



Forprosjektrapport

Wallboard for Zisson

Bachelorprosjekt 2020 / 2021

Gruppe 32

Martine Holmberg s236808

Camilla H. Carlsen s197235

Synne Angell s331153

Nikola Sekiw s333975



OSLO METROPOLITAN UNIVERSITY
STORBYUNIVERSITETET

zisson

Innholdsfortegnelse

Presentasjon	3
Prosjektets gruppemedlemmer	3
Oppdragsgiver	3
Veileder fra OsloMet	4
Sammendrag	4
Dagens situasjon	4
Mål og rammebetingelser	5
Teknologier/ verktøy	5
Kravspesifikasjon	6
Funksjonelle og ikke-funksjonelle krav	6
Systembeskrivelse	7
Løsning/alternativer	8
Analyse av virkninger	8
Risikoanalyse	9
Fremdriftsplan	10
Arbeidsmetodikk	11
Tentativ arbeidsplan	13

Presentasjon

Prosjektets gruppemedlemmer

Gruppen består av fire studenter som har jobbet tett sammen i over 2,5 år, der alle går siste året på bachelor ved OsloMet. Tre av gruppens medlemmer (Nikola, Camilla og Martine) går på Anvendt datateknologi, mens en (Synne) går på Informasjonsteknologi.

Rolle	Navn	Epost
Prosjektleder, teknisk ansvarlig (frontend/ backend), fullstack utvikler	Martine Holmberg	s236808@oslomet.no
UX-leder, tillitsvalgt, rapportansvarlig, fullstack utvikler	Camilla H. Carlsen	s197235@oslomet.no
Teknisk ansvarlig (frontend/ backend), fullstack utvikler	Synne Angell	s331153@oslomet.no
Testleder, koordinator, fullstack utvikler	Nikola Sekiw	s333975@oslomet.no

Oppdragsgiver

“Zisson er et innovativt IT-selskap med kontorer i Oslo og Stockholm, som utvikler og drifter kontaktsenter- og sentralbordløsninger. Zisson ble etablert i 2007 med ambisjoner om å forenkle og forbedre kommunikasjonen mellom bedrifter og sluttkunder.” ([Zisson](#), 2020). Zisson Contact Center er en skybasert og omnikanal telefonsentral. De setter raskt opp fleksible kommunikasjon- og kundeserviceløsninger for henvendelser fra telefoni, webchat, e-post, chatbot, sosiale medier og SMS. De er partnere med Telenor, Techstep, Friday Networks og Mobit. Kundene deres er bedrifter, og sammen med partnerne deres har de over 400 kunder i offentlig og privat sektor.

Adresse: Zisson AS, Tollbugata 32, 0157 Oslo

Rolle	Navn	Tittel	Epost
Kontaktperson	Thomas Tretli	CTO	thomas.tretli@zisson.com
Teknisk leder	Moritz Lilleby	Utvikler	moritz.lilleby@zisson.no

Veileder fra OsloMet

Rolle	Navn	Tittel	Epost
Veileder	Aiko Yamashita	Førsteamanuensis hos OsloMet, Senior Data Scientist hos DNB	Aiko.Yamashita@dnb.no (Aiko.Yamashita@oslomet.no)

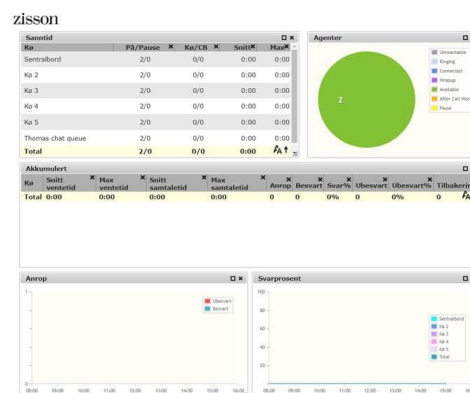
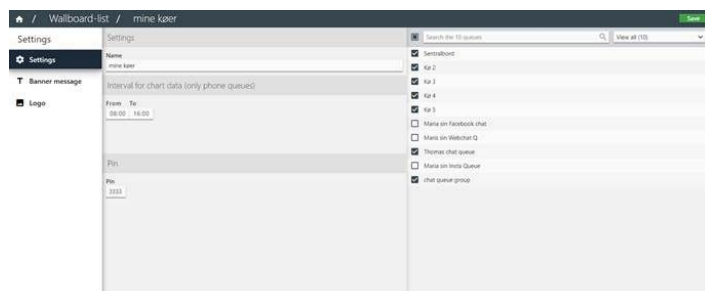
Sammendrag

Zisson ønsker å oppdatere dagens wallboard for å tilby kundene sine et bedre alternativ enn det som finnes. Systemet som skal lages skal ha et mer moderne design, samt teknologi, slik at det blir lettere for brukeren å administrere. Dette gjør det også lettere for utviklerne i Zisson å videreutvikle og vedlikeholde systemet.

Løsningen som skal utvikles baserer seg på en annen teknologi enn nåværende system, med C# på backend og Angular på frontend. Wallboardet skal lages etter en brukersentrert designprosess med testing av prototyper, hvor prototypen skal godkjennes før utviklingen av systemet skal starte. Dersom Zisson er fornøyd med systemet vil det kunne tas i bruk av Zisson sine kunder.

Dagens situasjon

Et wallboard viser informasjon om kundesenteret på TV-skjermer rundt kontoret, og kan gi en rask oversikt over hvor godt agentene jobber og hvilke køer som trenger mer oppmerksomhet. Wallboard viser vanligvis informasjon om hvordan de har klart seg i løpet av dagen i form av å vise informasjon om hvor mange som har svart, gjennomsnittlig taletid, gjennomsnittlig ventetid, maks ventetid, etc, samt også vise live visning av callsenterets agenter og køer. En kunde kan selv definere wallboard; hvilke køer som skal vises og dataintervallet.



Ovenfor: Eksempler fra gammelt wallboard.

I dag opplever Zisson wallboardet som utdatert både når det gjelder design og teknologi. Designet er opprinnelig fra 2010 og var laget som et testprosjekt. Bruken av frontend teknologi der agentene sitter er skrevet om med nyere teknologi som angular, og nå vil de gjøre det samme med wallboardet. Zisson ønsker at vi gjennom dette prosjektet skal løfte wallboardet til nye høyder ved bruk av nyere teknologier og redesign.

Mål og rammebetingelser

Målet med dette prosjektet er å lage et nytt og forbedret wallboard som skal være tilgjengelig på TV-skjermer rundt om på kontorene til kunder av Zisson som jobber på kontaktsenter-avdelinger. Applikasjonen skal være moderne og enkelt for en administrator å sette opp, samt være oversiktlig og intuitiv for brukerne av systemet.

Det skal være mulig å enkelt sette opp wallboardet med blant annet elementer som navn, agentgruppe, aktivitet, kø-oversikt, oppdrag, oppdragsgruppe, lokasjon på agent og samtalestatus, og mulighet for filtrering av ønskede elementer.

Teknologier/ verktøy

Valg av teknologi og verktøy er bestemt i samarbeid med Zisson da dette er noe de bruker i andre systemer.

Front-end

- Angular i TypeScript
- HTML, CSS og bootstrap

Back-end

- C# og .NET core

- Microsoft SQL server

Testing

- Verktøy: xUnit, TestCafe

Andre verktøy

- GitHub
- Google Drive/ Docs for dokumentasjon
- IDE - Visual studio code for frontend, visual studio for backend
- Jira - for fremstilling av brukerhistorier, backlog, sprintlog, code review
- Prototypeverktøy: Vil bli valgt senere i prosessen, hvor vi vet mer om hva slags type prototyper vi skal lage. Forslag til verktøy er: proto.io, adobe XD, Figma.
Hvis det ikke blir mulig å kunne møtes fysisk, vil det bli brukt verktøy for "brainstorming". Forslag til slike verktøy er : miro.com, atlassian.com
- Brukertestingsverktøy: Ved brukertester kommer vi mest sannsynlig til å utføre digitale intervjuer over Microsoft teams.

Kravspesifikasjon

Funksjonelle og ikke-funksjonelle krav

Nedenfor sees midlertidige krav vi har definert for oppgaven ut i fra den informasjonen vi har fått. Videre utdypning av kravene vil skje senere i prosessen i samråd med Zisson.

Funksjonelle krav:

- Vise oversikt over sentralbord med forskjellige elementer som er viktige for å gi informasjon om agentenes arbeid.

Ikke-funksjonelle krav:

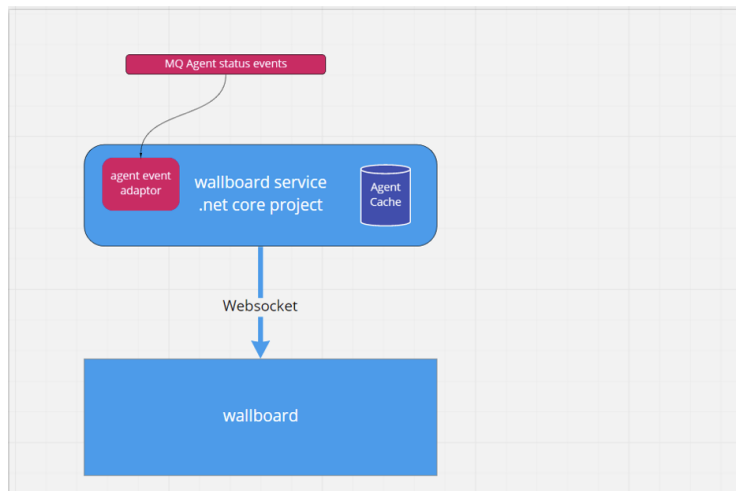
- Designe systemet med et helhetlig design ut i fra Zisson sitt interne system Wave
- Universell utforming
- Ha et fargebruk som samsvarer med det Zisson bruker i andre systemer.
- Krav om sanntid, så det skal tas i bruk cache.
- Enkelt å vedlikeholde
- Responsivt design
- Mulighet for å teste koden
- Brukervennlighet
- Pålitelighet

Systembeskrivelse

Systemet som skal lages er en oversikt, et wallboard, over hvordan agentene i et selskap arbeider.

Oppgaven inneholder to moduler:

- *Frontend*: her vil agentdata bli presentert
- *Backend*: her vil informasjon bli sendt til frontend



Hovedflyt - Endring av status til en agent:

1. Hendelser og statuser om en agent blir sendt fra en server.
2. Denne informasjonen lagres i en cache i wallboard-tjenesten.
3. Når en status til en agent blir endret må det dukke opp et varsel om dette på wallboardet.
4. Når en status til en agent endres skal hendelsen oversettes og utvides for å passe til informasjonen wallboardet trenger for å representere informasjonen om denne agenten.
5. Den nye hendelsen skyves deretter til WebSocket og wallboardet må oppdatere informasjonen og endringen om den gitte agenten.

Wallboardet bør initialiseres når en bruker går inn på det. Dette gjøres ved å koble til tjenesten med en WebSocket-forespørsel. Deretter skal en initialiserings-melding sendes for å kunne motta den nåværende tilstanden til agentene fra cachene på serveren.

Løsning/alternativer

Det er diskutert med Zisson om løsningen burde bli bygd opp fra bunnen av eller om vi skal jobbe med eksisterende løsning. Det er kommet fram til at den beste løsningen er å bygge frontend fra bunnen av, med ny arkitektur, der det bygges ut i fra en delvis eksisterende backendløsning. Hvor Zisson står for utvikling agent event adaptor og MQ agent status events.

Målet er å gå over til mer moderne teknologier som Angular (TypeScript), samt lage et mer moderne utseende. Siden det elementært er frontend som bør fornyes, kan backend-delen primært være som den er.

Fordeler:

- Bruk av Angular, en single page application, gjør at man er på samme html-side til enhver tid. Noe som gjør brukeropplevelsen responsiv. Da slipper man å kalle på server hver gang man skal gjøre noe. Trengs det noe fra server så kan det laste i bakgrunnen.
- C# er et effektivt språk å bruke da den har både ytelsen fra C++ og enkelheten til Java, samt at vi allerede er kjent med syntaksen.
- Zisson er godt kjent med Angular sitt rammeverk fra før. Dette er en fordel både med tanke på at hva Zisson kan hjelpe oss med, men også for Zisson sin del ved vedlikehold/videreutvikling av wallboardet.
- Kan være enklere å bygge opp ny frontend enn å sette seg inn i eksisterende kode
- Å bruke eksisterende backend da funksjonaliteten allerede er på plass der.

Ulemper:

- Krever mer forarbeid å sette opp en helt ny arkitektur i frontend.
- Kan bli en utfordring med tidsperspektivet, da det er mye som skal utvikles på kort tid.

Analyse av virkninger

Hvis løsningen vår er produksjonsklar og godkjent av Zisson, kan denne bli brukt videre i bedriften som en erstatning for den nåværende løsningen. Da trenger ikke Zisson å sette av et eget team som skal jobbe med dette på et senere tidspunkt.

Siden oppgaven er å fornye eksisterende wallboard vil også den nye løsningen kunne oppfattes som enklere og mer intuitiv å bruke. I tillegg kan den oppfattes som et bedre

alternativ visuelt og estetisk sett. Siden brukerne allerede er kjent med bruk av wallboard vil det i tillegg vil det være enklere for brukerne å kunne sette seg inn i det nye designet.

Det vil også bli enklere for Zisson å vedlikeholde applikasjonen da vi har tatt i bruk de samme teknologiene og verktøyene som de allerede er kjent med, samt god dokumentasjon av utviklingen.

Risikoanalyse

	1	2	3	4	5
Konsekvens					
Veldig alvorlig	5	10	15	20	25
Alvorlig	4	8	12	16	20
Moderat	3	6	9	12	15
Liten	2	4	6	8	10
Ubetydelig	1	2	3	4	5
	Veldig lav	Lav	Moderat	Høy	Veldig høy
	Sannsynlighet				

Hentet fra: <https://veratank.no/risikoanalyse/>

Vi har brukt modellen over for å fylle ut fargekode og nummer ut fra alvorlighetsgraden av konsekvensene.

Risiko er en sannsynlighet for at uønskede omstendigheter skjer. Under har vi vurdert risiko og mulig konsekvens i vårt prosjekt:

Nr	Hendelse	Sannsynlighet	Konsekvens	Risikoran gering	Forebyggende tiltak	Ansvarlig
1	Lite tid	Moderat	Alvorlig	12	Godt utarbeidet arbeidsplan og fremdriftsplan.	Utviklere
2	Kortvarig sykdom	Moderat	Liten	6	Standupmøter.	Prosjektleder, utviklere
3	Langvarig sykdom	Lav	Alvorlig	8	God fremdriftsoversikt. Standup-møter.	Prosjektleder, utvikler
4	Konflikt innad i gruppa	Lav	Moderat	6	God kommunikasjon, ta opp ting tidlig, evt. møter med en	Tillitsvalgt, utviklere

					utenforstående. Ta opp med tillitsvalgt.	
5	Teknologiske problemer	Moderat	Moderat	9	Ta kontakt med teknisk leder på Zisson og få hjelp.	Teknisk ansvarlig, utviklere
6	Tap av data	Lav	Veldig alvorlig	10	Lage backup underveis. Bruke Git.	Teknisk ansvarlig, utviklere
7	Kravendringer	Lav	Alvorlig	8	Smidig utvikling. Utarbeide krav sammen med Zisson. Sette av mer tid/bedre kommunikasjon underveis	Scrum master
8	Ekstra runder med brukertesting	Moderat	Liten	6	Sette av nok tid til designprosessen.	UX -leder.
9	Oppdragsgiver/ organisasjon går konkurs	Veldig lav	Veldig alvorlig	5		Zisson
10	Omfang av løsningen er mer krevende enn planlagt	Moderat	Alvorlig	12	Ikke gape over alt på en gang. Scrum sikrer at prosessen blir delt opp i konkrete mindre oppgaver.	Zisson, utviklere, veileder
11	Ny teknologi utviklerne ikke er kjent med	Veldig høy	Liten	10	Bruke tid på å sette oss inn i verktøy og teknologi.	Utviklere
12	Uenighet om kvaliteten på det ferdige produktet	Lav	Moderat	6	Alle er involvert i prosessen hele veien.	Zisson, utviklere

Fremdriftsplan

Gantt Chart

Aktivitet	Startuke	Varighet (uker)	4-10 Januar 1	11- 17 Januar 2	18-24 Januar 3	25-31 Januar 4	1-7 Februar (Sprint) 5	8-14 Februar (Sprint) 6	15-21 Februar (Sprint) 7	22-28 Februar (Sprint) 8	1-7 Mars (Sprint) 9
Planleggingsfase og skiving av forprosjekt:	1	4									
Rapportskriving	1	21									
Presentasjon	1	23									
Utarbeidelse av kravspesifikasjon	2	13									
Dagbok	2	15									
Oppsett av programmer	4	2									
Prototyping	5	3									
Utvikling	5	11									
Testing	5	11									
Ferdigstilling av sluttrapport	17	4,5									

et sprint review møte for å gå i gjennom og evaluere arbeidet som har blitt gjort i den foregående sprinten.

Hver planlagte arbeidsdag skal vi gjennomføre et standup-møte, hvor vi forteller hva vi jobber med, hvor langt vi har kommet og hva som er utfordrende med arbeidet. Dette for å vite hva de andre jobber med til enhver tid, slik at om sykdom skulle forekomme så er det enkelt å sette seg inn i arbeidsoppgavene raskt.

Scrum innebærer tre faser:

1. *Planlegging:* Her vil vi planlegge prosjektet og sette opp relevante verktøy vi trenger, samt ha planleggingsmøter og møter med veileder for å kunne sette i gang. I tillegg vil vi starte designprosessen med definering av kravspesifikasjon.
2. *Gjennomføring:* Her vil vi gjennomføre sprintene våre med utvalgte krav fra backlogen. Det vil bli utviklet et inkrement for hver sprint vi gjennomfører, slik at det blir enkelt å få tilbakemeldinger raskt underveis i utviklingen.
3. *Avslutning:* Her vil vi sette sammen dokumentasjonen laget underveis og ferdigstille sluttrapporten.

Vi kommer til å gjennomføre et testdrevet design. Dette innebærer at vi skriver enhetstester for alle funksjonene vi skal lage før vi begynner å kode disse. Det positive ved en slik tilnærming er at det kan gi bedre kvalitet på kode og minske antall bugs i produksjon.

Selve rapportskrivningen vil bli utviklet med en mer planmessig metode (waterfall), hvor vi kommer til å delegere oppgaver ut ifra tematikk.

Til designprosessen kommer vi til å bruke designmetoden design thinking, som innehar disse fem stegene:

1. Sette seg inn i brukerens behov.

Herunder skal vi snakke med produktansvarlig og komme i kontakt med brukere. Lage en undersøkelse for brukerne (hvis mulig). Utvikle user personas, og empathy map.

Under dette steget må vi svare på spørsmål som: Hvem er brukeren? Hva er problemet vi skal løse for dem? (spørre brukerne (hvis mulig) om hvilke problemer de møter i hverdagen). Evt. må vi prøve å finne ut av dette selv ved å bruke de ovennevnte metodene.

2. Definere problemet, dele inn i brukerhistorier.

Lage use cases, use case diagrammer, flow diagram.

3. Generere ideer til prototyper.

Workshop med free thinking, herunder bruke “yes, and”-fraser og ikke “can't” eller “no’s”, post it lapper, skissere forslag til prototyper, sette sammen forskjellige design boards.

4. Lage prototyper

Bli enig om hvilken prototype som passer best og utvikle denne med passende verktøy.

5. Teste prototypene

Presentere prototypen til produktavdeling. Få tilbakemeldinger. (så evt generere nye ideer til prototyper og lage nye prototyper eller starte utviklingen). Hvis mulig teste direkte på bruker.

Tentativ arbeidsplan

UKE	DATO	OPPGAVER
2	11-17.01	<u>Planleggingsfase</u> Arbeide med forprosjektet: <ul style="list-style-type: none">- Møte med Zisson 12.01- Skrive på forprosjektet- Lage arbeidsplan- Kickstart-møte med veileder 15.01
3	18-24.01	<u>Planleggingsfase</u> Arbeide med forprosjektet: <ul style="list-style-type: none">- Sende forprosjekt til veileder minimum to dager før 23.01- Leverer forprosjekt innen 23.1- Ferdigstille arbeidsplan- Møte med veileder og Zisson:<ul style="list-style-type: none">- Skrive under NDA- Spørre Zisson noterte spørsmål som har kommet opp under planleggingen Utarbeidelse av kravspesifikasjon: <ul style="list-style-type: none">- Avtale med Zisson møte for å definere brukerhistorier og krav sammen, samt finne ut av mulighetene for om vi kan komme i kontakt med noen brukere av systemet.

4	25-31.01	<p><u>Planleggingsfase</u></p> <p>Utarbeidelse av kravspesifikasjon:</p> <ul style="list-style-type: none"> - Datainnsamling: Sette oss inn i brukerens behov, få informasjon fra Zisson og evt. snakke med noen brukere - Lage brukerhistorier - Møte med Zisson for å utarbeide brukerhistorier sammen <p>Oppsett av programmer:</p> <ul style="list-style-type: none"> - Installere og sette opp all nødvendig programvare - Bli kjent med programvare vi er ukjent med - Få tilgang av Zisson til Git og andre ressurser/verktøy - Avtale med Zisson vedrørende Sprint planleggingsmøte <p>Arbeide med sluttrapport</p>
5	01-07.02	<p><u>Gjennomføringsfase (Sprint 1)</u></p> <p>Sprint planleggingsmøte</p> <p>Utarbeidelse av kravspesifikasjon:</p> <ul style="list-style-type: none"> - Lage brukerhistorier <p>Oppsett av programmer:</p> <ul style="list-style-type: none"> - Dersom noen har problemer eller mangler programvare skal dette ordnes denne uken - Sette oss inn i programvare vi er ukjent med <p>Prototyping:</p> <ul style="list-style-type: none"> - Workshop med free thinking - Skissere forslag til prototyper: lage low-fi prototyper - Bli enig om hvilken vi liker best, be om tilbakemelding fra Zisson - Lage high-fi prototype <p>Arbeide med sluttrapport</p>
6	08-14.02	<p><u>Gjennomføringsfase (Sprint 1)</u></p> <p>Prototyping:</p> <ul style="list-style-type: none"> - Lage high-fi prototype

		<ul style="list-style-type: none"> - Be om tilbakemelding fra Zisson - Brukertestet prototypen <p>Sprint review møte</p> <p>Arbeide med sluttrapport</p>
7	15-21.02	<p><u>Gjennomføringsfase (Sprint 2)</u></p> <p>Sprint planleggingsmøte:</p> <ul style="list-style-type: none"> - Velge ut oppgaver fra backlog <p>Internt møte:</p> <ul style="list-style-type: none"> - Delegere oppgavene som er valgt ut fra sprint backlog <p>Prototyping:</p> <ul style="list-style-type: none"> - Brukertestet prototypen - Gjøre endringer dersom det er behov - Ferdigstille prototypen <p>Arbeide med sluttrapport</p>
8	22-28.02	<p><u>Gjennomføringsfase (Sprint 2)</u></p> <p>Utvikling</p> <p>Testing:</p> <ul style="list-style-type: none"> - Lage enhetstester før koden produseres <p>Sprint review møte</p> <p>Arbeide med sluttrapport</p>
9	01-07.03	<p><u>Gjennomføringsfase (Sprint 3)</u></p> <p>Sprint planleggingsmøte:</p> <ul style="list-style-type: none"> - Velge ut oppgaver fra backlog <p>Internt møte:</p> <ul style="list-style-type: none"> - Delegere oppgavene som er valgt ut fra sprint backlog <p>Utvikling</p> <p>Testing:</p>

		<ul style="list-style-type: none"> - Lage enhetstester før koden produseres <p>Arbeide med sluttrapport</p>
10	08-14.03	<p><u>Gjennomføringsfase (Sprint 3)</u></p> <p>Utvikling</p> <p>Testing:</p> <ul style="list-style-type: none"> - Lage enhetstester før koden produseres <p>Sprint review møte</p> <p>Arbeide med sluttrapport</p>
11	15-21.03	<p><u>Gjennomføringsfase (Sprint 4)</u></p> <p>Sprint planleggingsmøte:</p> <ul style="list-style-type: none"> - Velge ut oppgaver fra backlog <p>Internt møte:</p> <ul style="list-style-type: none"> - Delegere oppgavene som er valgt ut fra sprint backlog <p>Utvikling</p> <p>Testing:</p> <ul style="list-style-type: none"> - Lage enhetstester før koden produseres <p>Arbeide med sluttrapport</p>
12	22-28.03	<p><u>Gjennomføringsfase (Sprint 4)</u></p> <p>Utvikling</p> <p>Testing:</p> <ul style="list-style-type: none"> - Lage enhetstester før koden produseres <p>Sprint review møte</p> <p>Arbeide med sluttrapport</p>

13	29.02-4.04	Påskeferie
14	05-11.04	<p><u>Gjennomføringsfase (Sprint 5)</u></p> <p>Sprint planleggingsmøte:</p> <ul style="list-style-type: none"> - Velge ut oppgaver fra backlog <p>Internt møte:</p> <ul style="list-style-type: none"> - Delegere oppgavene som er valgt ut fra sprint backlog <p>Utvikling</p> <p>Testing:</p> <ul style="list-style-type: none"> - Lage enhetstester før koden produseres <p>Arbeide med sluttrapport</p>
15	12-18.04	<p><u>Gjennomføringsfase (Sprint 5)</u></p> <p>Utvikling</p> <p>Testing:</p> <ul style="list-style-type: none"> - Fullstendig test av systemet <p>Sprint review møte</p> <p>Arbeide med sluttrapport</p>
16	19-25.04	<p><u>Gjennomføringsfase (Sprint 6)</u></p> <p>Sprint planleggingsmøte:</p> <ul style="list-style-type: none"> - Velge ut oppgaver fra backlog <p>Internt møte:</p> <ul style="list-style-type: none"> - Delegere oppgavene som er valgt ut fra sprint backlog <p>Utvikling:</p> <ul style="list-style-type: none"> - Rette opp i feil oppdaget ved testing? <p>Testing</p> <p>Sprint review møte</p> <p>Arbeide med sluttrapport</p>

17	26.04-02.05	<u>Avslutningsfase</u> Sluttrapport: <ul style="list-style-type: none"> - Internt møte: Delegere arbeid for sluttrapporten
18	03-09.05	<u>Avslutningsfase</u> Sluttrapport: <ul style="list-style-type: none"> - Skrive på sluttrapport
19	10-16.05	<u>Avslutningsfase</u> Sluttrapport: <ul style="list-style-type: none"> - Skrive på sluttrapport
20	17-23.05	<u>Avslutningsfase</u> Sluttrapport: <ul style="list-style-type: none"> - Gjennomgå sluttrapporten - Sende sluttrapport til veileder og Zisson for tilbakemeldinger
21	24-30.05	<u>Avslutningsfase</u> Sluttrapport: Innlevering 25.mai <ul style="list-style-type: none"> - Begynne på presentasjon
22	31.05-6.06	<ul style="list-style-type: none"> - Jobbe med presentasjon
23	07-10.06	<ul style="list-style-type: none"> - Jobbe med presentasjon - Presentasjon