flere tabeller, i denne databasen blir all informasjonen lagret. Databasen er modellert i seksjon "3.4 databasemodell".

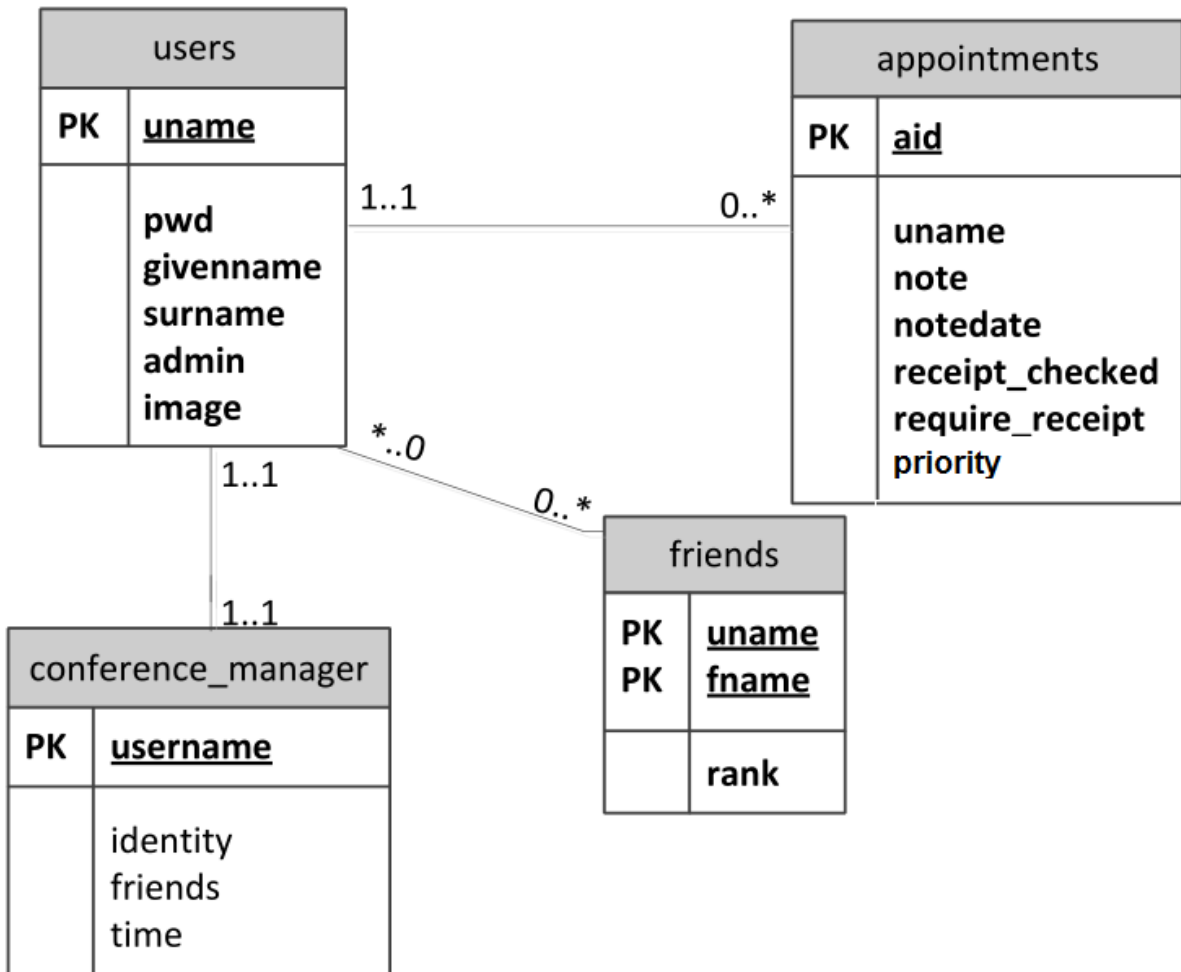
### 3.3 Deployment view



Figur 8: Deployment view av systemet

Figuren viser hele systemet og hvordan de ulike delene kommuniserer med hverandre. Nettbrettene blir koblet opp mot et trådløst nettverk, og har mulighet for å benytte seg av skriptene som kjøres på AKTIVe webservices. Denne tjenesten er kjernen i hele systemet, webservices tilbyr brukere å logge inn på AKTIVe applikasjonen, hente ut kontakter, avtaler og forbinde samtaler. På den andre siden finner vi hjelpeleierens maskin, hvor vedkommende oppdaterer avtalekalenderen, administrerer brukere og oppdaterer vennelistene. Fra webservices oppdateres de aktuelle tabellene i databasen med brukerinfo og annen data som blir generert fra skriptene i AKTIVe webservices.

### 3.4 Databasemodell

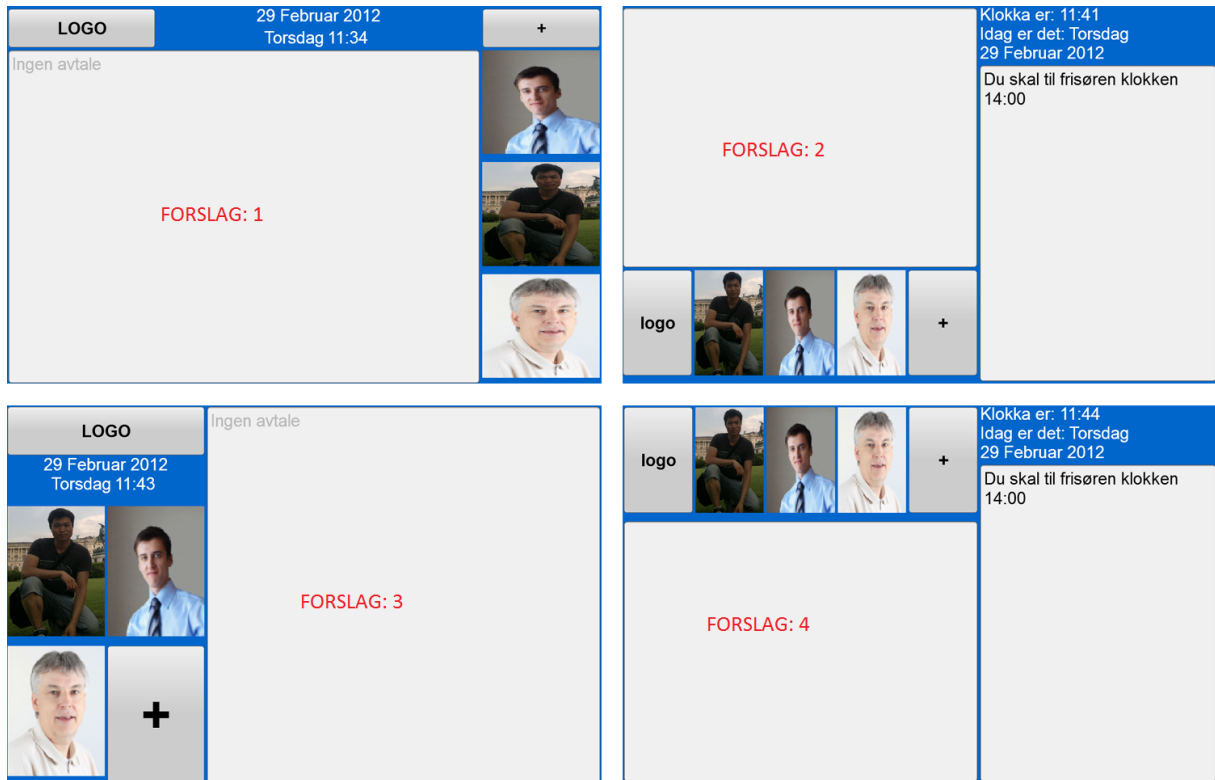


Figur 9: Databasemodell over systemet



### 3.5 Brukergrensesnitt i AKTIVe applikasjonen

Det ble presentert ulike brukergrensesnitt for interessenten og sluttbrukerne av systemet, under er de fire forslagene som ble presentert. Brukergrensesnitt nummer 1 ble tilslutt valgt.

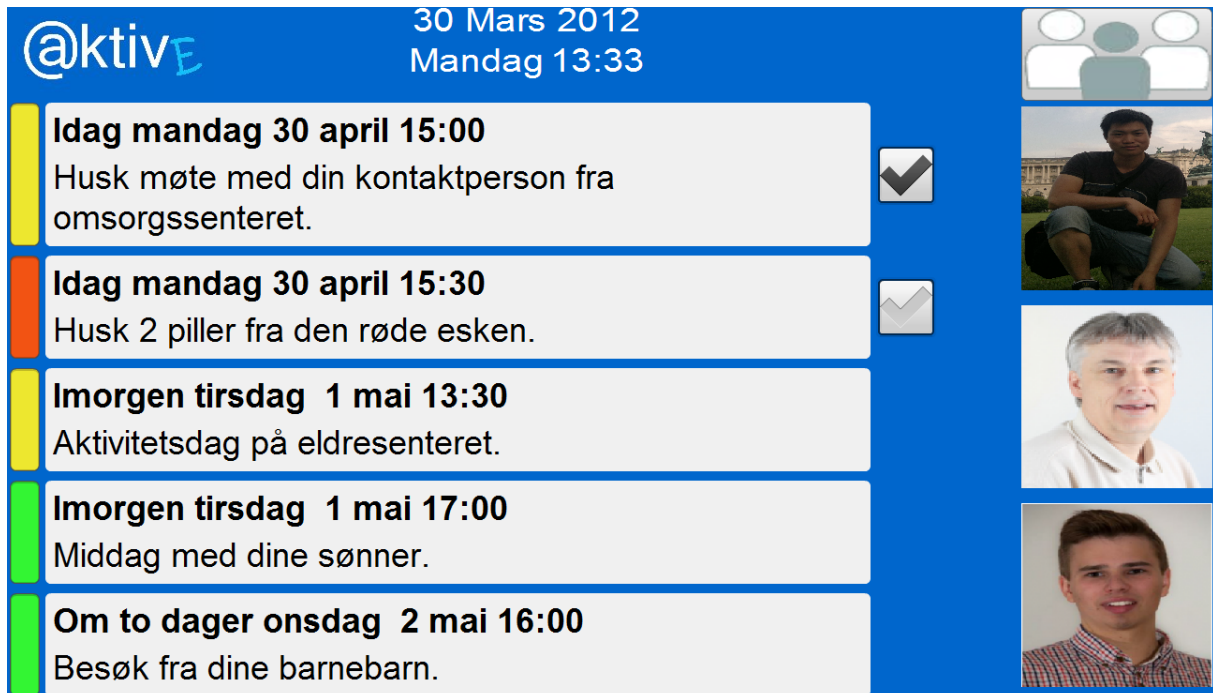


Figur 10: Forslag til brukergrensesnitt

disse brukergrensesnittene er ikke fullstendige. Forslagene ble presentert som *mockups* slik at interessenten skulle få et inntrykk av hvordan grensesnittet blir vist for sluttbrukerne.

### 3.6 Faktisk brukergrensesnitt

Illustrasjonen under viser brukergrensesnittet som ble implementert i AKTIVe applikasjonen. På venstre side blir avtalen markert med en fargekode. Den grønne fargekoden definerer at avtalen er 'lite viktig', fargen skifter også til gul eller rød, henholdsvis 'litt viktig' og 'viktig'.



Figur 11: Faktisk brukergrensesnitt i AKTIVe applikasjonen

Til høyre for avtalene er det avkryssningsbokser, hvor tjenestemottakeren kan huke av for utførte avtaler. Det er hjelpepleieren definerer om avtalen krever kvittering, da avtalen opprettes i webpanelet. Helt til høyre er listen over de mest kontaktede vennene, denne listen blir rangert ut fra algoritmen UpdateRank, som blir beskrevet senere i rapporten. Over den rangerte listen av kontakter finnes det en knapp hvor bruker kan trykke for å få en fullstendig oversikt over alle kontaktene som en kan ringe.

## 3.7 Designutfordringer

### 3.7.1 Lagring av kontakter

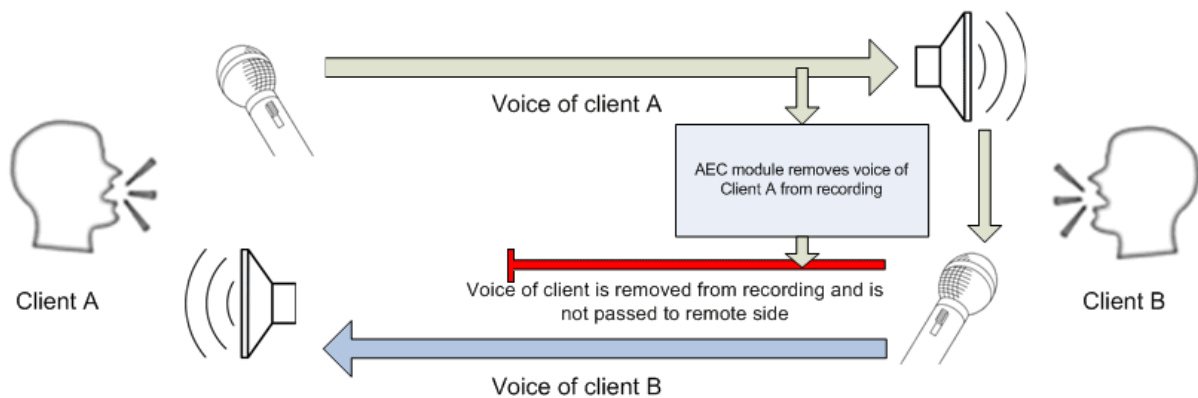
For at brukerne i systemet skal kunne kommunisere med hverandre, må det bli tilbudt en kontaktliste som holder oversikt over brukerne i systemet. I startfasen av prosjektet ble kontaktlisten implementert direkte på den håndholdte enheten, slik at alle kunne legge til alle som venn. Ved å benytte seg av denne metoden ble kontakter søkt opp i databasen ved hjelp av et PHP skript. Hvis kontakten eksisterte i databasen, ble det opprettet ett nytt objekt av denne kontakten. Interessenten ønsket i utgangspunktet at det skulle gjøres slik, men med tanke på personvernsopplysninger, ble denne løsningen ikke aktuell utover i prosjektfasen.

“Formålet med denne loven er å beskytte den enkelte mot at personvernet blir krenket gjennom behandling av personopplysninger. Loven skal bidra til at personopplysninger blir behandlet i samsvar med grunnleggende personvern hensyn, herunder behovet for personlig integritet, privatlivets fred og tilstrekkelig kvalitet på personopplysninger.”[2]

For å overholde personvernopplysningsloven, måtte en ny løsning implementeres. Funksjonaliteten for å legge til venner ble overlatt til hjelpepleieren/administrator hos kommunen. Ved å innføre denne metoden, opprettes det restriksjoner på hvem som kan være venner med hvem. Om en bruker av AKTIVe applikasjonen skal kunne legge til venner, må vedkommende kontakte hjelpepleieren. Fra sitt webpanel kan ansatte i kommunen legge til venner for en spesifikk bruker. Når kontakten blir lagt til i webpanelet vil AKTIVe applikasjonen oppdatere seg automatisk, slik at den nye kontakten blir presentert i kontaktlisten. Fordelene med å benytte seg av denne løsningen, er at brukere uten administratorrettigheter av systemet, ikke har oversikt over hvem som benytter seg av denne tjenesten.

### 3.7.2 Ekko problematikk

Når brukere kjører applikasjonen på nettbrett, kan det oppstå ekko under video- og samtalekommunikasjonen da mikrofonen fanger aktiviteten på høyttaleren. Dette er uheldig da det oppstår veldig mye støy og forstyrrende elementer under samtalen. AEC er en forkortelse på acoustic echo cancellation[3], dette er en teori om hvordan man kan fjerne ekko i samtaler, både ved hjelp av algoritmer på klient og server side. AEC kan også gjøres ved hjelp av hardware i enheten, men Google tillater oss ikke å benytte denne tjenesten i Android[4]. Da gjenstår alternativene på klient og serversiden. Serveren som forbinder samtalerne er satt opp og driftes av Adobe, denne har vi ikke mulighet for å selv konfigurere, så dette alternativet ble ikke realisert.



Figur 12: AEC i praksis<sup>2</sup>

Den eneste gjenstående muligheten da, var å implementere dette på klientsiden. Det ble først prøvd ut en enkel selvskreven algoritme som sjekker om det oppstår aktivitet på mikrofonen, om dette er tilfellet ble høyttaleren dempet. Denne algoritmen viste seg å være svært ineffektiv da utviklingsrammeverket vårt ikke har støtte for trådprogrammering, dette medfører stor bruk av ressurser på nettbrettet. Gruppen konkluderte med at vi ikke skulle bruke mer tid på dette, siden utfordringen allerede hadde krevd mye tid og ingen god løsning var i sikte. Det vil alltid bli spurt om tilgang til AEC brikken i enheten, om Google tillater dette i senere versjoner av Android, vil dette problemet bli løst uten videre endringer i kildekoden.

<sup>2</sup>Figur hentet fra: [http://lakeofsoft.com/vc/a\\_acousticechocancellation\\_AEC.html](http://lakeofsoft.com/vc/a_acousticechocancellation_AEC.html)

## 4 Implementering

### 4.1 Valg av utviklingsmiljø og språk

Vi sto fritt til å velge utviklingsmiljø og hvilket språk vi ville benytte oss av. Det var én faktor som styrte oss da vil skulle velge språk for AKTIVe applikasjonen, støtte for videokommunikasjon. Valget falt på Flex, utviklet av Adobe. Flex er et open-source rammeverk som lar oss kompilere den samme kodebasen slik at applikasjonen kan kjøre på *iOS*, *Android*, *Blackberry* og Windows. Rammeverket Flex er sammensatt av ActionScript 3.0, MXML og CSS. Førstnevnte språk brukes for å programmere logikken i applikasjonen, MXML definerer brukergrensesnittet og CSS er til for å tilpasse fargene og andre visuelle effekter i komponentene. Webpanelet er kodet i PHP, HTML, CSS, JavaScript og AJAX. Databasen er MySQL, versjon 5.0.9. For å utvikle i Flex, bruker vi Adobe Flash Builder 4.6, som opprinnelig koster penger, men Adobe tilbyr studenter en non-commercial utgave som er gratis. Webpanelet og skriptene er kodet i Notepad++, med WAMP som vi benytter av oss for testing lokalt, før panelet blir lagt ut på nettet.

### 4.2 Valg av tredjeparts tjenester

For å realisere video- og samtalekommunikasjonen ble det brukt betydelig tid i valg av tredjeparts tjenester. Valget falt til slutt på Adobe sin tjeneste Cirrus, men andre alternativer har også blitt prøvd ut. OpenTOK, et API fra selskapet TokBox var lenge et alternativ. Problemet med dette biblioteket var at den tilgjengelige versjonen kun hadde støtte for Adobe Air 2.6. Denne versjonen av Adobe Air tillater ikke brukere å velge hvilket kamera som skal benyttes, standardkameraet var satt til kamera som befant seg bak på enheten. Gruppen var i kontakt med OpenTok sin utviklerstab for å høre om vi kunne endre kildekoden, da det kun var et feil i en parameter som ble sendt til en funksjon. Selskapet tillot ikke oss dette, ei heller ønsket dem å fikse dette selv da dem jobbet med et native iOS bibliotek. Smack API var også et alternativ, men dette biblioteket fant gruppen svært vanskelig, stort og tungt. Det ville tatt for lang tid å implementere video- og samtalekommunikasjon. Adobe har et rikt utvalg av tjenester for bruker til bruker interaksjon, en av disse tjenestene heter Adobe LifeCycle. Adobe tar seg betalt for antall overført *megabyte* ved bruk av denne tjenesten, derfor ble dette uaktuelt for oppdragsgiver. Som nevnt tidligere falt valget på Adobe sin tjeneste Cirrus som oppretter video- og samtalekommunikasjon over P2P.

## 4.3 Implementasjon av webpanel

Kildekoden som realiserer funksjonalitetene i webpanelet er tildels inspirert av eksempler og prosjekter utført i faget WWW-teknologi, undervist av Øyvind Kolloen.

### 4.3.1 Registrer ny bruker

Denne modulen er meget essensiell i systemet vårt, fra webpanelet har administratorene mulighet for å registrere nye brukere av systemet. I hvert tekstfelt er det en sjekk på om bruker har fylt nok antall tegn for at brukeren skal bli registrert i databasen. Brukernavn feltet krever minst to tegn og passord feltet krever minst fem tegn. Hvis ikke disse kravene blir fulgt får bruker en melding i rød tekst under hvert tekstfelt, som ber vedkommende om å fylle inn riktig informasjon.



Registrer ny bruker

Brukernavn  
.....

Fornavn  
Etternavn

Administrator rettigheter

Velg fil Ingen fil valgt

Opprett ny bruker

**Meny**

- Hjem
- Registrer ny bruker
- Endre bruker
- Endre venner
- Ny avtale
- Kvitterte avtaler

Figur 13: Oppretting av brukere i webpanelet

#### 4.3.2 Endre bruker

Om en bruker har glemt sitt brukernavn eller passord, har administratorene av systemet mulighet for å endre disse verdiene i databasen. Denne funksjonaliteten tilbys i webpanelet. Når en klikker på linken "endre bruker" blir det presentert en nedtrekksliste over alle registrerte brukere i databasen, som vist på illustrasjonen under



Figur 14: Endring av brukere i webpanelet - Nedtrekksmeny

Brukeren velger her hvilken person i databasen som skal endres på. All informasjonen om en bruker kan endres på. Illustrasjonen på neste side, viser en skjermdump av denne funksjonaliteten i sin helhet.



Endre brukerdetaljer for hah



#### Endre personlig informasjon

Fornavn  
Etternavn

Bytt brukerinformasjon

#### Endre passord

Bytt passord

#### Endre profilbilde

Choose File No file chosen

Bytt profilbilde

#### Endre registrerte avtaler

Se avtaler

#### Slett bruker

Slett bruker

#### Meny

- Hjem
- Registrer ny bruker
- Endre bruker
- Endre venner
- Ny avtale
- Kvitterte avtaler

Figur 15: Endring av brukere i webpanelet



### 4.3.3 Endre venner

Denne funksjonaliteten er en essensiell bit i systemet, modulen i webpanelet realiserer muligheten for at brukere kan kommunisere med hverandre, i sanntid. Det er administratoren av webpanelet som velger hvilke kontakter en bruker kan opprette en samtale med. Når vedkommende klikker på fanen "Endre venner" presenteres en nedtrekksmeny som lister ut alle registrerte brukere i systemet, slik som illustrasjonen under viser.



Figur 16: Legge til venner i webpanelet - Nedtrekksmeny

Etter administratoren har valgt en bruker, vil det bli presentert en liste over denne brukers allerede lagrede venner. Ovenfor denne listen, er det en tekstboks med alle registrerte brukere i databasen som ikke allerede finnes i vennelisten. Fra denne listen, kan administratoren velge hvilken bruker som skal legges til som venn. Illustrasjonen på neste side, gir et bedre inntrykk av denne modulen.



Velg venner for henning

Haug, Frode	▲
Heggheim, Håvard	
Karlstad, Tommy	
Skjørestad, Olav	▼
Legg til	

**Meny**

- [Hjem](#)
- [Registrer ny bruker](#)
- [Endre bruker](#)
- [Endre venner](#)
- [Ny avtale](#)
- [Kvitterte avtaler](#)

Venneliste:

Harald Liodden ✖  
Lap To ✖  
Tobias Moe Thorstensen ✖  
Tom Røise ✖

Figur 17: Legge til venner i webpanelet

Om en bruker ønsker å fjerne en av sine venner, kan administratoren gjøre dette ved å trykke på det røde krysset ved siden av navnet. I bakgrunnen blir det kjørt et AJAX skript, som dynamisk oppdaterer oversikten over venner i databasen.

#### 4.3.4 Avtalekalender

Denne modulen tilbyr administratorene av systemet å oppdatere avtalekalenderne til tjenestemottakerne. Fra webpanelet kan administratorene velge dato for gjøremålet, klokkeslett, om avtalen krever kvittering og en tekstlig melding som beskriver avtalen.



Opprett avtale for henning, frode,

**Meny**

- Hjem
- Registrer ny bruker
- Endre bruker
- Endre venner
- Ny melding
- Kalender

Dato: 2012.03.23 12:40

mars 2012

ma	ti	on	to	fr	lø	sø
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

Tid 12:40

Time

Minutt

Ferdig

Figur 18: Oppretting av ny avtale

Kalenderen er laget ved hjelp av *jQuery* rammeverket. Kildekoden for pop-up kalenderen er funnet på nettsiden: <http://trentrichardson.com/examples/timepicker/>

Etter administratoren har lagt til en valgt dato og skrevet inn teksten, har en flere valgmuligheter for denne avtalen. Illustrasjonen under er ment for å vise denne funksjonaliteten i sin helhet.

The screenshot shows the 'Opprett avtale for hah,' form in the AKTIVe WebPanel. The form contains the following elements:

- A large empty text input field for the appointment description.
- A 'Dato:' field with the value '2012-05-7 00:00'.
- A 'Repetere avtale:' dropdown menu set to '1 ganger'.
- An 'Hver:' dropdown menu set to '1. dag'.
- A 'Prioritet:' dropdown menu set to 'Lite Vikti'.
- A 'Kvittering' checkbox, which is currently unchecked.
- A 'Post avtale' button at the bottom of the form.

To the right of the form is a 'Meny' (Menu) section with the following items:

- Hjem
- Registrer ny bruker
- Endre bruker
- Endre venner
- Ny avtale
- Kvitterte avtaler

Figur 19: Opprette ny avtale

Figuren over viser at administratoren har mulighet for å repetere avtalen, dette for å automatisere prosessen ved å legge til avtaler som skal inntreffe flere uker på rad. Prioriteten avtalen har kan predefineres før avtalen blir postet, ved å velge 'Lite viktig' blir avtalen markert grønn, 'Viktig' blir gul og 'Veldig viktig' blir markert rød i AKTIVe applikasjonen. Til slutt har administratoren mulighet for å bestemme om avtalen trenger kvittering, dette er spesielt egnet til avtaler som omhandler medisinerings eller andre viktige avtaler.

#### 4.3.5 Kvitterte avtaler

Tjenestemottakere har mulighet for å kvittere på avtalene som krever dette. Denne funksjonaliteten er implementert for at hjelpepleierne kan ha en oversikt over hvem som utfører gitte oppgaver. I webpanelet under fanen 'Kvitterte avtaler' har hjelpepleierne en oversikt over hvilke tjenestemottakere som har kvittert på avtalene.



Figur 20: Kvitterte avtaler

## 4.4 Implementasjon av nettbrett applikasjon

### 4.4.1 Innlogging

Før en bruker kan ta i bruk tjenestene som blir levert av AKTIVe, er en nødt til å logge inn. Avtalene og profilen for videochatten blir satt opp mot en brukerkonto, for hver eneste bruker i systemet. Livssyklusen til denne modulen starter i funksjonen `loginCheck()`, hvor det opprettes ett objekt inneholdende to tekststrenger. Den første strengen er brukernavnet som blir skrevet inn i tekstfeltet, mens den andre strengen inneholder passordet fra tekstfeltet.

```

1 private function loginCheck():void
2 {
3     var params:Object = { username:usern.text, password:pass.text };
4     loginValidator.send(params);
5     loginStatus.visible = true;
6     loginStatus.text = ATTEMPT_TO_LOGIN; //kobler til..
7 }

```

Listing 1: Logg inn fra applikasjonen

på linje tre, sendes objektet **params** til et PHP skript som validerer brukernavnet og passordet mot databasen. Før dataene kan sendes til skriptet, må det å deklarerer en `HTTPService` for at skriptet skal kunne bli kalt på fra AKTIVe applikasjonen.

```

1 <s:HTTPService id="loginValidator" url="{loginValidatorURL}"
2     useProxy="false" method="GET" result="loginResult()"
3     fault="httpfault(event)"><mx:request xmlns=""></mx:request>
4 </s:HTTPService>

```

Listing 2: `HTTPService` for innlogging

Neste side viser PHP skriptet som validerer informasjonen en bruker har tastet inn. Tilbake vil det bli returnert **1** om brukernavnet og passordet stemte med radene i databasen. Om passordet eller brukernavnet er feil, vil PHP skriptet returnere **0**.

```
1 <?php
2
3 //Defining the database credentials.
4 define( "DATABASE_SERVER" , "mysql.stud.hig.no" );
5 define( "DATABASE_USERNAME" , "*****" );
6 define( "DATABASE_PASSWORD" , "*****" );
7 define( "DATABASE_NAME" , "hpv12aktive" );
8
9 //connect to the database
10 $mysql = mysql_connect(DATABASE_SERVER, DATABASE_USERNAME, DATABASE_PASSWORD) or
11 die( mysql_error() );
12
13 //select the database
14 mysql_select_db( hpv12aktive );
15
16 //assign the data passed from Flex to this variables
17 $username = mysql_real_escape_string( trim( $_GET[ "username" ] ) );
18 $password = md5( mysql_real_escape_string( trim( $_GET[ "password" ] ) ) );
19
20 //Query the database to see if the given username/password combination is valid.
21 $query = "SELECT * FROM users WHERE uname = '$username' AND pwd = '$password'";
22 $result = mysql_fetch_array( mysql_query( $query ) );
23
24 if ( !$result )
25 {
26     $output = "0";
27 }
28 else
29 {
30     $output = "1";
31 }
32 echo ( $output );
33 ?>
```

Listing 3: Logg inn skriptet

Etter PHP skriptet har kjørt, kalles funksjonen loginResult() som sjekker om vi mottar **1**, om dette er tilfellet vil man bli sendt videre til hjemskjermen, hvor alle tjenestene blir presentert. Koden som realiserer dette blir presentert på neste side.

```
1 private function loginResult():void
2 {
3     if (loginValidator.lastResult.toString() == '1')
4     {
5         userNameInput.text = usern.text;
6         onConnect();
7     }
8
9     else
10    {
11        loginStatus.text = WRONG_USERNAME_OR_PASSWORD; //feil brukernavn eller passord
12    }
13
14 }
```

Listing 4: Logg inn skriptet

Da programvaren skal brukes av en målgruppe som ikke er vant med slike tjenester, ønsket interessenten en funksjonalitet hvor brukerne ikke behøvde å logge inn hver gang applikasjonen startet. Dette ble løst ved å ta vare på brukernavnet og passordet, og en funksjonalitet for å automatisk logge inn ved annengangs oppstart av applikasjonen. Brukernavnet og passordet lagres i et LocalSO objekt, som er av typen sharedobject. Disse dataene blir lagret som *cookies* på enheten. Måten dette gjøres på er vist med koden under.

```
1 private function saveLogin():void
2 {
3     if (rem.selected == true)
4     {
5         localSO.data.usern = usern.text;
6         localSO.data.pass = pass.text;
7     }
8 }
```

Listing 5: Kodesnutt som lagrer dataene

Når en bruker avslutter applikasjonen er disse dataene lagret, og ved neste oppstart blir vedkommende automatisk logget inn. Dette realiseres ved følgende kode:



```
1 private function checkLogin():void
2 {
3     if(localSO.data.hasOwnProperty("usern") && localSO.data.hasOwnProperty("pass"))
4     {
5         usern.text =localSO.data.usern;
6         pass.text =localSO.data.pass;
7         loginCheck();
8     }
9 }
```

Listing 6: Kodesnutt for automatisk innlogging

Logg inn funksjonaliteten er basert på en internett leksjon funnet på denne siden

[http://cookbooks.adobe.com/post\\_Create\\_a\\_login\\_system\\_with\\_Flex\\_and\\_PHP-7243.html](http://cookbooks.adobe.com/post_Create_a_login_system_with_Flex_and_PHP-7243.html)

#### 4.4.2 Video- og samtalekommunikasjon

Video- og samtalekommunikasjonen er en kompleks affære, sett fra et teknologisk perspektiv. I denne seksjonen vil denne modulen bli beskrevet detaljert. For at systemets brukere skal kunne opprette og motta videosamtaler må en først registreres i adobe sin tjeneste Cirrus. Koden under viser hvordan man registrerer brukere i databasen som holder identiteten til brukere i cirrus tjenesten.

```

1 $username= trim($_GET[ 'username' ] );
2 $identity= trim($_GET[ 'identity' ] );
3 $friends= trim($_GET[ 'friends' ] );
4 $time = trim($_GET[ 'time' ] );
5 $msg = "<?xml version=\"1.0\" encoding=\"UTF-8\"?>";
6 $msg .= "<result>";
7 $msg = "<?xml version=\"1.0\" encoding=\"UTF-8\"?>";
8 $msg .= "<result>";
9 if( $username != "" ){
10     $db=mysql_connect($host, $user, $pass) or die('<update>false</update>');
11     mysql_select_db($dbname);
12     // query to check for the username existence
13     $sql = "SELECT * FROM 'conference_manager' WHERE username = '". $username ."'";
14     $res = mysql_query($sql) or die(mysql_error());
15     if( mysql_num_rows($res) > 0){
16         // lets do an update
17         $sql_update = "UPDATE 'conference_manager' SET identity = '". $identity ."'
18             WHERE username = '". $username ."'";
19         $res = mysql_query($sql_update);
20         if( $res){
21             $msg .= "<update>true</update>";
22         } else {
23             $msg .= "<update>false</update>";
24         }
25     } else {
26         // lets do an insert
27         $sql_insert = "INSERT INTO 'conference_manager' (username, identity, time)
28             VALUES('". $username ."', '". $identity ."', NOW())";
29         $res = mysql_query($sql_insert);
30         if( $res){
31             $msg .= "<update>true</update>";
32         } else {
33             $msg .= "<update>false</update>"; } } }

```

Listing 7: PHP skript for å opprette bruker i cirrus

Dette PHP skriptet blir kjørt hver gang en bruker logger på AKTIVe applikasjonen. Før bruker starter en samtale, må kameraet og mikrofonen initialiseres. Dette gjøres ved å opprette objekter av typen Camera og Microphone, som følger med Flex rammeverket.

```

1 var mic:Microphone = getMicrophone();
2 var camera:Camera = getCamera('1');
```

Listing 8: Henter mikrofon og kamera fra enheten

Når en bruker ønsker å forbinde en samtale, er det mye prosessering av data backend. Først blir funksjonen under kalt på.

```

1 public function sendData(fname:String , lname:String):void
2 {
3     checkUserOnline = true;
4     var params:Object = { givenname1:fname, surname1: lname};
5     IDsearch.send(params);
6 }
```

Listing 9: Opprette videosamtale

Funksjonen mottar to parametere, to strenger som henholdsvis inneholder fornavn og etternavn på brukeren som skal ringes. Disse to parameteren blir brukt i linje fem, hvor et PHP skript kalles som søker i databasen etter denne brukeren.

```

1 <?php
2 require_once 'db.php';
3 $givenname = trim($_GET['givenname1']);
4 $surname = trim($_GET['surname1']);
5 $sql = "SELECT * FROM 'users' WHERE givenname = '". $givenname ." ' AND surname = '".
        $surname ." '";
6 $result = $db->query($sql);
7 $row = $result->fetch();
8 echo $row['uname'];
9 return $row['uname'];
10 ?>
```

Listing 10: Søk basert på fornavn og etternavn.

Når PHP skriptet har fullført, kalles funksjonen IDdone() automatisk. Denne funksjonen oppretter et kall mot adobe cirrus for å sjekke om brukernavnet eksisterer, samt nedlasting av bru-

kerinformasjonen. Denne informasjonen blir brukt til å sjekke om brukeren faktisk er pålogget og klar for å motta en videosamtale.

```

1 private function IDdone():void{
2     var friendID:String = IDsearch.lastResult.toString();
3     calledAFriend(userNameInput.text, friendID);
4     if(idManager)
5         idManager.lookup(friendID);
6     startCheckUserOnlineTimer();
7 }

```

Listing 11: Opprette videosamtale

idManager variabelen er av typen AbstractIdManager, en ferdiglaget klasse, skrevet av adobe. Etter den overnevnte kodensnutten har blitt kjørt, tar Cirrus-tjenesten over, og sørger for at samtalen blir forbundet. Når en bruker mottar en samtale, vil applikasjonen signalisere dette ved en ringetone, funksjonen under realiserer denne muligheten.

```

1 private function ringTone(event:SampleDataEvent):void {
2     for (var c:int=0; c<8192; c++) {
3         var pos:Number = Number(c + event.position) / Number(6 * 44100);
4         var frac:Number = pos - int(pos);
5         var sample:Number;
6         if (frac < 0.066) {
7             sample = 0.4 * Math.sin(2* Math.PI / (44100/784) * (Number(c + event.position)
8                 ));
9         }
10        else if (frac < 0.333) {
11            sample = 0.2 * (Math.sin(2* Math.PI / (44100/646) * (Number(c + event.position)
12                )))
13            + Math.sin(2* Math.PI / (44100/672) * (Number(c + event.position)))
14            + Math.sin(2* Math.PI / (44100/1034) * (Number(c + event.position)))
15            + Math.sin(2* Math.PI / (44100/1060) * (Number(c + event.position)));
16        }
17        else {
18            sample = 0;
19        }
20        event.data.writeFloat(sample);
21        event.data.writeFloat(sample);
22    }
23 }

```

Listing 12: Funksjon for å lage en ringtone

Når en bruker aksepterer forespørselen vil vedkommende motta video- og lyd fra oppringer, samt publisere sin egen video- og lydstrøm tilbake til vedkommende som startet samtalen. Koden under viser hvordan dette muliggjøres.

```
1 private function acceptCall():void {
2     stopRing();
3     incomingStream.receiveAudio(true);
4     incomingStream.receiveVideo(true);
5     remoteVideo = new Video();
6     remoteVideo.width = remoteVideoDisplay.width;
7     remoteVideo.height = remoteVideoDisplay.height;
8     remoteVideo.attachNetStream(incomingStream);
9     remoteVideoDisplay.addChild(remoteVideo);
10    // publish your stream
11    outgoingStream = new NetStream(netConnection, NetStream.DIRECT_CONNECTIONS);
12    outgoingStream.publish("media-callee");
13    var o:Object = new Object
14    o.onPeerConnect = function(caller:NetStream):Boolean {return true;}
15    outgoingStream.client = o;
16    netConnection.call(Relay, null, remoteId, Accept, userNameInput.text);
17    startVideo();
18    startAudio();
19    currentState = CallEstablished;
20    loadChatSettings();
21 }
```

Listing 13: Funksjon for å godta en samtale

Kildekoden for video- og samtalekommunikasjonsmodulen er basert på eksempelkode utgitt av adobe, opprinnelig kildekode finnes her: <http://labs.adobe.com/technologies/cirrus/samples/>

Kildekoden for å opprette brukere i Cirrus-tjenesten finnes på følgende side: <http://johnnytrops.com/blog/wp/2008/12/17/adobe-stratus-sample-code-working-with-php/>

### 4.4.3 Avtaler

Avtaler som oppdateres fra webpanelet vil bli presentert i en tekstboks. Dataene som blir lagret i databasen, blir hentet ved et PHP skript som gjør en spørring mot databasen. PHP filen inneholdene denne spørringen må bli deklartert i HTTPService, som tidligere. Avtalene blir hentet ved en enkel triviell MySQL spørring mot databasen våres. Datoformatteringen av avtalene blir gjort i dette skriptet, dette for å vise at man har en avtale 'Idag', 'Imorgen' eller 'Om to dager'.

```

1 $today = date("Y-m-d");
2 $tomorrow = mktime(0,0,0,date("m"), date("d")+1, date("Y"));
3 $intwodays = mktime(0,0,0,date("m"), date("d")+2, date("Y"));

```

Listing 14: Kodesnutt for å generere dato

Variablene som blir deklartert i koden over brukes i en if-setning for å sjekke hvordan datoen skal bli presentert.

```

1 $unix_time = strtotime($row['notedate']);
2 $day = strftime("%F", $unix_time);
3 $fulltime_no = strftime("%A %e %B %R", $unix_time);
4 if(($today <= $day) && ($day <= $intwodays)) {
5     if($day == $today) { $str.="Idag"; }
6     else if(strtotime($day) == $tomorrow) {$str.="Imorgen";}
7     else if(strtotime($day) == $intwodays) {$str.="Om to dager";}
8 }

```

Listing 15: Kodesnutt for å formatere dato

Nå vil strengen kun inneholde en gitt dag, for at brukeren også skal få med meldingen blir denne lagt på i slutten av strengen. Deretter vil strengen bli echo'et ut til AKTIVe applikasjonen. For at hjelpepleierne skal kunne holde oversikten over hvilke tjenestemottakere som har utført et beste gjøremål eller avtale, må nettbrettet kommunisere med webpanelet. Dette ble løst ved å implementere et skript som lytter på om avkrysningsboksen blir huket av.

```

1 $noteid = $_GET['aid'];
2 if($_GET['state'] == 'true')
3     $checked = 1;
4 else $checked = 0;

```

Listing 16: Kodesnutt som lytter til avkrysningsboksen

#### 4.4.4 Rangering av brukere

På fremsiden av applikasjonen blir tre kontakter presentert. Disse kontaktene er ikke tilfeldig utvalgt. Hver gang en samtale opprettes oppdateres raden **rank** i databasen som holder en oversikt over antall ganger man har kontaktet vedkommende. Skriptet som blir kalt heter `updateRank.php`. Dette er en selvskreven algoritme som oppdaterer rank verdien til den kontakten som blir oppringt.

```

1 $sql = "SELECT * FROM friends WHERE uname = '". $user. "' AND fname = '". $friend. "'";
2 $result = $db->query($sql);
3 $row = $result->fetch();
4
5 $rank = $row[ 'rank' ];
6 $rank++;

```

Listing 17: Kode for å hente kontakter fra database

For å unngå at en gammel venn som ble hyppig kontaktet en periode tilbake ikke fortsetter å toppe vennelisten langt frem i tid, har vi implementert en funksjonalitet for å nullstille rangering av venneforhold. Hvis rank verdien har blitt større enn 10, så hentes alle venner i synkende rekkefølge etter rank verdien til hvert venneforhold og puttes i et array.

```

1 if($rank>10) {
2     $sql = "SELECT * FROM friends WHERE uname = '". $user. "' ORDER BY rank DESC";
3     $result = $db->query($sql);
4
5     //puts all friends in descending order by rank in array
6     foreach($result as $row) {
7         $friend_buffer = $row[ 'fname' ];
8         array_push($f_array , $friend_buffer);
9     }

```

Listing 18: Kode for å legge brukere i et array

De tre mest populære vennene beholder sin plassering med rank 3, 2, 1 hvor høyest verdi er mest populær, resten av vennene får nullverdier.

```
1 $count = 3;
2 foreach($f_array as $data) {
3     $sql = "UPDATE friends SET rank = '". $count.'" WHERE uname = '". $user.'" AND
4         fname = '". $data.'"";
5     $sth = $db->prepare ($sql);
6     $sth->execute();
7     if($count>0)
8         $count--;
9     else $count = 0;
```

Listing 19: Kode for å rangere brukere

På denne måten sørger vi for at de tre mest kontaktede personene innenfor en rimelig periode alltid vil være raskt tilgjengelige på hjemskjermen til applikasjonen.



## 5 Testing og kvalitetssikring

Her beskrives metodikken og fremgangsmåten gruppen benyttet seg av under testing av programvaren.

### 5.1 Testgruppe

Testgruppen er sluttbrukere av systemet. Sluttbrukerene defineres som tjenestemottakerne og hjemmetjenesten i Gjøvik kommune. Testingen ble avholdt i samarbeid med Sørbyen omsorgs-senter.

### 5.2 Whitebox testing

Denne testmetoden beskrives med ulike terminologi i forskjellig litteratur. I denne rapporten benyttes uttrykket whitebox testing.

Whitebox testing er en testmetode som blir gjennomført internt i gruppen for å verifisere at koden som blir skrevet fungerer i henhold til ønsket bruk. Metodene som blir brukt er grunnleggende, hvor vi skriver inn ønsket input til programmet og vi forventer en predefinert output. Om det oppstår annen output enn det som er ønsket, vil koden bli gått igjennom trinn for trinn. På denne måten kan vi klassifisere hvor feilen oppstår under runtime.[5]

### 5.3 Blackbox testing

Blackbox testing, også kalt funksjonell testing, er en testmetode for å verifisere at programmet oppfører seg i henhold til satte funksjonelle krav.[6] Det er denne type testing som vil bli gjennomført på testgruppen, da sluttbrukerne er tilgjengelig og kan komme med innspill til implementert funksjonalitet. Feil som sees etter i en blackbox testing er:

- Feil eller *bugs* i brukergrensenettet.
- Mangel eller feil i implementert funksjonalitet.
- Feil ved initialisering av tjenester som applikasjonen er avhengig av.

Blackbox testing skal utelukkende bli gjennomført på personer utenom utviklingsgruppen, da vedkommende som har programmert den aktuelle modulen vet hvordan programflyten foregår.

### 5.4 Regresjonstesting

Regresjonstesting er en testmetodikk som avslører *bugs* eller feil i eldre moduler, når utviklingsgruppen implementerer ny funksjonalitet. I næringslivet vil det bli skrevet testscenario,

hvor man tester ny funksjonalitet først og deretter kjører eldre testscenarier. Dette er for å verifisere at den nye koden ikke affekterte den eldre kodebiten.[7] I denne bacheloroppgaven vil ikke slike testscenarier bli skrevet, men vi vil teste applikasjonen etter hver ny implementerte modul.

## 5.5 Enheter brukt under testing

### 5.5.1 HTC Flyer

Tekniske spesifikasjoner [8]	
Størrelse	195.4mm * 122mm * 13.2mm
Vekt	420 gram
Skjerm	7 tommer, touchskjerm
Prosesor	1.5 GHz
Operativsystem	Android 3.2
Lagring	32 GB
Minne	1 GB
Kamera	5 megapiksel bakkamera, 1.3 megapiksel frontkamera
Batterikapasitet	480 timer i standbymodus, 6-8 timer ved surfing/videotitting



Figur 21: Nettbrettet HTC Flyer

### 5.5.2 Samsung Galaxy Tab 1.0

Tekniske spesifikasjoner [9]	
Størrelse	175,3mm * 256,7mm * 8,6mm
Vekt	565 gram
Skjerm	10.1 tommer, touchskjerm
Prosesor	1.0 GHz dual core
Operativsystem	Android 3.1
Lagring	16 GB
Minne	1 GB
Kamera	3.2 megapixler bakkamera, 2.0 megapixler frontkamera
Batterikapasitet	2120 timer i standbymodus



Figur 22: Nettbrettet Samsung Galaxy Tab

## 6 Evaluering og avslutning

### 6.1 Resultater

Nå som utviklingsfasen er ferdiggjort er gruppen tilfreds med nedlagt arbeid og sluttproduktet. Interessentens ønsker og krav som fremkom på møtene er blitt oppfylt, noe som gir en god indikasjon på at ønsket funksjonalitet har blitt implementert. Hovedfunksjonaliteten som oppdragsgiver ønsket seg var video- og samtalekommunikasjon. Denne biten av applikasjonen var den prosessen som allokerte mest ressurser hos grupped medlemmene, men etter mye jobbing både dag og kveld ble denne modulen implementert. Etter mye testing internt ser vi at denne funksjonaliteten fungerer tilfredstillende, dette er noe interessenten også påpekte da denne modulen ble ferdigstilt.

Funksjonaliteten i webpanelet har endret seg mye gjennom utviklingsfasen, fra å starte som et panel hvor man kunne opprette brukere til å ha støtte for avtalekalender, administrering av kontakter i kontaktlisten og presentasjon av en oversikt over utførte gjøremål. Dette er en løsning som gruppen er meget fornøyd med, med tanke på hvordan webpanelet orginalt ble fremstilt.

Frekvensen av testing har ikke gått helt etter planen som vi spesifiserte i gannt-skjemaet og forprosjektet, men interessenten og gruppen har testet applikasjonen mye internt. Dette er en fase som gruppen ønsket å utføre på flere brukergrupper, men den begrensede tiden satte en hindring for dette. På tross av dette ser gruppen seg fornøyd med sluttproduktet og utførelsen av prosjektet.

### 6.2 Videre arbeid

Applikasjonen er i henhold til ønsket spesifiserte krav fra interessenten, men om gruppen skal peke på ny funksjonalitet som kunne blitt implementert finnes det forbedringspotensiale i programvaren. Webpanelet burde ha støtte for flere autentiseringsnivåer enn det som allerede eksisterer. En brukergruppe som har blitt ekskludert er familien, om webpanelet hadde hatt støtte for at familiemedlemmer kunne legge til private avtaler uten å gå gjennom hjelpepleiere, ville det økt frekvensen av avtaler og gjøremål. Den overnevnte funksjonaliteten krever da at webpanelet hadde blitt strippet for denne gruppen, da det er uheldig at utenforstående aktører som ikke har tilknytning til kommunen får muligheten til å hente ut opplysninger om alle brukere av systemet. En annen funksjonalitet som kunne blitt implementert er muligheten for å synkronisere avtalekalenderen med eksisterende løsninger på kommunen sin hjemmeside. Dette ville automatisert prosessen ved å legge til arrangementer i regi av kommunen, slik at

programvaren automatisk henter aktiviteter som blir publisert her, og lagt disse hendelsene i avtalekalenderen. Dette krever selvfølgelig at kommunen presenterer denne type informasjon i et format som gjør det mulig å ekstrahere og formatere informasjonen riktig i applikasjon, typisk formater for å løse dette kunne vært JSON eller XML.

### **6.3 Drøfting av gruppens arbeid**

I starten av prosjektfasen jobbet gruppemedlemmene samme om å finne teknisk løsninger på video- og samtalekommunikasjons modulen. Underveis i prosjekt fordelte vi oppgaver mellom gruppemedlemmene, de ulike rollene er beskrevet i seksjon 1.4.2. Gruppen konkluderte med og ser i etterkant at dette var en fornuftig måte å gjøre det på, da arbeidsprosessen ble effektivisert betraktelig. Vi har til tider viker fra denne arbeidsmåten, om en av gruppemedlemmene stod fast eller når tiden ble knapp, da vi etter beste evne prøvde å holde tidsfrister for å implementere ønsket funksjonalitet i en sprint. Kommunikasjonen innad i gruppen har fungert bra, da vi tidligere har jobbet med prosjekter sammen. Vi bestemte oss tidlig for at vi ønsket en kjernetid for bacheloroppgaven, alle hverdager mellom 0800 og 1600 har vi møtt i grupperommet og sittet sammen for å løse de problemer som har oppstått.

Kommunikasjonen med oppdragsgivere har fungert bra, begge har vært veldig responsive på mail og telefon. Gruppen skulle ønsket at arbeidsgiverne hadde vært litt mer spesifikke på hva som skulle utvikles, da gruppen til tider har oppfattet kravene litt utydelig og uklare. Vi ønsker også å påpeke at kommunen ikke tillot oss å sette systemet i drift på deres server, dette på grunn av en endringsanmodning som ikke ble gjennomført innen prosjektslutt. Det overnevnte problemet løste vi ved å lage en dokumentasjon slik at dem kan installere systemet i sin infrastruktur ved en senere anledning.

## 6.4 Konklusjon

AKTIVe har vært et utfordrende prosjekt, men gruppen ser at erfaringene fra dette prosjektet er verdt å ta med seg videre, både fra utviklingsfasen og planleggingen. Det å utføre et prosjekt som vil bli benyttet i en reell situasjon har vært interessant og lærerikt. Vi ser at mange av fagene vi har hatt på høgsolen har kommet til nytte, spesielt systemutvikling og programmeringsfagene. Vi føler at dette prosjektet har gitt oss en smakebit på hva som kan vente oss i fremtiden, spesielt med tanke på kravspesifisering og utvikling. Vi har opplevd å takle endringer i kravene og de utfordringer dette medfører, noe som til tider har vært frustrerende, men dette er en reell situasjon vi kan møte på ved senere anledninger. Gruppen har fått utfordret seg på områder vi ikke tidligere har jobbet med, dette gjelder rammeverket Flex og tilhørende utviklingsmiljø. PHP og Javascript har også vært nytt for noen av deltagerne i prosjektgruppen, en erfaring som er verdt å ta med seg videre. Gruppen synes bacheloroppgaven har vært en fin avslutning på høgsolen, da vi har fått erfart hvilke utfordringer vi kan møte på i arbeidslivet.

## Referanser

- [1] Hugh E. Williams og David Lane. *Web Database Applications with PHP and MySQL*. O'Reilly, second edition, 2004.
- [2] Lovdata. Lov om behandling av personopplysninger (personopplysningsloven). <http://www.lovdata.no/all/t1-20000414-031-001.html#1>. [Internett; lest 5-Mars-2012].
- [3] Digital Speech Algorithms. Acoustic echo cancellation and speech enhancement module for hands-free applications. [http://www.dspalg.com/AEC\\_Gen\\_new.html](http://www.dspalg.com/AEC_Gen_new.html). [Internett; lest 21-Mars-2012].
- [4] Adobe Forum. Is AEC on Android in the plans? <http://forums.adobe.com/message/4226494>. [Internett; lest 22-Mai-2012].
- [5] Laurie Williams. Whitebox testing. <http://agile.csc.ncsu.edu/SEMaterials/WhiteBox.pdf>, 2006. [Internett; lest 10-Februar-2012].
- [6] Laurie Williams. Testing Overview and Black-Box Testing Techniques. <http://agile.csc.ncsu.edu/SEMaterials/BlackBox.pdf>, 2006. [Internett; lest 20-Februar-2012].
- [7] Jung-Min Kim, Adam Porter, and Gregg Rothermel. An Empirical Study of Regression Test Application Frequency. <http://www.cs.umd.edu/~aporter/Docs/jstvr04sep.pdf>, 2004. [Internett; lest 20-Februar-2012].
- [8] HTC. HTC Flyer specification. <http://www.htc.com/jp/tablets/htc-flyer/#specs>. [Internett; lest 29-Februar-2012].
- [9] Samsung. Samsung 10.1 specification. <http://www.samsung.com/no/consumer/mobile/mobilephones/mobilephones/GT-P7510UWDNSE-spec>. [Internett; lest 29-Februar-2012].



## **7 Vedlegg**

## 7.1 A - Ordliste

1. **iOS**

Operativsystem kjørende på håndholde produkter fra Apple.

2. **Android**

Operativsystem for mobile enheter utviklet av Google.

3. **Blackberry**

Håndholdte enheter utviklet av firmaet Blackberry.

4. **P2P**

Ressursdeling på et nettverk, i motsetning til klient/tjener finnes det ingen sentral server, men brukerne deler ressursene seg i mellom.

5. **ADSL**

ADSL bruker telefonlinjen for å koble på et nettverk.

6. **WiFi**

Enheter med Wifi realiserer muligheten for å koble seg opp på et nettverk trådløst.

7. **3G**

3G muliggjør overføringen av data over det mobilnettverket.

8. **API**

API står for Application Programming Interface, et grensesnitt for kommunikasjon mellom ulike programvare.

9. **Megabyte**

Måleenhet på kapasiteten i elektroniske enheter.

10. **Bugs**

Fra det Engelske ordet 'bugs' som betyr innsekter, i dette tilfellet, sematiske feil i programkoden.

11. **jQuery**

jQuery er et javascript bibliotek. jQuery er åpen kildekode og lisensiert med MIT og GNU General Public License Versjon 2.

## 12. Cookies

Små data som inneholder informasjon om hva bruker lagrer på nettsteder, dette kan være brukernavn og passord.

## 13. Scrum Board

En oversikt over arbeidsoppgaver som skal gjøres, oppgavene som er under arbeid og arbeid som er avsluttet. Alt i inneværende sprint.

## 14. Tinyint

Et åtte bits tall.

## 15. Mockups

Uferdige produkter brukt for presentasjon av mulig sluttprodukt.

## **7.2 B - Veiledningsdokumenter gitt til sluttbrukere**

# Veiledning til bruk av:

@aktivE

## Webpanelet

1. Naviger deg til urlen, anbefales Google Chrome som nettleser:

<http://hovedprosjekter.hig.no/v2012/imt/in/aktive/webpanel/index.html>

Dette er URLen til prosjekthjemmesiden, kommunen vil etter hvert tilby denne tjenesten fra en annen URL. Foreløpig er den overnevnte som gjelder. Siden som presenteres da er:



Brukernavn

Passord

**[Har du ikke bruker? Kontakt en administrator](#)**

## Opprett nye brukere



Brukernavnet til denne bruker

Fornavn til bruker

Passordet til denne bruker

Registrer ny bruker

Brukernavn

\*\*\*\*\*

Fornavn

Etternavn

Administrator rettigheter

Etternavn til denne bruker

Velg fil Ingen fil valgt

Opprett ny bruker

Skal bruker være administrator?

Profilbildet til denne bruker

Ferdig? Opprett ny bruker

**Meny**

- Hjem
- Registrer ny bruker
- Endre bruker
- Endre venner
- Ny melding
- Kvitterte avtaler



## Endre bruker



Velg Bruker

Gundersen, Henning

OK

Velg bruker fra nedtrekkslisten

### Meny

- Hjem
- Registrer ny bruker
- Endre bruker
- Endre venner
- Ny melding
- Kvitterte avtaler



Endre brukerdetaljer for hah



Nytt fornavn/Etternavn

#### Endre personlig informasjon

Fornavn
Etternavn

Bytt brukerinformasjon

#### Endre passord

<input type="password"/>
--------------------------

Bytt passord

Endre passord?

#### Endre profilbilde

Choose File No file chosen

Bytt profilbilde

Endre profilbilde?

#### Endre registrerte avtaler

Se avtaler

Se alle avtaler/Endre avtaler

#### Slett bruker

Slett bruker

Slett bruker fra systemet

#### Meny

- Hjem
- Registrer ny bruker
- Endre bruker
- Endre venner
- Ny avtale
- Kvitterte avtaler

## Endre venner for en bruker



Velg Bruker

Gundersen, Henning

OK

Velg bruker fra nedtrekkslisten

### Meny

- Hjem
- Registrer ny bruker
- Endre bruker
- Endre venner
- Ny melding
- Kvitterte avtaler



Velg venner for hah

**Velg brukere fra listen**

Bjørn, Teodor  
Etternavn, Fornavn  
Karlstad, Tommy  
Obama, Barack

Legg til

**Legg til bruker til listen**

**Meny**

- Hjem
- Registrer ny bruker
- Endre bruker
- Endre venner
- Ny avtale
- Kvitterte avtaler

Venneliste:

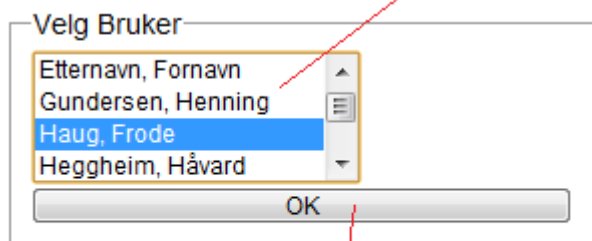
Frode Haug ✖  
Harald Liodden ✖  
Henning Gundersen ✖  
Lap To ✖  
Olav Skjørestad ✖  
Tobias Moe Thorstensen ✖

**Vennene listes opp her**

## Oppdatere kalender



Velg brukere, hold inne CTRL for å velge flere



Trykk ok

### Meny

- Hjem
- Registrer ny bruker
- Endre bruker
- Endre venner
- Ny avtale
- Kvitterte avtaler



Opprett avtale for frode,

**Skriv avtalen her**

### Meny

- Hjem
- Registrer ny bruker
- Endre bruker
- Endre venner
- Ny avtale
- Kvitterte avtaler

Dato: **Velg dato og klokkeslett**

2012-04-24 00:00

**Hvor mange ganger skal avtalen repeteres?**

Repetere avtale: 1 ganger

Hver: 1. dag  **Isåfall, i hvilket intervall?**

Prioritet: Lite Viktig  **Hvilken prioritet har avtalen?**

Kvittering  **Trengs det kvittering?**

## Oversikt over kvitterte avtaler

Velg "Kvitterte avtaler" fra menyen til høyre. Da vil skjermdumpen under bli presentert



### Kvitterte avtaler

Avtaler som er **grønne** er kvitterte!

**Idag:**

**Tobias Moe Thorstensens avtaler:**

**Husk å ta medisinen idag, Tobias!**

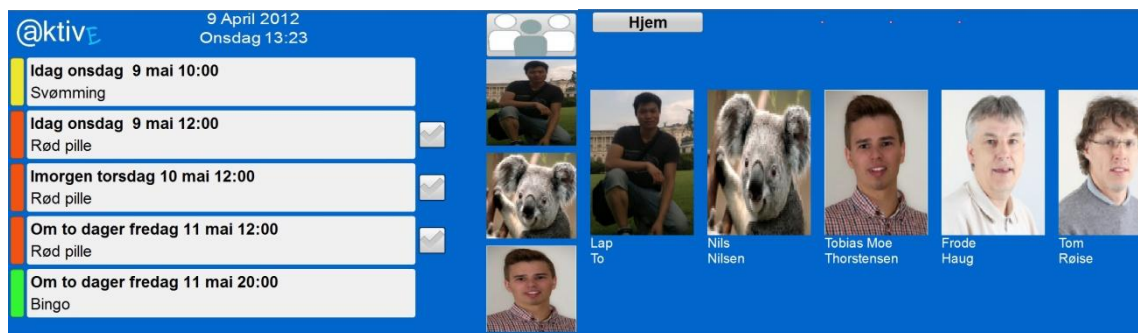
### Meny

- [Hjem](#)
- [Registrer ny bruker](#)
- [Endre bruker](#)
- [Endre venner](#)
- [Ny avtale](#)
- [Kvitterte avtaler](#)

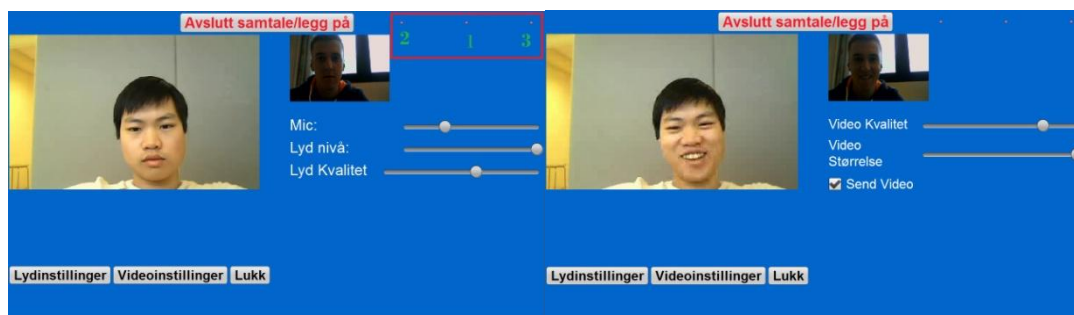
Her vil alle avtaler som har blitt kvittert i dag vist, som en ser av bildet vil avtalene som har blitt utført vist med grønn skrift, avtaler som ikke har blitt utført vil ha fargen svart.

# Nettbrett applikasjonen

Hovedsiden er der alle avtalene vises. Det vises avtaler innenfor et tidsintervall fra og med i dag og to dager frem i tid. Veldig viktige avtaler merkes med rødt, viktige med gult, og lite viktige med grønt. Avtaler som krever kvittering har en liten sjekkboks ved siden av meldingen som må hukes av. På høyresiden av skjermen finner du de tre meste kontaktede personene. Over disse finnes det en knapp man kan trykke for å få oversikt over alle kontaktene i telefonboken.



Når man ringer en kontakt eller mottar en samtale har man muligheten for å gjøre noen innstillinger. Disse innstillingene får man tilgang til ved å lokalisere tre små punkter opp i høyre hjørne av skjermen under en videosamtale. Man må trykke disse i en spesiell rekkefølge (se under) for å få opp innstillings menyen. Man kan så endre på videokvalitet, videostørrelse, lydvolum, mikrofonsensitivitet, og lyd kvalitet. Man har også muligheten for å skru av video slik at man kun sender lyd.





# Teknisk veiledningsdokument

## For oppsett av AKTIVE



## Oppsett av AKTIVE

Denne veiledningen er delt inn i to deler. En for oppsett av Webpanelet og en annen for oppsett av AKTIVE applikasjonen,

### Webpanel

For å sette opp AKTIVE Webpanel kreves følgende:

- en MySQL database
- et hjemmeområde med støtte for PHP5+

Vedlagt dokument følger filene db.sql og webpanel.zip som inneholder alle filene til webpanelet og script-filene til AKTIVE applikasjonen. Zip filen pakkes ut i det nye hjemmeområdet. Filen db.sql importeres i den nye databasen. Deretter må filen db.php som ligger i mappen /scripts og i webpanel/scripts/ konfigureres med ny url, databasenavn, brukernavn og passord for å fungere mot den nye databasen. Webpanelet skal nå være satt opp og klar til bruk.

### AKTIVE applikasjon

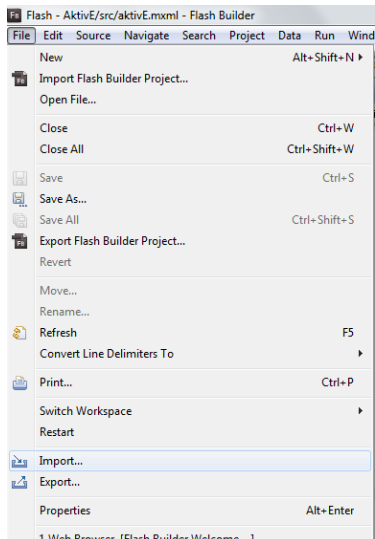
Ved oppsett på ny server lokasjon er det 2 forskjellige metoder for å få applikasjonen til å fungere.

#### Løsning 1

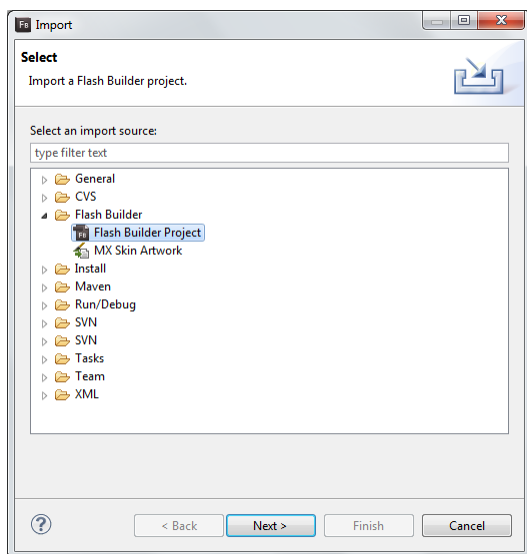
Den beste løsningen krever Adobe Flash Builder 4.6 eller høyere og endringer blir gjort direkte i programkoden.

Etter installasjonen av Flash Builder må AKTIVE prosjektet importeres.

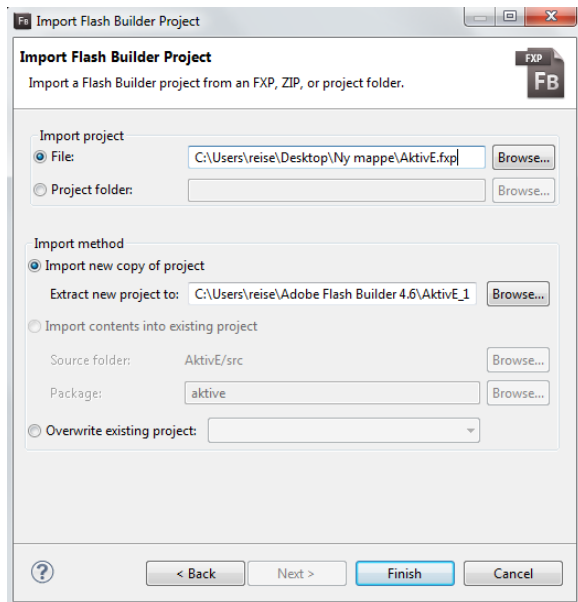
Velg "File" -> "Import"



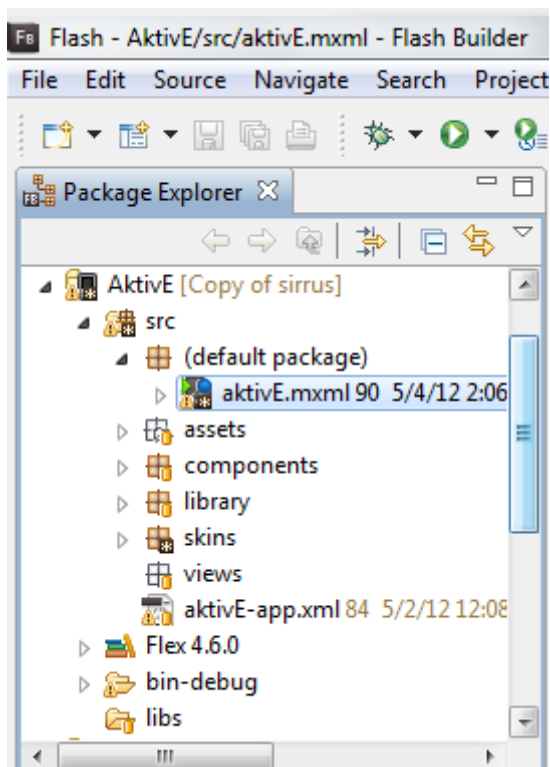
Velg Flash Builder Project



Under “Import project”, bla fram og velg AktivE.fxp filen. Trykk deretter på “Finish”



Prosjektet skal dukke opp under “Package Explorer”.  
Dobbelklikk på “AktivE-”-> “src”->”(default package)” ->”aktivE.mxml”



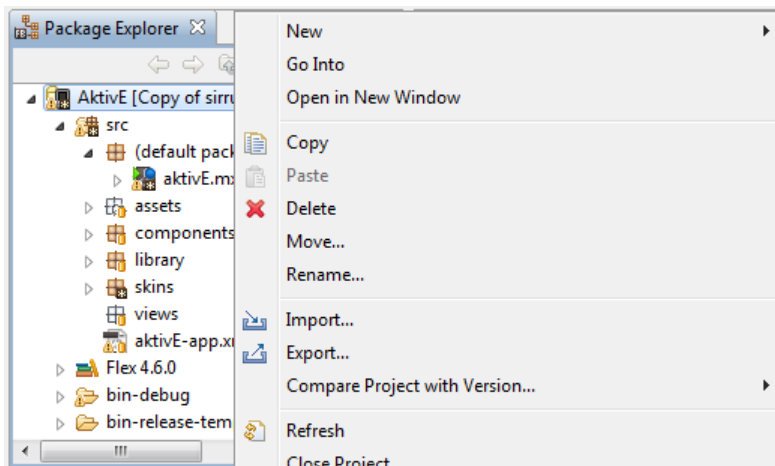
Bla nedover i koden eller søk etter ordet “scriptURL”.

http adressen bak variabelen scriptURL endres til den nye web lokasjonen. OBS! adressen skal være omfanget i gåsefnutter “ ” og avsluttes med en semikolon ; .

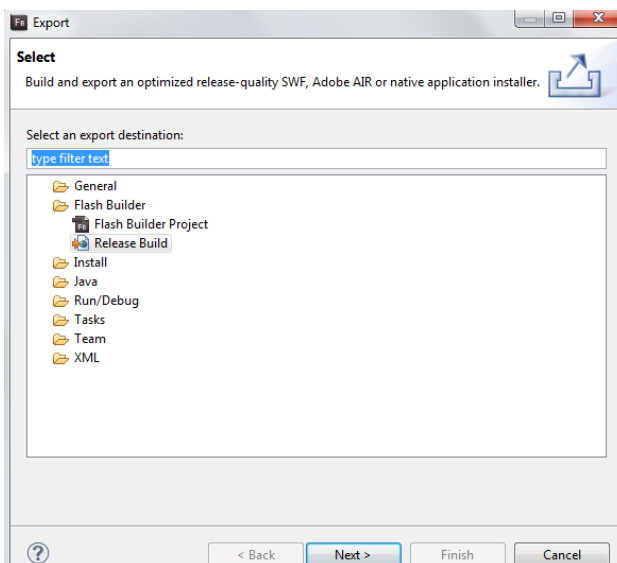
```
/* variable: scriptURL
The URL for the PHP script
*/
[Bindable]private var scriptURL:String = "http://hovedprosjekter.hig.no/v2012/imt/in/aktive/scripts/theOne.php";
```

Når dette er gjort, er det bare å eksportere den nye versjonen.

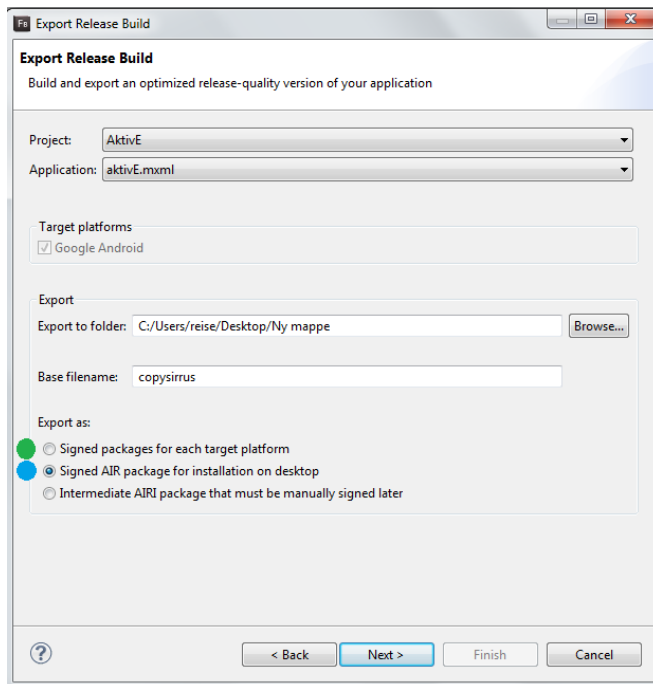
Høyreklikk på AktivE mappen under Package Explorer og velg “eksport”



Velg “Release Build” som vist på bilde og trykk “next”

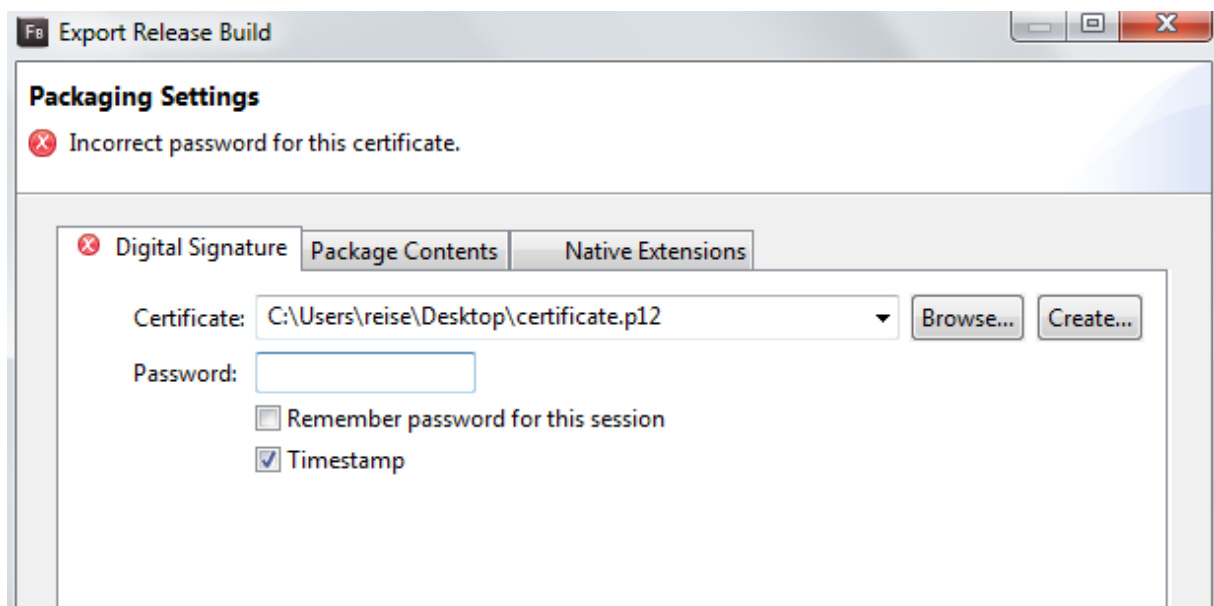


Velg det grønne alternativet “Signed packages for each target platform” for å lage en apk fil for android, eller den blå “signed AIR package for installation on desktop” for å lage en .exe for windows.



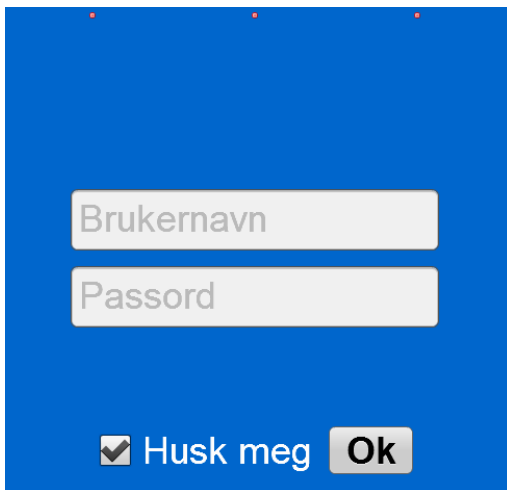
Velg fanen “Digital Signature”.

Velg “Browse..” og velg certificate.p12 filen som fulgte med prosjektet, og fyll in “hei12du” i passordfeltet uten gåsetegnene. Trykk deretter på finish.



## Løsning 2

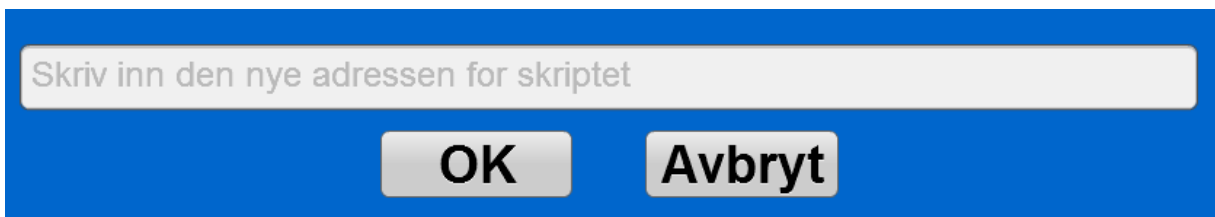
Den andre løsningen krever at endringen blir gjort på hver klient som benytter applikasjonen. Øverst i login skjermen har vi 3 røde prikker. Her må vi trykke på den midterste, deretter den venstre og tilslutt den høyre (prikkene blir grønne når dem registrerer trykket).



A screenshot of a login form on a blue background. At the top, there are three small red dots. Below them are two input fields: "Brukernavn" and "Passord". At the bottom, there is a checkbox labeled "Husk meg" which is checked, and an "Ok" button.

Når alle 3 prikkene er grønne, endrer skjermbilde seg. I det nye skjermbilde har vi muligheten til å skrive inn lokasjonen til theOne.php skriptet.

f.eks <http://hovedprosjekter.hig.no/v2012/imt/in/aktive/scripts/theOne.php>



A screenshot of a dialog box on a blue background. It features a text input field with the placeholder text "Skriv inn den nye adressen for skriptet". Below the input field are two buttons: "OK" and "Avbryt".

### **7.3 C - Statusrapporter og møtelogger**





## Møtelogg 18/1-12

**Tilstede:** Tommy Karlstad, Tom Røise, Lap To, Håvard Heggheim, Tobias Moe Thorstensen

**Sted:** A032

**Klokkeslett:** 1430-1530.

### **Referat:**

Vi hadde vårt første møte med oppdragsgiveren idag. Før møtet hadde vi forbedret oss, og valgt ut et åpent API kalt OpenTok, det viser seg å være litt problemer med dette biblioteket. Dagens tablets kommer med 2 kameraer, og velger openTok bak-kameraet som default, problemet som oppstår er at openTok kun støtter Adobe Air 2.6. Adobe har senere utgitt versjon 3.0 av Air hvor man kan velge hvilket kamera man ønsker å benytte seg av, men openTok støtter ikke denne versjonen.

Etter samtaler med oppdragsgiver har vi funnet ut at videokommunikasjon er det som står høyest på ønskelisten. Gruppen har derfor kontaktet androidutvikleren hos openTok i amerika hvor vedkommende skulle se på mulige løsninger. Andre prioritet fra oppdragsgiver var å få til et lettvindt tekstbasertmeldingssystem, hvor det ikke skjer mellomagring i et databaselager. Dette kan vise seg å være vanskelig med metoden vi har tenkt oss ut, men vi finner en løsningen på dette.

Neste møte med oppdragsgiver blir om 2 uker da gruppens første sprint er fullført. Enn så lenge vi vil jobbe for fullt med å finne en løsningen på videokommunikasjons problemet.



## Møtelogg 30/1-12

**Tilstede:** Tommy Karlstad, Tom Røise, Lap To, Håvard Heggheim, Tobias Moe Thorstensen

**Sted:** A032

**Klokkeslett:** 1530-1615.

### **Referat:**

Oppdragsgiver ville ha på plass video- og lydkommunikasjon. Denne modulen skal bli implementert i totalløsningen i løpet av denne sprinten som varer i 14 dager. Oppdragsgiver ville også at vi skulle fokusere på kvaliteten på videokommunikasjonen, noe som gruppen tar til etterretning.

Gruppen skal også få på plass et forenklet kontrollpanel, slik at brukere har mulighet for å justere hva som blir strømmen, altså valget mellom lyd og bilde, eller ingen av delene.

Neste møte som blir avholdt vil være den 13/2-12



## Møtelogg 14/2-12

**Tilstede:** Tommy Karlstad, Tom Røise, Lap To, Håvard Heggheim, Tobias Moe Thorstensen

**Sted:** A032

**Klokkeslett:** 10:00-1130

### **Referat:**

Møtet i dag demonstrerte gruppen video- og lydkommunikasjonen. Vi la også frem forslag til brukergrensesnitt for oppdragsgiver og hadde en diskusjon på hvordan dette skal se ut. Vi diskuterte også videre vei for prosjektet og hva som skal gjøres i den kommende sprinten. Arbeidsoppgaver som vil bli prioritert er:

- Brukergrensesnitt
- Online adminpanel og administrering av brukere, meldinger etc.
- Og en avtalekalender

Neste møte vil bli avholdt 27/2-12



## Møtelogg 28/2-12

**Tilstede:** Tommy Karlstad, Lap To, Håvard Heggheim, Tobias Moe Thorstensen

**Sted:** A032

**Klokkeslett:** 1430-1530.

### **Referat:**

I dagens møte ble det diskutert hva vi skal implementere i denne sprinten. Vi snakket om at vennelisten skal være en funksjon på webpanelet og ikke i selve applikasjonen. Dette medfører at pleietrengende ikke kan legge til venner selv. Gruppen skal også forbedre grensesnittet til en enklere og mer intuitiv løsning.



## Møtelogg 12/3-12

**Tilstede:** Tommy Karlstad, Lap To, Håvard Heggheim, Tobias Moe Thorstensen

**Sted:** Gjøvik Rådhus

**Klokkeslett:** 1300-1500

### **Referat:**

Dette møtet ble slått sammen med første test. På møtet ble applikasjonen og webpanelet tatt veldig godt i mot. Det kom mange forslag til hva som kan endres og forbedres i denne sprinten. Vi fikk fastsatt hvilken design som passet best. Under er punktene vi skal jobbe med i denne sprinten

- Design, farger og logo
- Lage et felt som definerer administrator
- Endre semantikken på webpanelet og i rapporten
- Ranking av venner



## Møtelogg 10/4-12

**Tilstede:** Tommy Karlstad, Tom Røise, Lap To, Håvard Heggheim, Tobias Moe Thorstensen

**Sted:** A032

**Klokkeslett:** 10:00-10:45

### **Referat:**

I dette sprintmøtet viste gruppen frem funksjonaliteten slik at det er mulig å kvittere for utførte gjøremål. Sammen med oppdragsgiverne diskuterte vi også veien videre i prosjektet. Vi kom frem til at dette blir den siste sprinten når det kommer til utvikling, hvor den siste sprinten vil det bli fokusert på å legge programvaren over til kommunens server. I denne sprinten vil vi fokusere på bug-fixes, kategorisering av avtaler, mulighet for å skjule datostemplingen i avtaler og pynte på brukergrensesnittet.



## Møtelogg 23/4-12

**Tilstede:** Tommy Karlstad, Tom Røise, Lap To, Håvard Heggheim, Tobias Moe Thorstensen

**Sted:** A032

**Klokkeslett:** 10:00-11:30

### **Referat:**

Holdt det siste møtet i dag, gruppen er inne i siste sprinten. Vi skal jobbe med følgende funksjonalitet:

- Front-End i webpanelet
- Bugsfikser og logikk i nettbrett applikasjonen
- Popup for avhuking av kvitteringer

Etter dette er ferdig vil programvaren bli testet og deployet på kommunens server.



## Statusrapport 20/2-2012

Fremdriftsplanen har til nå blitt fulgt gruppen er ajour med sprintene og planlagte tester er tilrettelagt. Ingen foreløpig endringer i gantt-skjemaet til gruppen. Organiseringen av gruppen er etter en autonomisk tankegang, man har selv ansvaret for handlinger man utfører. Håvard er fortsatt prosjektleder og Tobias er webansvarlig/dokumentansvarlig. Lap sitt hovedfokus til nå har vært å kode applikasjonen på nettbrettet, men hele gruppen har vært delaktig i kodingen på applikasjonen og webpanelet. Problemstillingene gruppen har møtt på har vært mange:

- Hvilket API skal vi velge for å realisere video- og lydkommunikasjon?
- Hvordan kan vi optimalisere video- og lydkvaliteten til et optimum?
- Hvordan skal vi få til kalender- og meldingssystemet i applikasjonen?

Måten vi har løst problemene på har vært å søke på internett etter APIer og teste ut disse for å se om dem utfører det vi ønsker. Optimaliseringen av video- og lydkvaliteten har blitt løst ved hjelp av koding, omstrukturering av kode og testet på ulike trådløse nettverk. Testingen har til nå foregått på høgskolen sitt trådløse nettverk og gruppe-medlemmenes private hjemmenettverk. I nærmeste fremtid vil gruppen få muligheten for å møte sluttbrukeren av systemet, hvor vi da satser på å få mer konkret tilbakemeldinger på hva som kan gjøres bedre.

Rapportskrivingen er ajour, gruppen fikk laget et første utkast av kravspesifikasjonen innen estimert dato. Kravspesifikasjonen har blitt litt endret, men gruppen satser på å ha den klar innen neste sprint.



Som sagt har gruppen møtt på problemer mtp. APIet for Video- og lydkommunikasjonen, men mye jobb og sene kvelder falt denne biten på plass, hvor lydkvaliteten nærmer seg Skype-kvalitet. Oppdragsgiver er veldig interessert i å få på plass mye funksjonaliteten, og derfor er det viktig at gruppen får på plass kravspesifikasjonen så kjapt som mulig.

Gruppen har gjennomført to sprinter hittil, hvor essensen har vært å få på plass video- og lydkommunikasjonen og forbedre denne. Webpanelet skulle startes på i løpet av den andre sprinten, hvor denne snart er ferdig. Rapporten er også selvfølgelig under kontinuerlig arbeid.

Tidsfristene er overholdt med god margin, hvor vi har levert alt i god tid før fristen. Gruppen føler seg meget fornøyd med progresjonen hittil i prosjektet. Motivasjonen falt litt da vi møtte på problemet med video- og lydkommunikasjonen, men gruppen fikk en vitamininnsprøytning da denne biten ble implementert, så alle deltageren er ved godt mot og jobber målrettet mot oppgavene arbeidsgiver gir oss i forkanten av en sprint. Samarbeidet er det ingenting å si på, det fungerer da vi tidligere har jobbet sammen på andre prosjekter. Arbeidsformene har vært at hvert gruppemedlem har tatt på seg en eller flere arbeidsoppgaver som blir løst i løpet av dagen. Gruppen har møttes på tildelt grupperom fra 08:00-16:00, noen dager litt lengre, andre dager litt kortere.

Kontakten med veileder Frode Haug har fungert supert. Vi avholder møter hver torsdag klokken 11, hvor gruppen kommer med status på prosjektarbeidet og får gode konstruktive tilbakemeldinger på rapporten.



## Statusrapport 20/3-12

### **Fremdrift**

Gruppen er i god gang med fremdriften av programvaren. Vi ligger foran skjemaet med utviklingen, men den første testen ble utsatt pga. sykdom fra interessentens siden. Testen ble gjennomført mandag 12/3-12, hvor gruppen viste frem webpanelet og nettbrett applikasjonen for seks av kommunens ansatte på rådhuset. Gruppen er nå halvveis inne i den fjerde sprinten, av totalt syv sprinter. I denne sprinten blir det jobbet med brukergrensesnittet i webpanelet og nettbrett applikasjonen, basert på tilbakemeldingene fra testen.

### **Organisering av gruppens arbeid og ansvarsområder**

Lap er hovedutvikleren for nettbrettapplikasjonen, mens Håvard jobber med backend i webpanelet og Tobias jobber i frontend av webpanelet, samt ansvarlig for rapporten og øvrige dokumenter.

### **Klargjøring av problemstilling**

I starten av sprinten diskuterte vi sammen med interessenten hvordan vi skulle legge til venner som kunne kontaktes via samtale- og videokommunikasjon. I starten av prosjektet kunne alle brukere søke hverandre opp, men i tråd med personvernsløven fant interessenten og gruppen ut at dette ikke var godt nok. Derfor må administratorene av webpanelet bestemme hvilken personer som kan være venner og kontakt hverandre, dette for at ikke alle brukere av systemet skal vite hvem som er en tjenestemottaker i kommunen.

### **Rapportskriving**

Rapporten er vi godt i gang med, det skrives i henholdt til gannt-skjemaet, nemlig overveis i prosjektets levetid. Rapporten inneholder nå en innledning, kravspek, arkitektur og design bit, implementerings kapittel er snart ferdigskrevet.

**Muligheter/trusler/problemer?**

Siden forrige sprint har gruppen ikke møtt på flere trusler eller problemer. Det eneste har vært små problemer med formattering av dato og å få omgjort morgen dagens dato til "i morgen" i tekstboksen hvor meldingene presenteres.

**Hva er avsluttet/Hvilke oppgaver er ferdige?**

Video- og samtalekommunikasjonen er så å si ferdig, gruppen jobber kun med små bugs og tweaks. Webpanelet blir også ferdig i løpet av denne sprinten, innloggingen er også ferdig.

**Hva er under arbeid?**

Webpanelet er under arbeid, den avsluttende fasen. Brukergrensesnittet i nettbrett applikasjonen er også under arbeid. Kalender og meldingssystemet er også under arbeid.

**Er tidsfristene overholdt?**

Tidsfristene er overholdt, det eneste avviket vi hittil har hatt på gannt-skjemaet er den utsatte testen.

**Gruppens motivasjon**

Gruppens motivasjon er fortsatt god, det jobbes veldig godt med oppgaven og progresjonen er god. Det merkes til tider at det skal bli godt med en ukes pause i påskeferien.

**Hvordan oppleves kontakten med veileder**

Som før er kontakten med veileder veldig god og vi mottar gode konstruktive tilbakemeldinger, intet å utsette på dette.



## Statusrapport 14/5-2012

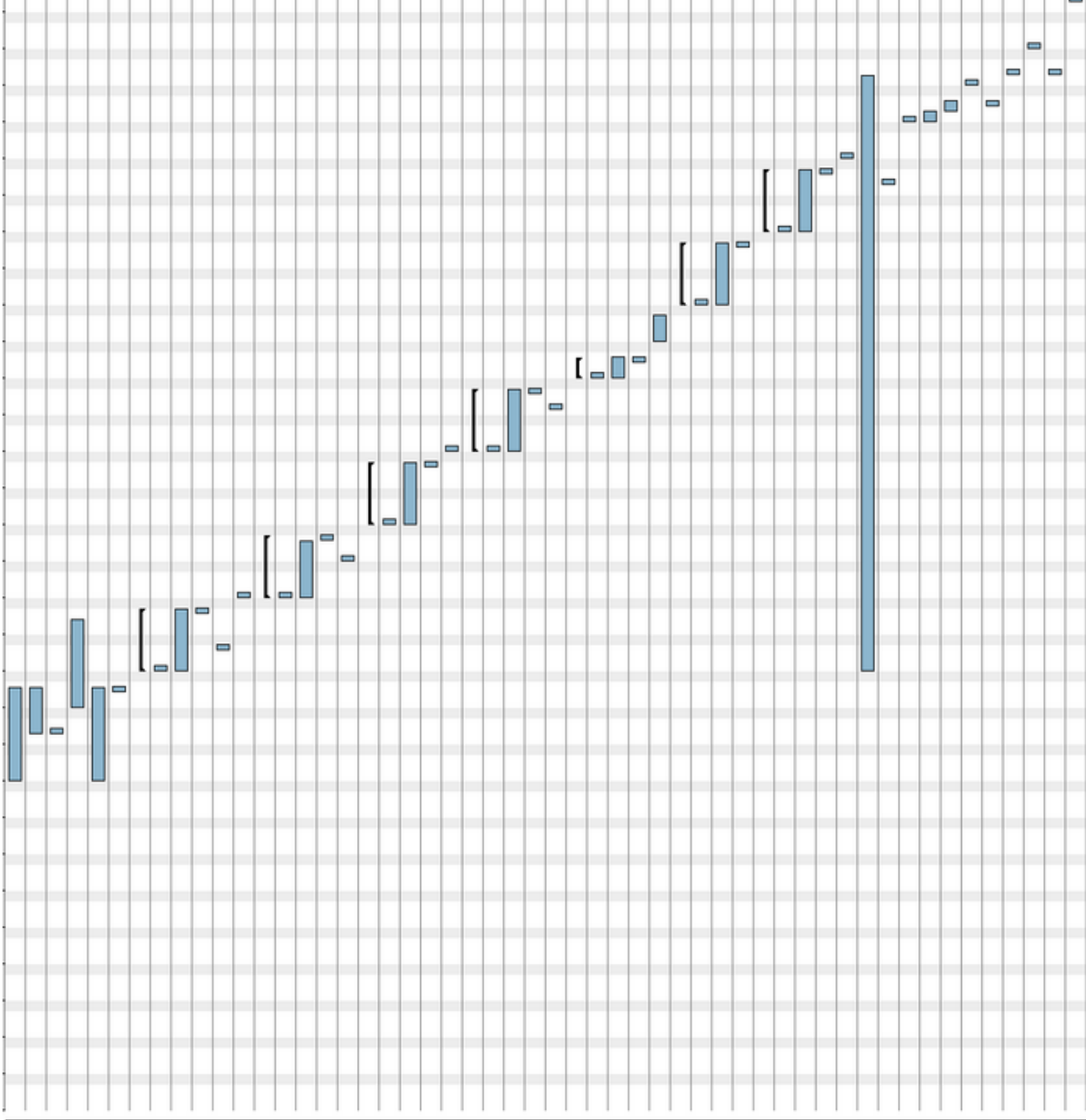
Fredriften i prosjektet er à jour, vi ligger foran tidsskjemaet noe vi er veldig fornøyd med. Organiseringen av arbeidsfordelingen i gruppen er den samme som tidligere, hvor Lap har hatt ansvaret for Flex applikasjonen, Håvard har kodet backend i webpanelet og Tobias har skrevet rapport og kodet frontend i webpanelet. Problemstillingen har ikke endret seg i prosjektfasen, men kravene har som smått blitt endret, disse endringene fremkommer i møteloggene som ligger på nettsiden til prosjektet. Rapporten har som tidligere blitt skrevet underveis og blir ferdigstilt i løpet av syv dager.

Gruppen har ikke møtt på de store problemene siden sist statusrapport. Det eneste som er verdt å påpeke er en bug i video- og samtalekommunikasjonen. Dette dreide seg om forskjell i debug-versjonen og deploy-versjonen . Problemet lå i eksekveringen av kode, det var en del funksjoner som ikke ble eksekvert i riktig rekkefølge. Gruppen konkluderte med at dette var på grunn av for rask prosessor på testmaskinene, utrolig nok. Denne buggen er rettet opp og gruppen har ikke funnet flere bugs i skrivende stund.

Gruppen er ferdig med utviklingsarbeidet og alle de planlagte sprintene. Dette medfører at det ikke vil bli implementert ny funksjonalitet, kun retting av bugs. Gruppen jobber med kommentering av koden, slik at vi kan generere en dokumentasjon for at programvaren skal kunne videreutvikles. Rapporten er heller ikke helt ferdig, men som nevnt tidligere vil den bli ferdigstilt i løpet av en drøy uke. Tidsfristene er overholdt, det eneste som gjenstår er levering av rapporten i fronter samt utarbeidelse av en plakate samt presentasjonen av oppgaven.

Motivasjonen til gruppemedlemmene er ikke svekket, men det skal bli godt å bli ferdig med oppgaven, da det har krevd mye tid som vi kunne brukt til andre fag på skolen. Som før oppleves kontakten med veileder meget godt, gode konstruktive tilbakemeldinger og gode råd.

#### **7.4 D - Faktisk gannt-skjema**



Forprosjekt
Product backlog
Møte med oppdragsgher
Utbetjelse av kravspøk
Testing av tekniske løsninger
Levering av forprosjekt og artale
Sprint 1
Sprint planning meeting
Utbjle modul
Sprint review meeting
Etablering av nettside
Test hos kommunen
Sprint 2
Sprint planning meeting
Utbjle modul
Sprint review meeting
Statusrapport
Sprint 3
Sprint planning meeting
Utbjle modul
Sprint review meeting
Test hos sluttbruker
Sprint 4
Sprint planning meeting
Utbjle modul
Sprint review meeting
Statusrapport
Sprint 5
Sprint planning meeting
Utbjle modul
Sprint review meeting
Påsketferie
Sprint 6
Sprint planning meeting
Utbjle modul
Sprint review meeting
Sprint 7
Sprint planning meeting
Utbjle modul
Sprint review meeting
Beitautgjvelse
Utbetjelse av rapport
Levering av publiseringartale
Statusrapport
Korrekturlesing av sluttreport
Fulføring av sluttreport
Rapport til trykk
Offisiell utgivelse
Levering av rapport
Utbetjelse av plakat
Elektronisk levering av rapport
Muntlig presentasjon

## **7.5 E - Forprosjekt**





GJØVIK KOMMUNE

## **Forprosjekt Bacheloroppgave 2012**

*Dette dokumentet inneholder beskrivelse av prosjektet, rammer, mål og systemutviklingsmodell*

Skrevet av Lap To, Håvard Andreas Heggheim og

Tobias Moe Thorstensen

# 1. Mål og rammer

## 1.1 Bakgrunn

I dagens samfunn er det mange eldre som sliter med å henge med på teknologiens utvikling. Den eldre generasjonen har ofte lite erfaring med bruk av digitale verktøy som for eksempel nettbrett og smarttelefoner. Vi tror, og mener at dagens applikasjoner og verktøy er altfor avansert og krevende i bruken for de eldre. Vi ser at dagens applikasjoner ofte kommer med mange muligheter og opsjoner, som til tider kan øke læringskurven for nye brukere.

De fleste applikasjoner og digitale verktøy i dag er lett forståelige for den yngre generasjonen, men nærmest umulig å forstå for den eldre generasjonen. Vår oppgave er å lage en applikasjon som vil være lett å bruke, med et simpelt og forståelig grensesnitt. Den eldre generasjon går per dags dato glipp av mange unike muligheter som den nye teknologien gjør tilgjengelig. Disse verktøyene/applikasjonene har muligheten til å forenkle livet deres betraktelig. En simpel oppgave som å lese avisen kan gjøres enklere på en håndholdt enhet. Man reduserer behovet for fysiske bevegelser, ved å unngå å gå til butikken for å kjøpe avisen. De fleste nyheter kan i dag også leses helt gratis via internett. Hvis man er veldig engasjert i en nyhetssak har man også muligheten til å dele sine meninger, så hele verden kan lese, reflektere og kommentere deres tanker og ideer.

Teknologiske verktøy er også en fantastisk portal for underholdning. Både Bingo og Yatzy kan spilles med mange tusen personer fra hele verden. Vi skal utvikle en applikasjon som passer spesielt godt for eldre med tanke på at i dag sitter det mange alene i sofaen uten mulighet for å kunne komme seg ut og sosialisere seg med andre. Det blir da et stort behov for ansatte fra hjemmetjeneste som kan komme og prate litt med dem f.eks når en god venn har gått bort, eller butikken har gått tom for smør. Vi ønsker å tilby en høyteknologisk enkel, og sosial løsning som letter arbeidet for hjemmetjenesten, og samtidig sprer glede hos eldre, der de får større muligheter til sosialere seg i omgangskretsen sin.

## **1.2 Prosjektmål**

Målet med AKTIVe er å forenkle kommunikasjonen mellom omsorgstjenesten og hjemmeboende pasienter. Et annet mål er at AKTIVe skal kunne brukes i mellom pasientene som en sosial kommunikasjonsplattform. Ved å få lettet kommunikasjonen mellom helsepersonell og pasienter, vil vi kunne spare omsorgstjenesten for mye arbeid. Hjelpepersonell bruker i dag mye tid på å kjøre til og fra pasienter, noe som er veldig ressurskrevende, ineffektivt, og lite miljøvennlig.

Gruppen skal utvikle en web applikasjon som skal spesielt være utviklet for Android baserte nettbrett, men vil ha muligheter for å kunne fungere på andre plattformer. En ferdig utviklet og stabil versjon skal være ferdig innen utløpet av mai måned 2012. Den skal hovedsaklig fungere som en kommunikasjonsplattform for omsorgstjenesten. Hvor helsepersonell kan utveksle meldinger og sende avtalepåminnelser til pasienten.

Applikasjonen skal inneholde muligheter for video – og samtalekommunikasjon. I tillegg skal den fungere som en informasjonsportal for nyheter, værmeldinger, aktiviteter o.l.

## **1.3 Rammer**

- Nettbrettet må ha støtte for kamera i front.
- Klienten/plattformen må ha støtte for Adobe air.
- Applikasjonen må være brukervennelig med et ryddig og oversiktlig grensesnitt da sluttbrukere ikke er vant brukere av en slik tjeneste.
- Nettbrettet må ha støtte for WiFi og/eller 3G
- Nettbrettet må ha støtte for minnekort for at kameraet skal kunne virke ordentlig.

## **2. Omfang**

### **2.1 Oppgavebeskrivelse**

Oppgaven er å utvikle en applikasjon hovedsaklig rettet mot nettbrett som kjører Android.

Applikasjonen som utvikles skal tilby den eldre en enkel kommunikasjonsplattform, oversikt over siste nyhetssaker og værmeldingen. Utfordringen i prosjektet blir derfor ikke bare å produsere koden, men også utvikle et brukergrensesnitt som sluttbrukeren forstår, dermed er vi nødt til å redusere kompleksiteten i designet.

Vi som gruppe har også lyst til å kikke på muligheten til å tilby f eks familiemedlemmer eller andre godkjente personer tilgang til å utføre videosamtaler med pasientene.

### **2.2 Avgrensninger**

Fokuset blir å lage et brukervennlig grensesnitt, muligheten for hjelpepleieren å sende skriftlige meldinger til brukeren og videokommunikasjon mellom pasienter og hjelpepleier. Applikasjonen skal først og fremst fungere på Android baserte enheter og muligens optimaliseres for andre plattformer i fremtiden.

Applikasjonen er ikke ment for å sende sensitiv informasjon og sikkerheten blir derfor begrenset til en autentiserings server som krever brukernavn og passord når applikasjonen blir kjørt. På klientensiden vil den automatisk huske brukernavn og passord da denne skal ikke innholde noe sensitiv informasjon om brukeren.

Gruppen benytter ”OpenTok”, et bibliotek for videokommunikasjon for å redusere unødvendig arbeid.

## 2.3 Programvare som vil bli benyttet under prosjektperioden

Gruppen vil benytte seg av ulike programvarer for å kunne utføre oppgaven i best mulig ånd.

- **Subversive**
  - Programvare for å holde versjonskontrollering av kildekoden som blir produsert innad i gruppen.
  
- **Google Docs**
  - Cloud-basert teksteditor for samkjøring av rapport og forprosjekt.
  
- **ArgoUML**
  - Open Source programvare for å modellere programvaren. Typisk Use Case og klassediagram.
  
- **Adobe Flash Builder**
  - Integrated Development Enviroment for utviklingen av selve programvare.
  
- **Trello**
  - Cloud-basert oppslagstavle for å holde orden på oppgaver innad i en sprint.
  
- **GanttProject**
  - Program for å opprette og vedlikeholde et Gannt-skjema.

## **3. Prosjektorganisering**

### **3.1 Ansvar og roller**

#### **Oppdragsgiver**

Oppdragsgiveren vår er Høgskolen i Gjøvik i samarbeid med Gjøvik Kommune. Representant fra høgskolen er Tom Røise. I tillegg vil Tommy Karlstad og Einar Malmedal fra Gjøvik kommune bidra til å svare på teoretiske spørsmål angående oppgaven.

#### **Veileder**

Kontaktpersonen vår er Frode Haug. Frode vil være vår største ressurs når det kommer til tekniske spørsmål knyttet til oppgaven.

#### **Prosjektleder**

Gruppen har bestemt at et selvstyre eller en tilnærming til autonomisk styreform passer best. Om det oppstår konflikter innad i gruppen vedrørende avgjørelser er det Håvard Andreas Heggheim som har siste ordet.

#### **Webansvarlig**

Webansvarlig i prosjektet vil være Tobias Moe Thorstensen. Oppgaven til webansvarlig vil være å holde hjemmesiden vedlike samt oppdatere og skrive nyhetsartikler.

#### **Delt ansvar**

Gruppemedlemmene blir enig innad i gruppen om mindre oppgaver som må utføres. Hvis det oppstår uenigheter eller andre problemer vil det være prosjektlederens ansvar å få orden sakene.

### **3.2 Rutiner og regler i gruppa**

Hver deltager på gruppen har signert en kontrakt hvor det har blitt enighet om å dedikere minst 30 timer i uken på prosjektet. Vi sitter på skolen og jobber i vårt tildelte grupperom fra 08.00 - 16.00. Noen av gruppemedlemmene har forelesningstimer som vil foregå i dette tidsrommet, men resten av tiden blir brukt på å jobbe med prosjektet. Deviasjoner fra dette vil diskuteres muntlig innad i gruppen.

Se vedlagt dokument om gruppregler for mer utfyllende informasjon om regler i gruppen.

## **4. Planlegging, oppfølging og rapportering**

### **4.1 Utviklingsmetodikk**

Gruppen har valgt at vi benytter oss av utviklingsmodellen Scrum. Utviklingsmodellen er som kjent en smidig utviklingsmodell som lar gruppen være åpen for endringer ganske så kjapt gjennom prosjektarbeidet. Dette gir oss en frihet med tanke på endringer og ny funksjonalitet som oppdragsgiver ønsker. Daglige “stå opp møter” vil bli holdt i i starten av dagen slik at vi kan oppsummere arbeider som har blitt gjort den foregående dagen. Grunnpilaret i scrum er sprinter, dette for å organisere hvilke funksjonaliteter som skal implementeres. Gruppen har i lag bestemt at en sprint skal gå på 2 uker, hvor vi implementere funksjonalitet steg for steg. Den tradisjonelle metoden med post-it lapper blir pensjonert til fordel for webapplikasjonen Trello, hvor det er “online post-it lapper”.

### **4.3 Plan for statusmøter og beslutningspunkt**

Da gruppen benytter seg av utviklingsmetodikken scrum er vi avhengig av å planlegge hver sprint. Her vi planleggingen bestå av diskusjoner om hvilke funksjonaliteter som skal implementeres. Denne form for møteaktivitet blir kalt for sprint backlog på fagspråket. Om man følger metodikken til punkt og prikke, skal denne loggen bestemmes av produkteier, men siden vår oppdragsgiver er ekstern og ikke har mulighet for å møtes oss hver 14. dag, ser vi i gruppe oss nødt til å bestemme prioriteringene selv ut i fra gitte prioriteringer med oppdragsgiver i startfasen av prosjektet.

Etter hver sprint skal implementer funksjonalitet demonstreres for oppdragsgiver, men igjen, vår oppdragsgiver er ekstern og har derfor ikke mulighet for å møte oss etter endt sprint. Vi i gruppen vil benytte oss av Sprint retroperspektiv som er en fase i scrum hvor vi ser tilbake på forrige sprint, hvor vi evaluerer denne og kommer med forslag til hva som kunne blitt gjort bedre og tar med oss dette til neste sprint.

## 5. Kvalitetssikring

### 5.1 Dokumentasjon, standardbruk og kildekode

Gruppen benytter JsDoc, et dokumentasjonsverktøy som autogenererer en lettleslig og pen dokumentasjon basert fra kommentarer i kildekoden.

### 5.2 Konfigurasjonsstyring

Vi kommer til å benyttes oss av Subclipse for Eclipse for å holde versjonskontroll. Vi har benyttet oss av denne plugin til Eclipse tidligere, og har erfaring med hvordan den fungerer. Vi kommer til å kjøre hyppige oppdateringer slik at hvert medlem på gruppen alltid jobber på siste tilgjengelig versjon i repository. Vi kommer også til å fordele oppgavene slik at vi aldri jobber på samme fil samtidig, av tidligere erfaringer blir dette bare rot.

### 5.3 Risikoanalyse

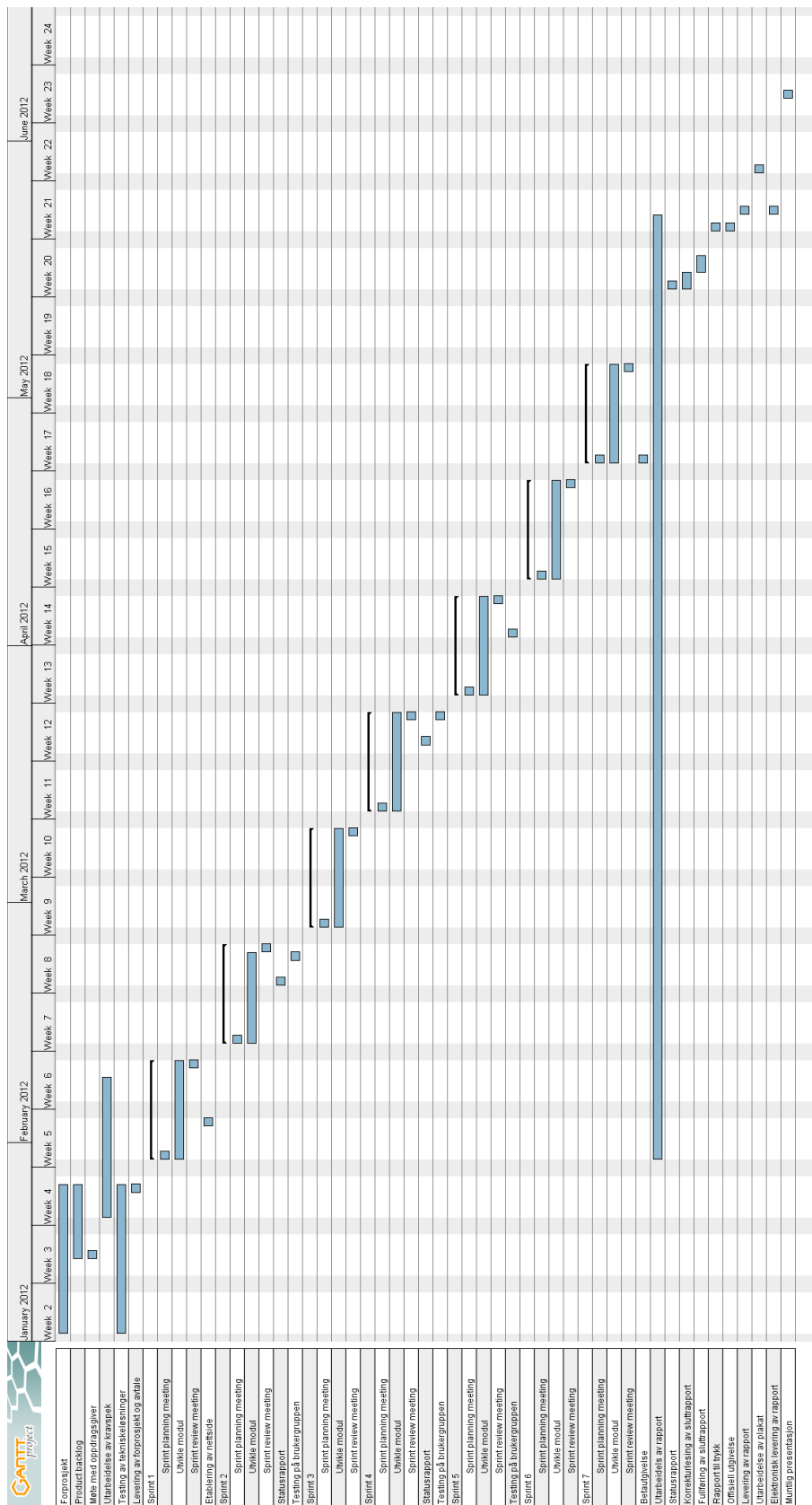
Tabellen under gir et inntrykk av risikoen til prosjektet. Koeffisientene som blir oppgitt under er oppgitt i en skala fra 1 – 10. Utfallet blir da basert på den matematiske formelen:

$$Risiko = \frac{Sannsynlighet}{Konsekvens} \times 100$$

Type	Sannsynlig	Konsekvens	Risiko
Sykdom innad i gruppen	3.0	8.0	37,5 %
Feil på utstyr	1.5	9.0	16.6 %
Komplekst design	2.0	7.0	28,5 %
Deler av applikasjon møter ikke kravspesifikasjonen	2.5	8.5	29,4 %
Liten kompetanse innad i gruppen	3.0	9.0	33,3 %
Dårlige estimater	2.0	7.5	26,6 %



## 6. Plan for gjennomføring



## 7.6 F - Grupperegler

# Gruppregler - kontrakt

Prosjektleder: Håvard A. Heggheim

Prosjektleder har ansvar for at oppgaven blir gjennomført innen gitte tidsrammer. Prosjektleder har også ansvar for å holde en jevn fordeling av arbeidsoppgaver mellom de ulike deltagerne på gruppen. Prosjektlederen vil også alltid ha den avgjørende stemmen ved uenigheter innad i gruppen.

Ethvert gruppelem har rettighet til å snakke/signere for gruppen.

Enhver deltager skal dedikere 30 timer hver uke til å jobbe med Bachelor-oppgaven. Vi sitter på skolen og jobber, om ikke annet er avtalt.

Det skal til enhver tid praktiseres Clean Coding\* når applikasjonen utvikles.

Ved ferie/reise må dette avtales i god tid, og evt. planlagt arbeid i dette tidsrommet må gjøres på forhånd.

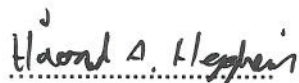
Deltagere skal sette av tid til å hjelpe hverandre hvis det trengs hjelp med å løse et problem i forbindelse med en oppgave.

Grupperommet skal til enhver tid holdes ryddig og rent. Det skal ikke kastes mat som lett begynner å lukte i søppelet.

Eventuelle kostnader som måtte oppstå under prosjektet fordeles mellom deltagerne.

Dersom en av deltagerne på gruppen ikke utfører avtalt arbeid/oppgave gis det først et muntlig advarsel, hvis problemene vedvarer vil deltageren få et skriftlig varsel, før et evt med møte veileder der personen kan bli utelatt fra gruppa.

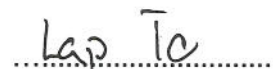
\*<http://www.amazon.com/Clean-Coder-Conduct-Professional-Programmers/dp/0137081073>



Håvard Heggheim



Tobias Moe Thorstensen



Lap To

## **7.7 G - Prosjektavtale**



HØGSKOLEN I GJØVIK

## PROSJEKTAVTALE

mellom Høgskolen i Gjøvik (HiG) (utdanningsinstitusjon),

Gjøvik kommune, Helse og omsorg (oppdragsgiver), og

LAD TO

Håvard Andreas Heggheim

Tobias Moe Thorstensen (student(er))

Avtalen angir avtalepartenes plikter vedrørende gjennomføring av prosjektet og rettigheter til anvendelse av de resultater som prosjektet frembringer:

1. Studenten(e) skal gjennomføre prosjektet i perioden fra 9/1-12 til 23/5-12.

Studentene skal i denne perioden følge en oppsatt fremdriftsplan der HiG yter veiledning. Oppdragsgiver yter avtalt prosjektbistand til fastsatte tider. Oppdragsgiver stiller til rådighet kunnskap og materiale som er nødvendig for å få gjennomført prosjektet. Det forutsettes at de gitte problemstillinger det arbeides med er aktuelle og på et nivå tilpasset studentenes faglige kunnskaper. Oppdragsgiver plikter på forespørsel fra HiG å gi en vurdering av prosjektet vederlagsfritt.

2. Kostnadene ved gjennomføringen av prosjektet dekkes på følgende måte:
  - Oppdragsgiver dekker selv gjennomføring av prosjektet når det gjelder f.eks. materiell, telefon/fax, reiser og nødvendig overnatting på steder langt fra HiG. Studentene dekker utgifter for trykking og ferdigstillelse av den skriftlige besvarelsen vedrørende prosjektet.
  - Eiendomsretten til eventuell prototyp tilfaller den som har betalt komponenter og materiell mv. som er brukt til prototypen. Dersom det er nødvendig med større og/eller spesielle investeringer for å få gjennomført prosjektet, må det gjøres en egen avtale mellom partene om eventuell kostnadsfordeling og eiendomsrett.
3. HiG står ikke som garantist for at det oppdragsgiver har bestilt fungerer etter hensikten, ei heller at prosjektet blir fullført. Prosjektet må anses som en eksamensrelatert oppgave som blir bedømt av faglærer/veileder og sensor. Likevel er det en forpliktelse for utøverne av prosjektet å fullføre dette til avtalte spesifikasjoner, funksjonsnivå og tider.
4. Den totale besvarelsen med tegninger, modeller og apparatur så vel som programlisting, kildekode, disketter, taper mv. som inngår som del av eller vedlegg til besvarelsen, gis det en kopi av til HiG, som vederlagsfritt kan benyttes til undervisnings- og forskningsformål. Besvarelsen, eller vedlegg til den, må ikke nyttes av HiG til andre formål, og ikke overlates til utenforstående uten etter avtale med de øvrige parter i denne avtalen. Dette gjelder også firmaer hvor ansatte ved HiG og/eller studenter har interesser.

Besvarelser med karakter C eller bedre registreres og plasseres i skolens bibliotek. Det legges også ut en elektronisk prosjektbesvarelse uten vedlegg på bibliotekets del av skolens internett-sider. Dette avhenger av at studentene skriver under på en egen avtale hvor de gir biblioteket tillatelse til at deres hovedprosjekt blir gjort tilgjengelig i papir og nettgave (jfr. Lov om opphavsrett). Oppdragsgiver og veileder godtar slik

offentliggjøring når de signerer denne prosjektavtalen, og må evt. gi skriftlig melding til studenter og dekan om de i løpet av prosjektet endrer syn på slik offentliggjøring.

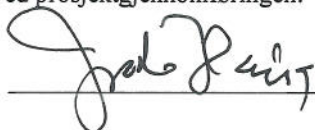
5. Besvarelsens spesifikasjoner og resultat kan anvendes i oppdragsgivers egen virksomhet. Gjør studenten(e) i sin besvarelse, eller under arbeidet med den, en patentbar oppfinnelse, gjelder i forholdet mellom oppdragsgiver og student(er) bestemmelsene i Lov om retten til oppfinnelser av 17. april 1970, §§ 4-10.
6. Ut over den offentliggjøring som er nevnt i punkt 4 har studenten(e) ikke rett til å publisere sin besvarelse, det være seg helt eller delvis eller som del i annet arbeide, uten samtykke fra oppdragsgiver. Tilsvarende samtykke må foreligge i forholdet mellom student(er) og faglærer/veileder for det materialet som faglærer/veileder stiller til disposisjon.
7. Studenten(e) leverer oppgavebesvarelsen med vedlegg (pdf) i Fronter. I tillegg leveres et eksemplar til oppdragsgiver.
8. Denne avtalen utferdiges med et eksemplar til hver av partene. På vegne av HiG er det dekan/prodekan som godkjenner avtalen.
9. I det enkelte tilfelle kan det inngås egen avtale mellom oppdragsgiver, student(er) og HiG som nærmere regulerer forhold vedrørende bl.a. eiendomsrett, videre bruk, konfidensialitet, kostnadsdekning og økonomisk utnyttelse av resultatene.

Dersom oppdragsgiver og student(er) ønsker en videre eller ny avtale, skjer dette uten HiG som partner.

10. Når HiG også opptrer som oppdragsgiver trer HiG inn i kontrakten både som utdanningsinstitusjon og som oppdragsgiver.
11. Eventuell uenighet vedrørende forståelse av denne avtale løses ved forhandlinger avtalepartene i mellom. Dersom det ikke oppnås enighet, er partene enige om at tvisten løses av voldgift, etter bestemmelsene i tvistemålsloven av 13.8.1915 nr. 6, kapittel 32.

12. Deltakende personer ved prosjektgjennomføringen:

HiGs veileder (navn):



Oppdragsgivers  
kontaktperson (navn):

Tommy KARLSTAD

Student(er) (signatur):

Oddis Moe Morsensen

dato 25/1-12

Håvard A. Heglein

dato 25/1-12

Lap Te

dato 25/1-12

dato \_\_\_\_\_

Oppdragsgiver (signatur):

Tommy Karlstad

dato 25/1-12

IMT Dekan/prodekan (signatur):



dato

25/2-12

## 7.8 H - Dokumentasjon av kildekode

Dokumentasjonen for AKTIVe applikasjonen finnes på følgende side

<http://hovedprosjekter.hig.no/v2012/imt/in/aktive/Dokumentasjon/>



## **7.9 I - Logg og timeføring**

# Arbeidslogg over sprinter

Dette dokumentet inneholder utdrag fra alle gruppe medlemmenes logger, komprimert til å omhandle hver utførte sprint.

## **Sprint #1**

Den første sprinten allokerte video- og samtalekommunikasjonen mye tid. Vi testet ut mange løsninger. Gruppen presenterte OpenTok sin løsning ovenfor oppdragsgiver. Gruppen hadde også hyppig kommunikasjon med utviklerne bak OpenTok. Forprosjektet ble også skrevet i løpet av denne sprinten. Rapporten ble også startet på i denne sprinten.

## **Sprint #2**

Denne sprinten ble brukt til å spesifisere kravene i samarbeid med oppdragsgiver, samt vise frem løsningen vi hadde funnet ut ved hjelp av Adobe Cirrus. Vi var også godt i gang med PHP skriptene for og registre brukere. Nettbrett applikasjonen ble også kodet i denne sprinten, kontaktlisten ble kodet og et søk for finne brukere i systemet.

## **Sprint #3**

Webpanelet ble kodet i denne sprinten, vi hadde fått mye ønsket funksjonalitet som oppdragsgiver ønsket å ha på plass i løpet av sprinten. Det var en del bugs i video- og samtalekommunikasjons modulen som ble fikset, det eneste som ikke ble fikset var ekko problematikken. Kravspesifikasjonen ble ferdig i løpet av denne sprinten. Webpanelet har nå støtte for å legge til avtaler.

## **Sprint #4**

Webpanelet ble jobbet med i denne perioden, arkitekturfasen ble skrevet i rapporten og støtte for rangering av brukere i nettbrett applikasjonen. Avtalefunksjonaliteten i webpanelet er blitt kodet om etter oppdragsgivers ønske. Vennefunksjonaliteten i nettbrett applikasjonen har også blitt ferdig kodet i denne sprinten.

## **Sprint #5**

Webpanelet har fått funksjonalitet for å vise frem kvitterte avtaler, og nettbrett applikasjonen har nå støtte for å vise fargekoder basert på avtalens prioritet satt i webpanelet.

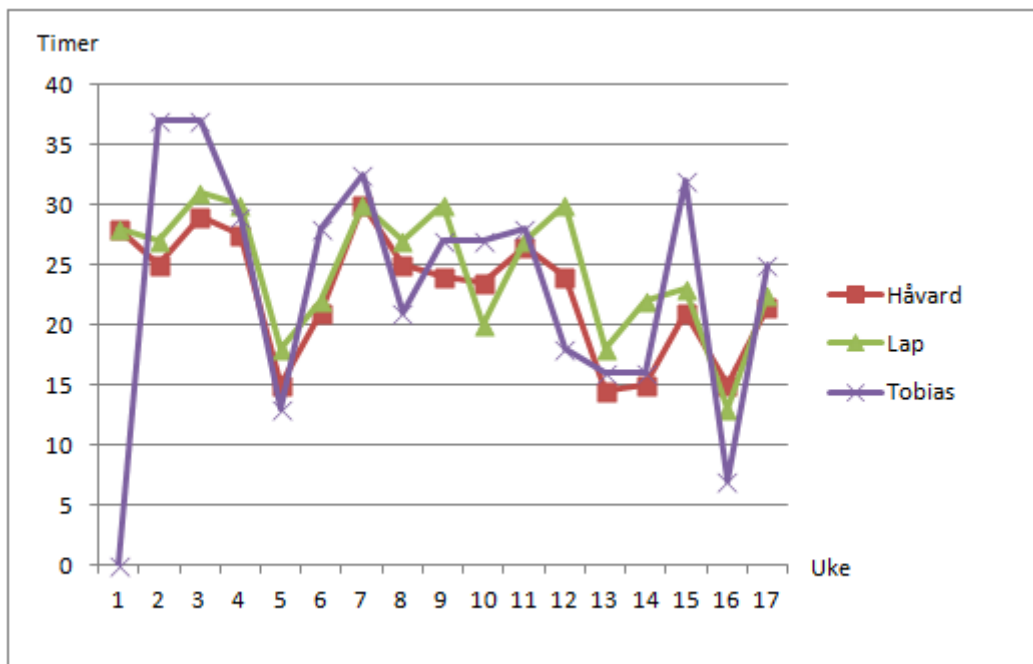
## Sprint #6

I denne sprinten har vi jobbet med funksjonaliteten ”automatisering av avtaler”, slik at man kan repetere en avtale. I nettbrett applikasjonen har gruppen fikset noen bugs. Rapporten begynner å bli ferdig nå, alle kapitlene har blitt skrevet ferdig.

## Sprint #7

Vi har ikke utføre mye arbeid i forhold til å implementere ny funksjonalitet, men heller holdt fokus på å fikse bugs i programvaren. Rapporten har også blitt skrevet på.

## Graf over arbeidsmengden per uke per gruppemedlem



Timeantallet som er lagt inn i grafen over gjelder kun for utførte sprinter og arbeidet med forprosjektet. Dette vil si at gruppen ikke har logget arbeidet i slutfasen av prosjektet. Arbeidet med presentasjonen og slutføringen av rapporten er ikke inkludert i grafen, dette medfører at det har oppstått avvik i timeantallet.

