

BACHELOROPPGAVE:



Gitek Register

FORFATTERE:

Martin Trehjørningen Harestad
Håvard Myrvold Johannessen
Vetle Nikolai Løland Lundsvoll
Martin Stenen

Dato: 16.05.2018

SAMMENDRAG

Tittel:	Gitek Register	Dato :	16.05.2018
Deltaker(e)/	Martin Trehjørningen Harestad		
	Håvard Myrvold Johannessen		
	Vetle Nikolai Løland Lundsvoll		
	Martin Stenen		
Veileder(e):	Frode Haug		
Evt. oppdragsgiver:	Khai Van Ngo ved Gitek AS.		
Stikkord/nøkk elord (3-5 stk)	Gitek, Integret, Web, JavaScript, HTML		
Antall sider: 105	Antall vedlegg: 6	Publiseringsavtale inngått: ja	
Kort beskrivelse av bacheloroppgaven:			
<p>Gitek Register går ut på å lage en modernisert implementasjon av et eldre system brukt av Det norske Skogfrøverk. Den nye løsningen skal være en webløsning som håndterer flere aspekter ved registrering og bestilling av frø på ulike treslag fra innland og utland. Systemet skal effektivisere hele prosessen med å sanke, teste, analysere, registrere og selge frø til kjøpere fra hele verden.</p> <p>Løsningen er fullt integrert inn i Gitek sine systemer og kjører på en av deres servere. Gitek Register er utviklet med HTML, CSS, JavaScript og jQuery.</p>			

ABSTRACT

Title:	Gitek Register	Dato :	16.05.2018		
<hr/> <hr/> <hr/>					
Participants/	Martin Trehjørningen Harestad				
	Håvard Myrvold Johannessen				
	Vetle Nikolai Løland Lundsvoll				
	Martin Stenen				
Supervisor(s)	Forde Haug				
Employer:	Khai Van Ngo with Gitek AS				
Keywords (3-5)	Gitek, Integrated, Web, JavaScript, HTML				
Number of pages/words:	105	Number of appendix:	6	Availability:	open
Short description of the bachelor thesis:					
<p>Gitek Register involves making a modernized implementation of an older system used by Det norske Skogfrøverk. The new solution is a web-based solution that handles multiple aspects of registering and placing orders of seeds from different tree species from in and out of country. The system improves the efficiency of the entire process of gathering, testing, analyzing, registering and selling seeds to buyers across the globe.</p> <p>The solution is fully integrated in Gitek's systems and is running on one of their servers. Gitek Register is developed using HTML, CSS, JavaScript and jQuery.</p>					

Forord

Gruppen ønsker å takke Gitek og deres ansatte for å ha deltatt aktivt i prosjektet både som project owner og veiledere, samt for deres tekniske assistanse.

Gruppen takker også gruppens veileder, Frode Haug, for hans assistanse gjennom hele prosjektet, samt alle ansatte ved NTNU som delte sin kunnskap med gruppen og hjalp oss forbedre prosjektet.

Forord	3
Figur liste	7
Tabell liste	8
1. Introduksjon	9
1.1 Bakgrunn	9
1.2 Beskrivelse	9
1.3 Avgrensning	11
1.3.1 Prosjektavgrensning	11
1.3.2 Teknisk avgrensning	11
1.4 Målgruppe	11
1.5 Kompetanse	12
1.6 Formål	12
1.6.1 Effektmål	13
1.6.2 Resultatmål	13
1.6.3 Læringsmål	13
1.7 Roller	13
1.8 Rapporten	15
1.8.1 Terminologi	15
1.8.2 Organisering	16
1.8.3 Praktisk	17
2. Kravspesifikasjon	18
2.1 Kravspesifikasjon	18
2.2 Personas	20
2.2.1 Use cases	24
2.2.2 User Stories	31
3. Produktplanlegging	34
3.1 Modell	34
3.1.1 Utviklingsmetode	34
3.2 Taskboard	36
3.3 Prototyper	37
4. Design og Arkitektur	47
4.1 Arkitektur og mønstre	47
4.1.1 Lagdelt arkitektur	47
4.1.2 Repository struktur	48

4.1.3	Flatt webløsning hierarki	51
4.2	Oppdragsgivers Rammeverk	53
4.3	Brukergrensesnitt	53
4.3.1	Responsivt design	54
4.3.1.1	Forskjellige plattformer	56
4.3.2	Forms	59
4.3.3	Navigasjon og fremside	60
4.3.4	Login	62
4.3.5	CRM-modulen	63
4.3.6	Produktregister-modulen	69
4.3.7	Analyse-modulen	73
4.3.8	Rapport-modulen	75
4.3.9	Ordre-modulen.	77
5.	Implementasjon og koding	81
5.1	Utviklingsmiljø	81
5.1.1	Server	81
5.1.2	IDE	82
5.1.3	Repository	82
5.1.3.1	SourceTree og GitFlow	82
5.2	Web-løsning	84
5.2.1	HTML5	84
5.2.2	Javascript og jQuery	84
5.2.3	Tredjeparts-biblioteker	84
5.2.3.1	Bootstrap	84
5.2.3.2	DataTables	85
5.2.3.3	HighCharts	86
5.2.3.4	Font Awesome	89
5.3	Oppdragsgivers Biblioteker	90
5.3.1	Database	91
5.3.2	PDF	91
5.3.3	Dynamisk lasting	93
6.	Testing	94
7.	Avslutning	96
7.1	Diskusjon og drøfting	96
7.1.1	Integrasjon	97
7.2	Resultater	99
7.2.1.1	Effektmål	99
7.2.1.2	Resultatmål	99

7.2.1.3 Læringsmål	100
7.3 Videre utvikling og arbeid	101
7.4 Evaluering av gruppens arbeid	101
7.4.1 Innledning	101
7.4.2 Organisering	101
7.4.2.1 Kommunikasjonskanaler	101
7.4.2.2 Møter	102
7.4.3 Fordeling av arbeid	103
7.4.4 Prosjekt som arbeidsform	103
7.5 Konklusjon	104
Kilder	105
Vedlegg	106
Vedlegg A - Planer	106
A.1 - Gantt diagram i fra prosjektplanen	106
A.2 - Gantt diagram i fra slutten av prosjektet	107
Vedlegg B - Prototyper	108
Vedlegg C - Kontrakter	117
Vedlegg D - Rapporter og referater	121
D.1 - Møttereferater	121
D.1.1 Møttereferater i fra møter med Gitek	121
D.1.2 Møttereferater i fra møter med veileder	134
D.1.3 Møttereferater i fra gruppens interne møter	141
D.1.4 Møttereferater i fra møter med ansatte hos NTNU	149
Vedlegg E - Time logg	152
Vedlegg G - Prosjektplanen	155

Figur liste

1. Flyten av data i mellom modulene	19
2. CRM use case	25
3. Produktregistrerings Use case	26
4. Ordre use case	27
5. Analyse/Rapport use case	28
6. vårt YouTrack board	37
7. Prototype - forside pc	38
8. Prototype - forside tablet	38
9. Prototype - valg i header	40
10. prototype - generell layout	41
11. Prototype - tabellfremvisning	42
12. Prototype - endring av data i tabell	43
13. Prototype - min side	44
14. Prototype - pagination	45
15. Prototype - fremvisning av produkter	46
16. lagdelt arkitektur	48
17. Repository struktur 1 - fra prosjektstart	49
18. Repository struktur 2: fra 21.03.2018	50
19. Flatt vs dypt hierarki	51
20. Prosjektets kategorier	52
21. Ajax kall	53
22. Bootstrap gridsystemet	54
23. Eksempel tablet portrait	55
24. Eksempel tablet landscape	56
25. Eksempel mobil portrait	57
26. Eksempel mobile landscape	58
27. Fremside - pc	59
28. Eksempel på affordance	60
29. Login siden	61
30. Brukeradministrasjon	62
31. Detaljert brukerinformasjon	63
32. Visning av medlemmene i en organisasjon	64
33. Brukerprofil	64
34. Rediger profil	65
35. Organisasjons Tabell	67
36. Organisasjonsinfo	68
37. Signifiers for åpning og lukking av faner	68
38. Bilde fra registrerings formen for Frøpartier	70
39. Sliders for talverdier	72
40. Sliders for talverdier 2	73

41. valg av analyse	74
42. Analysekort	74
43. valg av dato	75
44. Rapport illustrasjoner	75
45. Rapport illustrasjoner 2	76
46. Frøsertifikat	78
47. frontend for ordre modulen	79
48. frontend for ordre modulen 2	80
49. SourceTree branch tracker	83
50. DataTable kode eksempel	86
51. Sektordiagram som viser prosentvis totalmengden av hvert treslag i kilogram	87
52. HighChart kode eksempel	88
53. HighChart kode eksempel 2	89
54. Bruk av Font Awesome i HTML	89
55. Skript i fra oppdragsgivers portefølje legges til	90
56. Ajax kall for å hente data i fra databasen (tatt ut av sammenheng)	91
57. Funksjon gjør HTML om til PDF	92
58. Dynamisk lasting av HTML side	93

Tabell liste

1. Bootstrap responsivitet	54
----------------------------	----

1. Introduksjon

I dagens IT samfunn er det viktig for bedrifter å anskaffe seg ny teknologi og nye systemer som gir dem et fortrinn overfor deres konkurrenter. Samtidig har skyteknologi blitt mer og mer fremtredende. Løsninger lages og opprettholdes i skyen til fordel for direkte på maskinen til sluttbrukeren. Mange bedrifter har allerede systemer på plass som er integrert inn i organisasjonen. Mange av disse løsningene begynner å bli svært gamle, og kjører kun på utdaterte systemer. Disse er ofte utfordrende å vedlikeholde og kostnaden av å videreutvikle systemet er større enn gevinsten. I slike situasjoner, er det ofte nødvendig å gjøre noen endringer for å holde seg konkurransedyktig. Dermed kan organisasjonen velge mellom å skaffe seg et helt nytt system som løser de samme problemene som tidligere. Et annet alternativ er å lage et lignende system som det gamle, med ny teknologi. Vårt prosjekt håndterer det andre valget.

1.1 Bakgrunn

Stiftelsen *Det norske Skogfrøverket*, heretter kalt kunden, ble opprettet i 1996. Kunden har ansvar for å administrere landets frøforsyning, samling av frø, og distribuering av alle sorter frø til planteskoler og skogeiere. For å holde oversikt over daglig drift, som for eksempel salg og varebeholdning, har kunden benyttet seg av en tidligere utviklet programvare for å kunne administrere de aller fleste av sine arbeidsoppgaver [1].

Denne løsningen ønsker nå kunden å modernisere ved å overføre den eksisterende funksjonalitet i programvaren over i en ny og mer dagsaktuell webløsning. I denne anledning har kunden vært i dialog med Gitek, som har fått i oppdrag å levere ny programvare etter kundens ønske. Vår studentgruppe har gjennom oppdragsgiver fått gitt muligheten til å utvikle en løsning direkte integrert inn i Gitek's systemer som bacheloroppgave. Funksjonalitet fra den eksisterende applikasjonen må bli erstattet og implementert i den nye webløsningen [1].

1.2 Beskrivelse

Gruppen har fått i oppgave å lage en webapplikasjon for oppdragsgiver. Kunden har tidligere hatt en løsning lagd i Paradox som er utdatert, og er vanskelig å navigere seg gjennom uten mye erfaring med systemet. Gruppen er av den oppfatning at Paradox applikasjonen viste for mye informasjon på skjermen. Samtidig var navigasjon mellom de forskjellige modulene forvirrende og krevde i noen tilfeller svært mange tastetrykk.

Oppdragsgiver valgte å sette opp deres nye løsning som en webapplikasjon. Siden gruppen skulle lage en applikasjon tilnærmet oppdragsgivers løsning, ble det som et mål for den nye løsningen å forbedre de aspektene gruppen anså som dårlige i Paradox løsningen.

Gruppens løsning har tre brukergrupper:

- *Eksterne brukere*: Enkeltmannsforetak og organisasjoner som kjøper og selger frø gjennom kunden. Planteskoler, i løsningen, er en variant av organisasjoner.
- *Interne brukere*: Kundens ansatte som organiserer, lagrer, registrerer og analyserer frøene samt fakturerer ved salg av frø.
- *Administrator*: Oppdragsgiver vil administrere løsningen og brukerbasen. I praksis samme privilegier som interne brukere.

Gjennom utviklingen av webløsningen hadde gruppen et fokus på brukergrensesnitt og user experience og forsøkte å oppnå en god brukeropplevelse ved å undersøke og dra inspirasjon fra design patterns og prinsipper fra Googles Material Design og Apples Human Interface Guidelines. Gruppen ikke også beskjed av oppdragsgiver at den nye løsningen måtte inneholde gjenkjennbare elementer for de eksisterende brukerne. Informasjon ble presentert for brukeren i den gamle løsningen måtte også vises i den nye.

Oppdragsgiver hadde utarbeidet en liste over funksjonelle krav det nye systemet skulle møte. Disse kravene utgjorde kjernefunksjonaliteten som er fundamental i oppgaven. For oversiktens skyld har de gjort valgt å kategorisere disse i moduler. Samtidig gjorde de oss oppmerksomme på at disse modulene var laget for å vise kunden hva som ville inngå i løsningen, og det var ikke tatt hensyn til arbeidsmengden som vil inngå i modulene. Gruppen valgte derfor å lage vår egen versjon av listen over modulene, som gruppen mente ville være en mer realistisk gruppering av relaterte oppgaver, og fordelte arbeidet bedre. Dette ble godt mottatt av oppdragsgiver, og endte opp med å bli endelig versjon.

- CRM-modul
 - Registrering, pålogging og endring av personlig informasjon for brukere
 - Registrere og vedlikeholde planteskoler, og deres kunder
 - Visning av alle brukere og organisasjoner
- Produktregister-modul
 - Registrere og endre informasjon som tilhører produkter.
 - Opprette, importere og støtte nedlastning av sertifikater for frøpartier
- Analyse-modul
 - Registrere og vedlikeholde analyser på produkter
 - Lage analysekort
- Ordre-modul
 - Beregne pris og tilgjengelighet på produkter for kunde
 - Registrere og vedlikeholde rapportering på frøsetting
 - Web-shop som støtter
 - Automatisk genererte bestillinger
 - Manuelle bestillinger
 - Legge inn ordre på vegne for kunder
 - Lage rapporter for bestilte varer med varens tilstand
 - Reservasjon av produkter

- Digitalt fakturasystem med epost
 - Legge inn og behandle kreditnota
- Rapport-modul
 - Frø på lager
 - Sankeområder
 - Aktuelle frøparti
 - Blandet frøparti fra flere lokaliteter
 - Viser lagerrapport for alle/spesifikke treslag
 - Sankerapport
 - Prissammenligning fra tidligere år

1.3 Avgrensning

For å sette rammer rundt hva slags applikasjon gruppen skulle lage, ble det bestemt avgrensninger i samarbeid med oppdragsgiver.

1.3.1 Prosjektavgrensning

Løsningen er en enterprise-applikasjon, ment for bruk i jobbsammenheng som en del av porteføljen til kunden. All funksjonalitet og brukergrensesnitt som ikke passer ville derfor ikke bli implementert. Det ble derfor ikke vurdert noen tiltak for å tiltrekke nye brukere.

Siden oppdraget gikk ut på å modernisere kundens eksisterende programvare, valgte gruppen å avgrense funksjonaliteten til vår applikasjon til matche, forenkle, og forbedre den eksisterende funksjonaliteten. Annen funksjonalitet vil kun være det gruppen anser som forbedring av den gamle løsningen. Skulle oppdragsgiver ønske ny funksjonalitet ville dette fått lavere prioritet, og bli implementert etter ferdigstilt grunnfunksjonalitet.

1.3.2 Teknisk avgrensning

Oppdragsgiver ønsker, som tidligere nevnt, en webapplikasjon som den nye løsningen. I tillegg har gruppen valgt å integrere webapplikasjonen inn i oppdragsgivers systemer. Dette medførte noen avgrensninger i seg selv. Det kunne ikke brukes PHP, da dette programmeringsspråket ikke er støttet i av oppdragsgivers backend. Gruppen skulle heller ikke lage noe av backenden til systemet, da denne allerede eksisterer i oppdragsgivers system. På grunn av dette valgte gruppen å avgrense programmeringsspråkene til HTML, CSS og JavaScript (og derunder jQuery og AJAX) for å sørge for full kompatibilitet med oppdragsgivers systemer.

1.4 Målgruppe

Målgruppen for rapporten og prosjektet som en helhet er de som ønsker innsikt i prosjektet og den tilhørende rapporten. Eksempler er sensor, ansatte ved NTNU, andre studenter, og oppdragsgiver.

Applikasjonens målgrupper er basert på de tre brukergruppene. Dette kom som et resultat av at løsningen er av typen *enterprise*.

1. *Eksterne brukere*: Denne målgruppen består av alle kundene til *Det Norske Skogfrøverket*, og er derfor den største brukergruppen. Disse brukerne har ikke nødvendigvis kjennskap til hvordan systemet fungerer. Det er derfor viktig at løsningen er enkel å bruke, spesielt nettbutikken og ordrefunksjonaliteten. Denne målgruppen er også veldig viktig siden den har betydning for inntekten til kunden.
2. *Interne brukere*: Den nest største brukergruppen. Denne målgruppen består av de ansatte hos *Det Norske Skogfrøverket*. Denne gruppen trenger derfor tilgang til verktøy og data som de eksterne brukerne ikke trenger tilgang til. Eksempler er rapporter og oversikter over organisasjonene.
3. *Administrator*: De ansatte hos Gitek. Deres oppgaver er å redigere data, for eksempel data i databasen som bruker informasjon, og å drifte systemet. Følgelig har denne målgruppen samme privilegier og tilgang som *interne brukere*.

1.5 Kompetanse

Gruppen hadde allerede en del erfaring når det kommer til bruken av web-baserte programmeringsspråk. Spesielt HTML, JavaScript, jQuery, PHP, og grunnleggende CSS. I tillegg til dette har gruppe medlemmene også tilegnet seg kunnskaper og erfaring med bruk av databaser (SQL og noSQL), systemutvikling, samt programmering av forskjellige typer applikasjoner i forskjellige programmeringsspråk.

Det gruppen måtte tilegne seg av kompetanse var hovedsakelig hvordan de interne Gitek-modulene fungerte og hvordan de skulle brukes. Det ble også behov for å lære hvordan å benytte oppdragsgivers servermiljø. Gruppen måtte også fordype seg i jQuery siden det meste av programmeringen endte opp med å bli gjort i dette språket. Det innebar at gruppen også måtte sette seg inn i mer ukjente biblioteker, som gruppen ikke har benyttet seg av for å løse oppgaver relatert til de forskjellige modulene.

1.6 Formål

Oppdragsgiveren startet utviklingen av den faktiske løsningen for kunden parallelt med gruppen. Gruppen hadde kortere tidsfrister, og avsluttet prosjektet før oppdragsgivers ferdigstilte produkt. Dermed fikk oppdragsgiver mulighet til å se gruppens løsningsforslag, og observere muligheter og utfordringer. Oppdragsgiver ønsker også å bruke bachelor prosessen som en mulighet for dem å tilegne seg ledererfaring, samt at de kan få et innblikk i hvilke verktøy og metoder som brukes av studentene. Siden gruppen valgte å integrere løsningen inn i deres systemer, ble dette systemet stresstestet av utviklere uten tidligere erfaring med rammeverket for mulige utfordringer relatert til kompatibilitet.

Gruppens formål med bachelorprosjektet var å tilegne seg erfaring med å jobbe med bedrifter gjennom prosjektet, samt å samarbeide med hverandre og en product owner i et større prosjekt. Videre ønsker gruppen mer erfaring med å jobbe tett opp mot ukjente systemer og biblioteker.

1.6.1 Effektmål

- Forenkle brukergrensesnittet og brukeropplevelsen ved å benytte seg av kjente patterns og prinsipper.
- Skape en applikasjon som løser de samme problemene som den gamle løsningen.
- Lette arbeidsflyten til ansatte hos kunden.
- Utvikle et program med høy brukervennlighet, slik at det blir enklere for brukerne av programmet å utføre daglige arbeidsoppgaver.
- Lett å vedlikeholde/administrere ved å flytte fra gammel databasestruktur
- Oversiktlig kodebase, som tillater enkel ekspansjon og videreutvikling.

1.6.2 Resultatmål

- Utvikle programvaren med direkte integrasjon inn i oppdragsgivers database/server struktur.
- Levere en brukervennlig og intuitiv løsning.
- Overføre eksisterende funksjonalitet inn i web-løsningen.
- Levere programvare med forskjellige rettigheter for ulike brukergrupper.
- Utvikle en webløsning optimalisert for bruk på desktop og nettbrett.
- Levere en bacheloroppgave som møter de oppsatte kriterier.

1.6.3 Læringsmål

- Tilegne seg nye kunnskaper på programmeringsspråk, programvare og teknologi som skal anvendes i oppgaven.
- Effektiv kommunikasjon og koordinasjon.
- Få erfaring innen flere rammeverk og API for webløsninger.
- Bli bedre kjent med prinsipper og patterns som øker brukeropplevelse
- Tilegne seg erfaring i utvikling med et mellomledd mellom utviklere og sluttbrukere.
- Få erfaring ved å jobbe med en reell jobbsituasjon hvor gruppen utvikler et tenkt produkt til sluttbrukere, nærmere enn studiets proof of concepts.

1.7 Roller

Gjennom hele prosjektet har gruppen, og eksterne aktører, blitt tildelt roller som definerer hva de skal gjøre. Disse rollene er definert under.

Gruppeleder(Vetle):

- Gruppelederen har ansvar for å styre møter og ha oversikt over agendaer for møtene, både interne og eksterne.

- Gruppeleder vil videre brukes som kommunikasjonskanal mellom gruppen og eksterne aktører for eksempel oppdragsgiver og veileder.
- Det er derfor naturlig og ha rollene gruppeleder og scrum-master som en og samme rolle.

Referent(Martin T.)

- Skal skrive referater fra gjennomførte møter

Rapportansvarlig(Martin S.)

- Skal ha oversikt over dokumenter som må skrives i sammenheng med bacheloroppgaven og den endelige rapporten, samt å kommunisere dette ut mot gruppa.

Design- og kodekoordinator(Håvard)

- Har ansvar for å koordinere design- og kodevalg slik at disse imøtekommer gruppens målsetninger, og arbeidsgivers ønsker og krav rundt design og kode

Oppdragsgiver (Gitek, Khai Van Ngo)

- Oppdragsgivers roller er å gi deres mening om hvordan gruppen presisere og kvaliteten på arbeidet gruppen leverer.
- Oppdragsgiver stiller også med egne moduler som gruppen skal bruke for å utvikle applikasjonen. Oppdragsgiver har derfor sagt seg villig til å stille med teknisk hjelp til bruk av disse modulene.

Veileder (Frode Haug)

- Veileders rolle i prosjektet er å lede gruppen og gi gruppen råd slik at gruppen skal kunne gjøre en best mulig jobb på prosjektet.
- Veileder skal ikke hjelpe gruppen med å programmere løsningen eller å skrive dokumenter.

1.8 Rapporten

De følgende underkapitlene tar for seg informasjon som leseren av rapporten kan trenge i løpet av de opp kommende kapitlene, samt praktisk informasjon om rapporten.

1.8.1 Terminologi

Denne ordlisten inneholder ord som brukes av gruppen for å beskrive en spesifikk ting, enhet, eller person.

Ord brukt av gruppen	Forklaring
Oppdragsgiver	Gitek AS
DNS / kunden	Det Norske Skogfrøverket

Denne ordlisten tar for seg tekniske ord og uttrykk som gruppen bruker i løpet av rapporten og gir dem en kort forklaring.

Ord brukt av gruppen	Forklaring
Internet Information Service (IIS)	IIS er en Microsoft web-server ment for å kjøre web relaterte tjenester [5].
Paradox	Gammelt databasesystem brukt mye på 80- og starten av 90-tallet ¹
ObjectPAL	<i>Object-Oriented Paradox Application Language</i> . Programmeringsspråk for Paradox ²
Epics	<i>"An epic is a large body of work that can be broken down into a number of smaller <u>stories</u>, or sometimes called Issues..."</i> ³

¹ "Paradox (database) - Wikipedia." [https://en.wikipedia.org/wiki/Paradox_\(database\)](https://en.wikipedia.org/wiki/Paradox_(database)). Åpnet 13 mai. 2018.

² "ObjectPAL - Wikipedia." <https://en.wikipedia.org/wiki/ObjectPAL>. Åpnet 13 mai. 2018.

³ "Epics | Atlassian." <https://www.atlassian.com/agile/project-management/epics>. Åpnet 15 mai. 2018.

1.8.2 Organisering

Rapporten består av de følgende kapitlene:

1. **Introduksjon**

Det første kapitlet er introduksjonen til oppgaven, ment for å gi leseren en grunnleggende forståelse for oppgaven.

2. **Kravspesifikasjon**

Kapitlet handler om prosjektets kravspesifikasjon, samt de artefaktene og prosessen gruppen har vært gjennom for å utforme denne kravspesifikasjonen.

3. **Produktplanlegging**

Produktplanleggingskapitlet er ment for å gi leseren bedre innsikt i hvordan gruppen styrte prosjektet.

4. **Design**

Kapitlet handler om prosjektets grafiske og arkitektoniske valg og implementasjon.

5. **Implementasjon og koding**

Ment for å gi leseren en forståelse for hva gruppen har gjort, samt en bedre forståelse for valgene gruppen tok rundt implementasjonen av applikasjonen og dens kode.

6. **Testing**

Viser til og begrunner valgene gruppen har tatt vedrørende testing av applikasjonen.

7. **Konklusjon**

Kapitlet dekker gruppens erfaring med å jobbe på prosjektet, samt en sammenligning av hvordan prosjektet gikk satt opp i mot gruppens originale planer for prosjektet.

8. **Kilder**

Dette kapitlet inneholder listen over alle kildene brukt i rapporten.

9. **Vedlegg**

Alle vedleggene som gruppen mener er relevante for rapporten/prosjektet.

1.8.3 Praktisk

Rapporten er skrevet på norsk (bokmål), unntak av dette er diverse fagord/program spesifikke ord som ikke har noen eller en god nok oversettelse på norsk.

En av avtalene gruppen inngikk med Gitek før produksjonen av oppgaven kunne starte var at kundenes personlige informasjon, diverse deler av Gitek stacken og at dokumenter gruppen har fått av Gitek ikke kunne deles. På grunn av dette har gruppen endret alle navn og e-postadresser som vises i rapporten til tilfeldige navn som gruppen har kommet på selv. Gruppen har derfor også i noen kapitler måtte skrive rundt diverse temaer relatert til stacken ved at gruppen ikke deler hvordan forskjellige systemer fungerer i dybden.

I løpet av teksten kommer vi til å referere til kilder på to måter. For litterære- eller større web kilder kommer gruppen til å referere til dem ved hjelp av Vancouver stilen, med [kilde nummer] notasjoner i teksten og kildeliste i kapittelet "Kilder". Mindre kilder blir referert til fortløpende i teksten med pekere til sidens fotnote.

2. Kravspesifikasjon

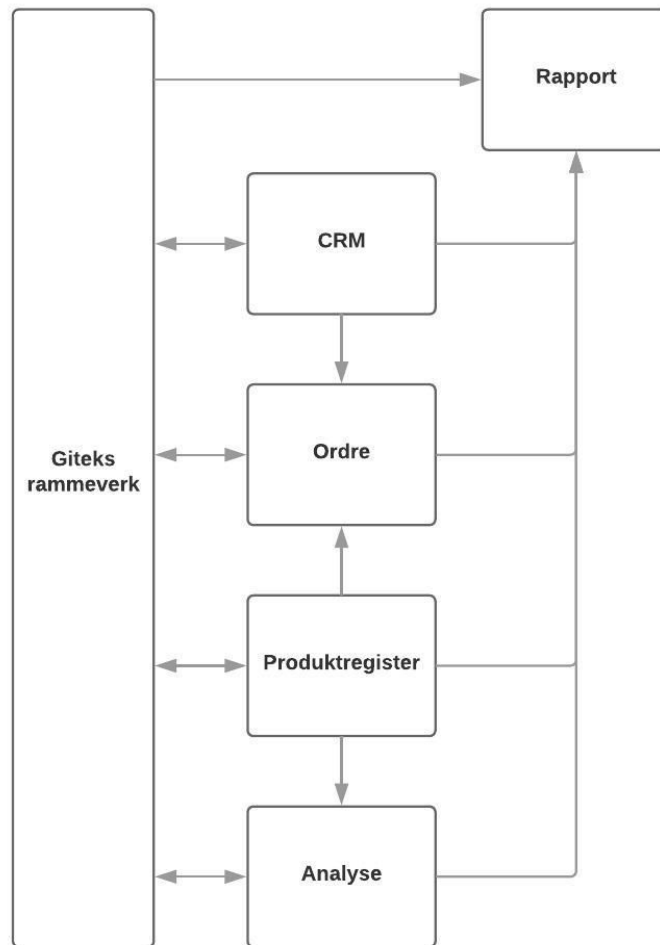
For å organisere arbeidet og ha en klar idé om hva sluttproduktet kom til å bli, utformet gruppen en kravspesifikasjon med overordnede punkter om hva som skulle inn i det endelige produktet. Disse kravene fordeles og stykkes opp i mindre oppgaver som gruppen håndterer i sprinter. Å utforme kravspesifikasjoner er noe gruppen hadde erfaring med fra tidligere i studiet, men det er noen omstendigheter rundt oppgaven som har forandret hvilke metoder som ble benyttet.

Den gamle løsningen er basert på en applikasjon utviklet i ObjectPAL med en Paradox database i backend. Den gamle applikasjonen bærer store preg av at det har vært utviklet "midlertidige" løsninger for problemer, ettersom behovene har dukket opp. Som resultat har hovedmenyen mange valg, hvor flere av dem leder til samme funksjonalitet. Strukturen innover i applikasjonen er forvirrende, med skjemaer som går over flere sider og navigasjon som går i "loop".

2.1 Kravspesifikasjon

Som nevnt tidligere er gruppens kravspesifikasjon basert på de funksjonelle kravene til Gitek. Modulene i de funksjonelle kravene var delt opp med tanke på å presentere den konseptuelle idèen av løsningen til sluttbrukeren og måtte omstruktureres for å benyttes som en kravspesifikasjon. Gruppen har ikke fått tillatelse fra Gitek til å ha med i rapporten de funksjonelle kravene vi fikk utdelt. De funksjonelle kravene fra Gitek består av mange av de samme punktene som den endelige kravspesifikasjonen. Gruppen har endret på hvilke oppgaver som går inn i de forskjellige modulene og endret på enkelte punkter for å samsvare med en utviklers perspektiv. Med tanke på at løsningen skulle utvikles modulbasert ble det valgt å samle funksjonalitet som var like tematisk sett, i samme modul. Punkter i de funksjonelle kravene ble også samlet inn til å skape større epics så gruppen senere kunne dele disse opp til mindre stories og senere task.

For å kartlegge hvor det er avhengigheter på tvers av modulene, har gruppen satt modulene i et diagram. Pilene viser dataflyten fra modul til modul.



Figur 1: Flyten av data i mellom modulene

Ved å skape et klassediagram hvor gruppen fokuserte på hvilke data modulene ville benytte seg av, var det lett å se i hvilken rekkefølgen modulene burde utvikles. CRM og Produktregister henter data fra Giteks rammeverk og kan jobbes med uten annen implementasjon. Analyse krever data fra Produktregister, Ordre krever data fra både CRM og Produktregister, og Rapport krever data fra alle modulene.

Den endelige kravspesifikasjonen ble:

- CRM-modul
 - Registrering, pålogging og endring av personlig informasjon for brukere
 - Registrere og vedlikeholde planteskoler, og deres kunder
 - Visning av alle brukere og organisasjoner
- Produktregister-modul
 - Registrere og endre informasjon som tilhører produkter.

- Opprette, importere og støtte nedlastning av sertifikater for frøpartier
- Analyse-modul
 - Registrere og vedlikeholde analyser på produkter
 - Lage analysekort
- Ordre-modul
 - Beregne pris og tilgjengelighet på produkter for kunde
 - Registrere og vedlikeholde rapportering på frøsetting
 - Web-shop som støtter
 - Automatisk genererte bestillinger
 - Manuelle bestillinger
 - Legge inn ordre på vegne for kunder
 - Lage rapporter for bestilte varer med varens tilstand
 - Reservasjon av produkter
 - Digitalt fakturasystem med epost
 - Legge inn og behandle kreditnota
- Rapport-modul
 - Frø på lager
 - Sankeområder
 - Aktuelle frøparti
 - Blandet frøparti fra flere lokaliteter
 - Viser lagerrapport for alle/spesifikke treslag
 - Sankerapport
 - Prissammenligning fra tidligere år

2.2 Personas

Siden gruppen ikke hadde noen kontakt med sluttbruker ble det valgt å benytte personas for å skape en sterkere knytning til sluttbrukerne og holde fokus på brukeropplevelsen av løsningen. Her har vi og benyttet oss av user stories, vanligvis skrevet av kunden, for å foredle kravspesifikasjonen vi har utarbeidet. Dette hjelper oss å konseptualisere kravspesifikasjonen som oppgaver brukerne ønsker å løse ved hjelp av løsningen vår. Dermed deles opp de store kravene til mer håndterlige stories. Disse stories'ene oversettes til epics som deles videre inn i tasks som vi legger inn i YouTrack som vi vil komme nærmere inn på senere. User stories og personas vil og holde fokuset på personer som vil benytte seg av løsningen og hjelpe oss å tenke på brukeropplevelsen av løsningen vår.

For å dekke flest mulige brukere har gruppen også laget fem personas av forskjellige aldre, interesser, og erfaringer med bruk av skogfrøverkets systemer og produkter. I tillegg til å være en morsom og annerledes øvelse, hjelper personas gruppen med å se løsningen fra et annet perspektiv. Har man laget en applikasjon kan alt den gjør virke innlysende, men for en utenforstående kan systemer fort bli kompliserte og uforståelige. Spesielt med tanke på at løsningen gruppen utarbeider kommer fra en gammel applikasjon med et rotete grensesnitt, var brukeropplevelse, gjenkjenning og intuitiv struktur på menyvalgene noe vi så på som spesielt

viktig. Siden gruppen ikke hadde muligheten til å ha kontakt med mulige sluttbrukere, forsøkte medlemmene jevnlig å distansere seg fra egne formeningar. Dette ble gjort ved å drøfte hvordan de forskjellige personas kunne oppfatte løsningen.

Personas danner også grunnlaget for use cases og user stories som gruppen har benyttet seg av for å dele opp de mer omfattende punktene i kravspesifikasjonen. Disse epics som er et artefakt fra smidige utviklingsmetoder blir delt ned i stories som igjen danner grunnlaget for hvilke tasks gruppen planlegger løsningen ut fra.

Alle bilder brukt til personas er hentet fra <https://www.pexels.com/> og er lisensiert under en CC0-lisens.

Navn: Edvin Alfson
Alder: 43 år
Sivilstand: Gift
Brukertype: Intern bruker, ansatt i Skogfrøverket



Edvin har vært i Skogfrøverket så lenge han kan huske. Edvin bor i et småbruk og er hobbybonde på siden av jobben i Skogfrøverket. Han har vært gift i 22 år og har 4 barn. Edvin har nettopp fått smarttelefon men liker ikke at knapper og det analoge forsvinner. Han liker å snekre og fikle med stormaskiner og skulle noe gå i stykker ordner Edvin det selv. På fritiden liker Edvin best turer i skog og mark og kan lett identifisere spiselig og ikke-spiselig sopp. Edvin er stolt av sin egen praktiske kunnskap og setter denne egenskapen høyt hos seg selv og andre. Edvin ser ikke fordelen av å oppgradere systemet Skogfrøverket bruker da han kjenner den gamle ut og inn. Han har vært utenlands bare to ganger i sitt liv og synes det var for varmt.

Navn: Flemming Botten
Alder: 25 år
Sivilstand: Singel
Brukertype: Eksternbruker, privatkunde.



Flemming har valgt å starte med skogsoppdrift etter fullført studie som agronom. Han har derfor blitt med i organisasjonen "Skog og Myr", siden de gir gode råd og rabatter på utstyr til nye skogsfolk. Flemming er opptatt av å være modernisert, og bruker derfor så ofte som han kan den mest moderne teknologien han kan få tak i. Flemming liker å bruke mesteparten av tiden sin alene ute i skogen. På grunn av dette har han ikke skaffet seg en kjæreste enda. Når han ikke er ute i skogen liker Flemming å se på film på den nye laptopen sin.

Navn: Else Christiansen
Alder: 33 år
Sivilstand: Separert
Brukertype: Eksternbruker, planteskole.



Else er utdannet gartner og liker å få ting til å gro og spire. Hun liker godt å sette seg ned med en god bok og en kopp te eller snakke til de sene natterstimer med en god venninne. Hun leter aktivt etter den store kjærligheten og er en selvutnevnt romantiker. Holder sin Facebook og Instagram daglig oppdatert. Har nettopp tiltrådd som administrerende direktør i planteskolen hvor hun jobber.

Navn: Ivar Dal
Alder: 61 år
Sivilstand: Gift
Brukertype: Internbruker, ansatt i Skogfrøverket.



Ivar har jobbet hos Skogfrøverketi mange år og har dermed vært med på deres utvikling gjennom årene. Mange av hans venner jobber også for DNS, det var også der han traff sin kone Tina. Ivar liker å lese om nynorsk litteratur historie. Han bruker derfor en del tid til å rette på andre sin nynorsk på internett. På tross av sin alder er Ivar en hyppig bruker av internett og vegrer seg ikke for ny teknologi.

Navn: Arne Eriksen
Alder: 43 år
Sivilstand: Gift
Brukertype: Superadmin, Gitek.



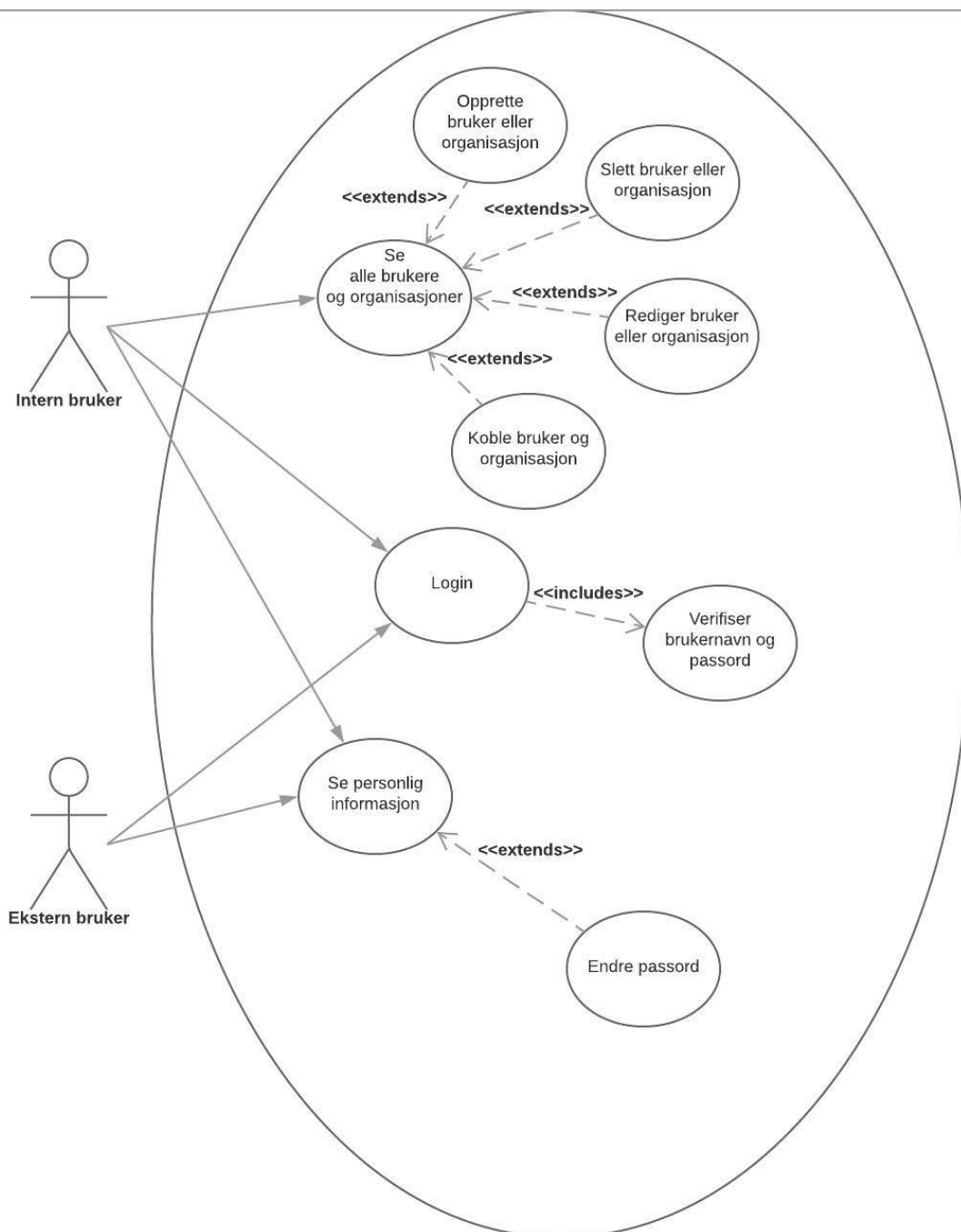
Arne Eriksen, seniorutvikler hos Gitek. 43 år gammel, byttet programmeringsspråk 3 ganger de siste 5 årene og IDE 5 ganger. På fritiden spiller han badminton og skriver anmeldelser på filmer. Har kone og 2 barn i alderen 5 og 11 år, er aktiv på foreldremøtene og håper at barnene vil følge i hans fotspor.

2.2.1 Use cases

Gruppen har laget use cases for å demonstrere hvilke funksjonaliteter de forskjellige brukergruppene har tilgang til. Use casene, kombinert med personas og deres user stories, danner grunnlaget for hvilke tasks gruppen endte med å definere. Siden løsningen er sentrert rundt modulbasert utvikling og leveranse, har gruppen utformet use case diagrammer for de 5 modulene beskrevet i kravspesifikasjonen. Analyse og rapport modulene har blitt slått sammen da det er få unike interaksjoner brukerne har med systemet i disse modulene.

Gruppen har valgt å benytte seg av to aktører da det største skillet i løsningen vil være mellom interne brukere med alle rettigheter og eksterne brukere med få rettigheter. Som nevnt i kapittel 1 er de interne brukerne ansatte i Skogfrøverket. I use casene vil administratorer gå under denne gruppen da det ikke var noen funksjonalitet beskrevet i kravspesifikasjonen som var spesifikt til administratorene. Eksterne brukere består av kunder av skogfrøverket, både de tilknyttet planteskoler og de som ikke er tilknyttet planteskoler. Ved enkelte tilfeller vil brukere tilknyttet planteskoler ha spesifikk funksjonalitet og det vil vises med markering på linjen i use case diagrammet, på samme format som <<includes>> og <<extends>>.

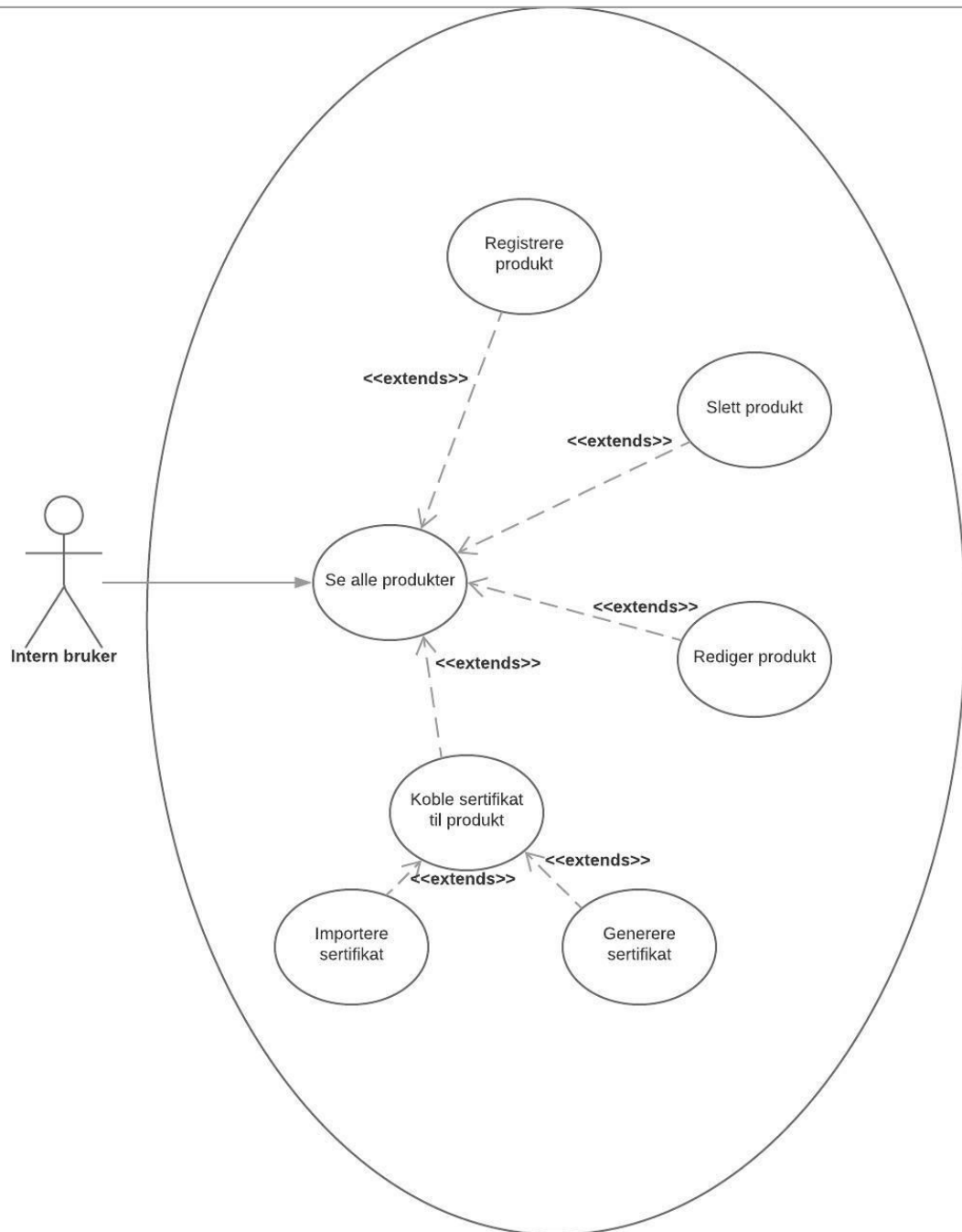
CRM USE CASE



Figur 2: CRM use case

I CRM-modulen vil en ekstern bruker kun ha tilgang til å se egen informasjon og å logge inn i applikasjonen. Interne brukere vil kunne opprette, redigere eller slette alle brukere ved å søke gjennom et brukerregister. Alle brukere vil og ha tilgang på egen informasjon hvor man kan endre passord. For redigering av annen informasjon kreves rettigheter.

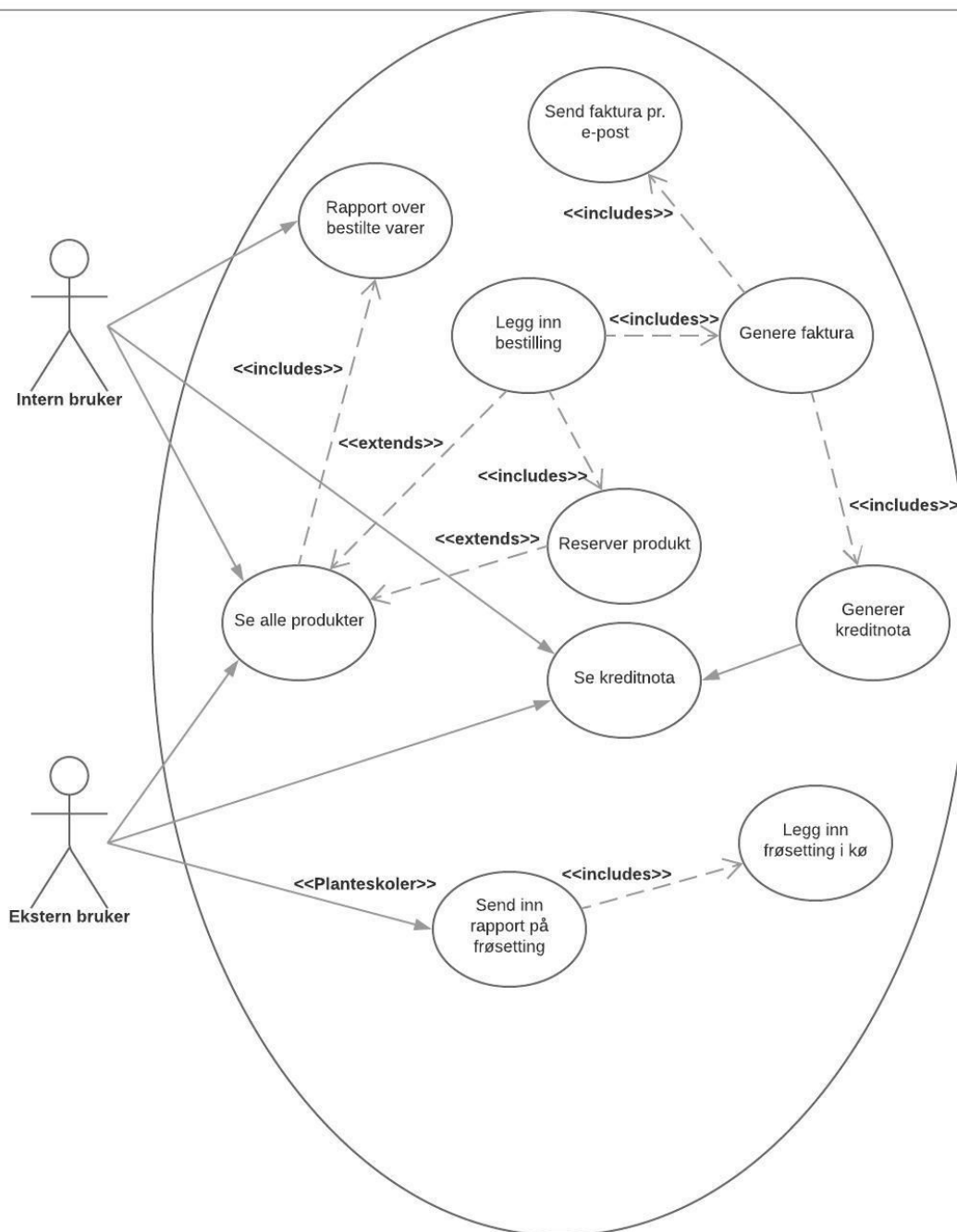
PRODUKTREGISTER USE CASE



Figur 3: Produktregistrerings Use case

Det er kun interne brukere som benytter seg av produktregister-modulen. Som ved crm-modulen velger brukeren ut fra funksjonalitetene som extender seg ut fra "Se alle..." oversikten. Alle produkter kan og generere få heftet på et sertifikat i pdf-format.

ORDRE USE CASE

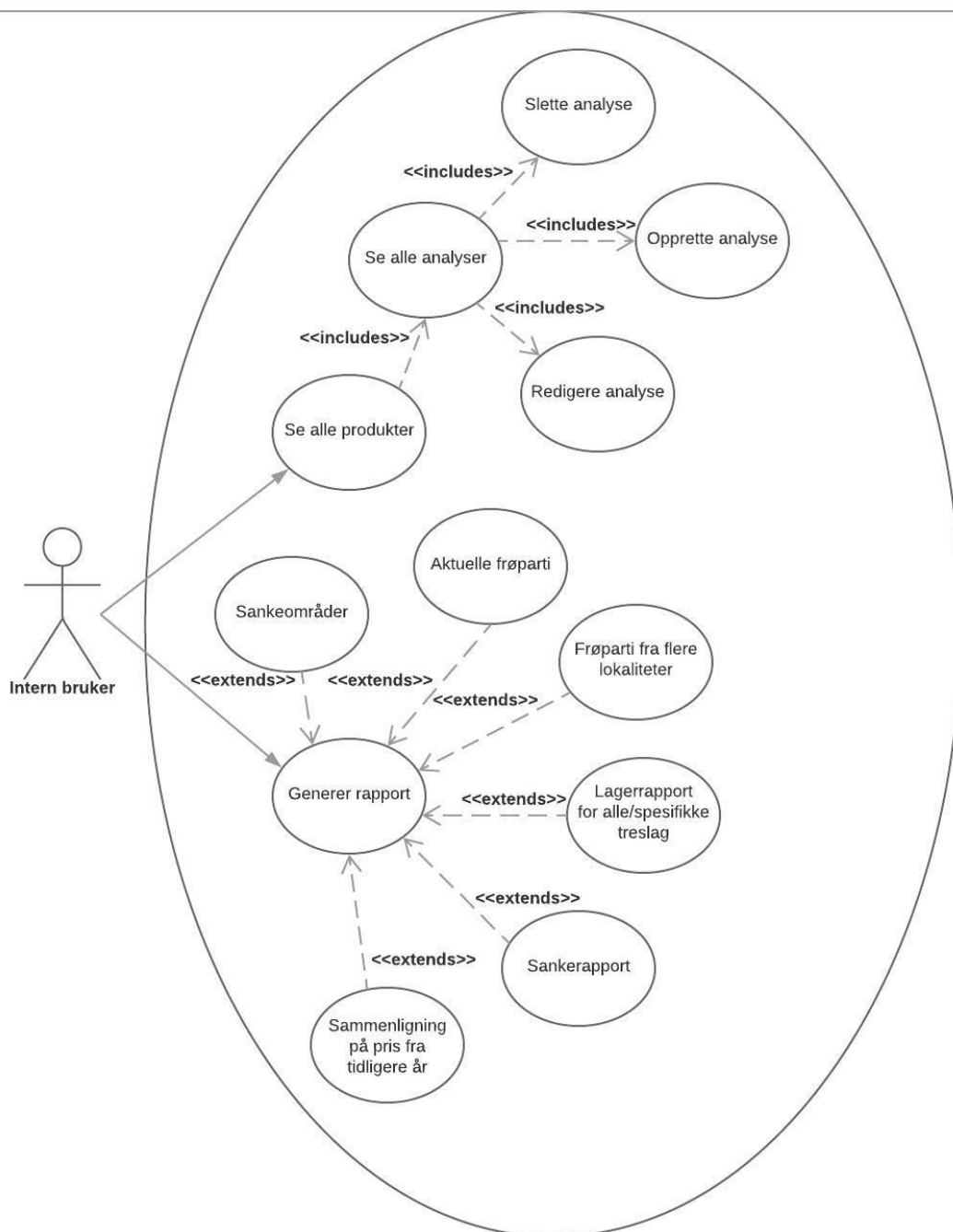


Figur 4: Ordre use case

Både interne og eksterne brukere vil ha muligheten til å se produkter, interne brukere vil og ha muligheten til å se registrerte produkter som ikke er på lager. Det er kun interne brukere som kan hente ut rapporter. Eksterne brukere vil kunne bestille eller reservere et produkt, ved bestilling av produkt vil det bli generert en faktura som danner grunnlag for rapporter og fakturaoversikt. Ved kjøp av produkt vil en epost bli sendt til kjøper. Interne brukere knyttet til

planteskoler kan sende inn rapport på frøsetting som vil behandles av interne brukere på lik måte som registrering av nytt produkt fra produktregister-modulen.

ANALYSE/RAPPORT USE CASE



Figur 5: Analyse/Rapport use case

På det sammenslåtte use caset med analyse og rapport er det igjen kun interne brukere som er representert og igjen går valgene fra en oversikt over mange enheter til funksjonalitet for disse enhetene i dypere menynivåer.

Det som går igjen for brukere, produkter og analyser, er at interne brukere kan benytte seg av det som kalles CRUD, et akronym for opprette(Create), se alle(Read), redigere(Update) og slette>Delete).

Ut fra use casene lagde gruppen extended use cases for å beskrive de forskjellige handlingene vist i use casene på et mer detaljert nivå. For hver modul skapte gruppen et tabulært use case på en sentral funksjonalitet for å kartlegge den dypere: (2, s.145)

CRM: Bruker/organisasjon CRUD	
Aktører	Intern bruker
Beskrivelse	Create, read, update og delete av brukere og organisasjoner. Intern bruker får presentert en oversikt over brukere eller organisasjoner (Read) og kan fra oversikten velge en av de tre funksjonalitetene. Skal brukeren legge ved ny informasjon blir han presentert for et skjema hvor det kan fylles inn informasjonen som skal endres.
Data	Brukerdata og organisasjonsdata.
Stimulus	Intern bruker går inn i menyvalget for organisasjoner eller brukere. Videre trykker han på rediger/opprett/slett knapp i brukers oversikt
Respons	Visning av eller oppdatering av oversikten over dataen.
Kommentarer	Redigering og opprettelse av ny bruker eller organisasjon er det eneste som gir et skjema. For redigering, opprettelse og sletting får bruker opp et dialogvindu. Det er og mulig for bruker å koble brukere og organisasjoner sammen i begge oversiktene.

Produktregister: Vise sertifikat	
Aktører	Intern bruker
Beskrivelse	Bruker navigerer seg inn til en produktoversikt og velger menyvalget for å vise sertifikat. En ny fane åpnes som visning av pdf-fil med mulighet for nedlastning.
Data	PDF-fil knyttet til sertifikat, produkter. Ved generering: Analyse, ordre og produkter.
Stimulus	Bruker trykker på vis knapp i produktoversikten.
Respons	Åpning av ny fane med pdf-visning. Enten av koblet pdf-fil eller generert pdf-fil.
Kommentarer	Dersom sertifikatet er lastet opp tidligere vises denne filen i ny fane, ellers vises et generert sertifikat i ny fane. Ved generering av nytt sertifikat hentes data fra databasen.

ORDRE: Bestille produkt	
Aktører	Ekstern bruker.
Beskrivelse	Ekstern bruker velger produkt ut fra oversikt, velger produktet han ønsker. Produktet legges i handlekurv og reserveres, bruker går til utsjekking og faktura genereres og sendes per e-post.
Data	Brukerinfo, produkter, faktura, handlekurv.
Stimulus	Bruker legger produkt i handlekurv og sjekker ut.
Respons	Oversikt over kjøp, e-post med faktura.
Kommentarer	Kan ikke testet med live data da dette vil gjennomføre en bestilling mot Skogfrøverkets systemer. Kan testes ved å stoppe siste kall mot Giteks API.

ANALYSE: CRUD	
Aktører	Intern bruker
Beskrivelse	Create, Read, Update og Delete av analyser. Bruker får presentert alle produkter med tilhørende analyser(Read), og velger å lage ny analyse, redigere/gjenoppta eksisterende analyse eller slette analyse.
Data	Produkter, analyse.
Stimulus	Bruker trykker på menyvalg for analyser, får presentert oversikt over frøparti med tilhørende analyser og velger et av alternativene.
Respons	Oppdatert oversikt over analyser.
Kommentarer	Analyser har mange felter som skal fylles inn. Lages som et multistep skjema hvor bruker kan lagre påbegynte analyser.

Rapport: Generer rapport	
Aktører	Intern bruker
Beskrivelse	Intern bruker velger rapport etter behov, får generert tabeller og figurer med mulighet til å laste ned på forskjellige format.
Data	Avhengig av rapporten, blant annet produkter, analyser, lagring, brukerdata, faktura.
Stimulus	Bruker trykker på menyvalget rapport, velger derifra hvilken rapport han ønsker.
Respons	Rapport generert på data fra database.
Kommentarer	Enkelte rapporter kan slås sammen. Kan og lage flere rapporter utenfor kravspesifikasjonen under utvikling.

2.2.2 User Stories

I tillegg til use cases og extended use cases lagde gruppen user stories. Use cases beskriver godt hvordan løsningen skal oppføre seg, hvor lite er åpent for tolkning. User stories en veldig vag og generell beskrivelse av hva brukeren ønsker å gjøre og hvordan han benytter løsningen til å gjennomføre det. Dermed kan gruppen både følge de mer rigide use casene for å sørge for at funksjonaliteten blir som forventet og samtidig planlegge ut fra user stories for å fordre det å tenke utenfor boksen.

CRM:

Arne skal endre på brukerinformasjonen til Else fra 30 år til 33 år. Han logger inn i løsningen og finner oversikt over alle brukere. Ved å søke etter Else kan han finne riktig bruker og trykker seg inn for å se hennes kort og endre alderen.

Arne skal legge til en ny planteskole med ansatte. Han legger inn informasjonen til brukerne og organisasjonen og kobler dem sammen.

Produktregister:

Edvin har vakt på mottak av produkter, ved hjelp av nettbrett registrerer han informasjon fortløpende om produktene som kommer inn. Edvin registrerer kun den informasjonen han har for øyeblikket om hvert produkt. Når han er ferdig med ett produkt beveger han seg raskt videre til neste produkt.

Senere skal Edvin registrere den informasjonen han manglet da produktene ankom lageret. Han søker opp partiene via id og registrerer den informasjonen som mangler.

Ivar legger merke til at det har blitt registrert feil antall kilo på gårsdagens produkter. Han søker opp partiene på id og endrer feltet for antall kilo.

Flemming vurderer å kjøpe opp noen parti med frø for å plante opp et skogsfelt, han logger inn på løsningen, søker opp hvilken art han ønsker og får se kilospris, hvor mye som er på lager og hvor langt unna dette er med forventet leveringsdato.

Else har allerede plantet ett skogsfelt hvor frøene skal leveres til Skogfrøverket ved modning. Hun logger inn på løsningen og fyller ut informasjonen i et skjema. Senere legger hun merke til at hun valgte feil art på frøene som ble plantet. Hun søker gjennom sin oversikt over frøsetting og endrer informasjonen.

Registrering:

Ivar skal legge til eldre sertifikat til løsningen som en del av oppgraderingen fra Paradox. Han søker opp partiet via id og laster opp filer som ligger på det lokale filsystemet.

Flemming har kjøpt seg et parti og får vedlagt med kvitteringen, sertifikatet for partiet på pdf-format.

Edvin skal kvalitetsikre at partienes sertifikater inneholder riktig informasjon. Han søker opp partiene på id og generer sertifikatene.

Ordre:

Flemming skal bestille seg frø for å plante et skogsfelt, han går inn på sine anbefalte frøparti og finner arten som passer best til hans område. Han ser nærmere på informasjon om det spesifikke frøpartiet og bestiller det.

Else ringer inn til Skogfrøverket for hjelp med å bestille et frøparti. Edvin tar telefonen og de har en diskusjon om hvilket frøparti Else burde velge. Edvin søker opp frøpartiet og bestiller det ved å søke opp Else og finne hennes bruker.

Ivar ønsker en oversikt over hvilke bestillinger som ikke har blitt levert enda. Han trykker seg inn på rapporter og velger hvilke felter han ønsker å se, deretter laster han dem ned som pdf.

Else ønsker seg frø av en art som Skogfrøverket ikke har inne for øyeblikket. Hun søker opp arten og legger inn en bestilling på antall kilo frø hun ønsker.

Flemming skal se over sin faktura historikk. Han får opp en liste, hvor han kan styre sortering, hva som vises og laste ned som pdf.

Analyse:

Ivar skal registrere analyser på forskjellige partier. Han fyller ut informasjonen han har i skjemaet han får presentert og jobber seg gjennom de forskjellige frøpartiene.

Edvin ønsker å se analysene som har blitt gjort på et frøparti og søker opp partiet. Han laster ned analysekortet på pdf for å kunne se all informasjon som har blitt lagret på analysen til frøpartiet.

Rapport:

Ivar ønsker en oversikt over frø på lager. Han henter ut tabeller og modeller som viser frem frødata på forskjellige format og laster ned på pdf de tabellene og modellene han trenger.

Edvin skal skrive et nyhetsbrev hvor han drøfter popularitet til forskjellige treslag og hvordan dette har forandret tilgjengelighet og pris på frøene. Han henter ut de tabellene og modellene han trenger og laster de ned på pdf-format.

3. Produktplanlegging

Dette kapitlet handler om hvilke metoder gruppen bestemte seg for å bruke gjennom utviklingen av prosjektet, samt planleggingen og forarbeidet for hver modul.

3.1 Modell

For å strukturere utviklingsprosessen er det høvelig å velge en utviklingsmetode som passer prosjektets natur. Gruppen begynte arbeidet med å liste opp de forskjellige modellene. Det handlet om hva slags arbeidsmetodikk modellene har, og hvilke styrker og svakheter de ulike modellene hadde. Dette gav oss et grunnlag som kunne hjelpe oss å sette rammer rundt selve gjennomføringen og dermed bidra til å gi prosjektet best mulig tilpassede arbeidsvilkår.

3.1.1 Utviklingsmetode

Når det kom til valget av utviklingsmetode begrunnet gruppen valget basert på oppdragsbeskrivelsen (se vedlegg) og veiledning fra veileder og oppdragsgiver. Oppdragsgiver, som skrevet i oppdragsbeskrivelsen, ønsket en modulvis inndeling av løsningen med tilhørende releaser. Basert på dette drøftet gruppen som det var hensiktsmessig med en en smidig- eller plandreven utviklingsmetode.

Den første utviklingsmodellen som ble evaluert var fossefallsmodellen, et typisk eksempel på plandreven utviklingsmetode. I forbindelse med drøfting av plandrevne utviklingsprosesser var fossefall den mest aktuelle. Andre modeller som f.eks RUP ble ansett som for omfattende for prosjektarbeidet og selve størrelsesforholdet på prosjektet. I forbindelse med utvikling av den plandrevne utviklingsprosess må man utvikle både planer og strukturer for alle prosessaktiviteter før man kan starte å jobbe med dem. Denne strukturen kan inndeles i fem stadier (2, s.47-48):

1. Krav, analyse og definisjon.
2. System- og program design.
3. Implementasjon og enhetstesting.
4. Integrasjon og system testing.
5. Operativ oppetid og vedlikehold.

Hver fase skal være klar og avsluttet før neste fase i prosessen kan starte. I praksis vil disse utviklingsstadiene overlappe hverandre og gi tilbakemeldinger med informasjon til hverandre. Gruppen vurderte denne modellen grunnet modul leveringens struktur. Enhver modul skulle lages og leveres iterativt, med et klart skille mellom hver modul. Dette i seg selv lignet allerede på fossefallsmetoden.

En plandrevet utviklingsmodell vil lønne seg dersom man utvikler en løsning som er forutsigbar, hvor man kan utarbeide en solid plan i oppstarten av prosjektet, og hvor kravspesifikasjonen vil holde seg stabil gjennom hele prosjektet.

En stor fordel med plandrevne utviklingsmetoder er det faktum at alt arbeid med kravspesifikasjon og design ferdigstilles før implementasjon av løsningen begynner. Dermed kan man avdekke feil og logiske brister ved krav, design og arkitektur før arbeidet med å utvikle løsningen har begynt. Dermed hindrer man større refaktoreringer av systemet som fort kan bli et tidssluk. Ved å ha solide og godt planlagte dokumenter og artefakter før man starter prosessen med å utvikle, vil legges til rette for enkel tidsplanlegging og minske faren for spaghetti-kode.

Kostnaden ved å produsere og godkjenne dokumenter ved bruk av plandrevne utviklingsmodeller kan vise seg å øke tidsbruken betraktelig, disse planene blir lagt i forkant av implementering og tolererer ikke uforventede forandringer. Det er derfor vanlig, ved jobbing i plandrevne utviklingsmodeller, at feil oppdaget i retrospekt av et stadie ikke blir tatt hånd om.

Både gjennom produktspesifikasjonen og konsultasjoner, uttrykte oppdragsgiver sin oppfordring om å bruke smidig metodikk. Den sterkeste argumentasjonen for valg av en smidig utviklingsmodell er at oppdragsgiver ønsker iterativ modulbasert leveranse av det web-løsningen. I tillegg til dette argumenterer Sommerville for at prosjekter som gruppens burde løses med smidige metoder i stedet for fossefallsmetoden: *“The waterfall model is not the right process model in situations where informal team communication is possible and software requirements changes quickly. Iterative development and agile methods are better for these systems”* (2, s.49). Dette var svært nærme vår situasjon, med stor sannsynlighet for endringer i planene, og høy andel uformelle samtaler.

Ettersom valget nå stod på utviklingsmodeller innenfor smidig tilnærming var det tre modeller som ble videre drøftet, Scrum, kanban og eXtreme Programming (XP). Etter at gruppen hadde utarbeidet en samarbeidsplan med oppdragsgiver og veileder om ukentlig møteaktivitet, og at systemet skulle utvikles i moduler som strekker seg fra en til to ukers tidsrom, så gruppen det mest hensiktsmessig å jobbe innenfor rammene til Scrum. Et sterkt argument for å benytte seg av scrum sprinter som arbeidsmetodikk, var de fastsatte møtene med oppdragsgiver hver fredag.

Ved drøfting av alle smidige utviklingsmetoder fant gruppen styrker og karakteristikk i alle modellene. Flere av disse karakteristikkene valgte gruppen å inkorporere i vår utviklingsmodell. Gruppen vedtok blant annet at i de tyngste modulene, med høyest kompleksitet, så skulle gruppen anvende par-programmering fra XP. Gruppen valgte også å ta med noen elementer fra Kanban sin verktøykasse ved arbeid av modulene etter behov, som “kanban board”, “user-stories”, og “epics”.

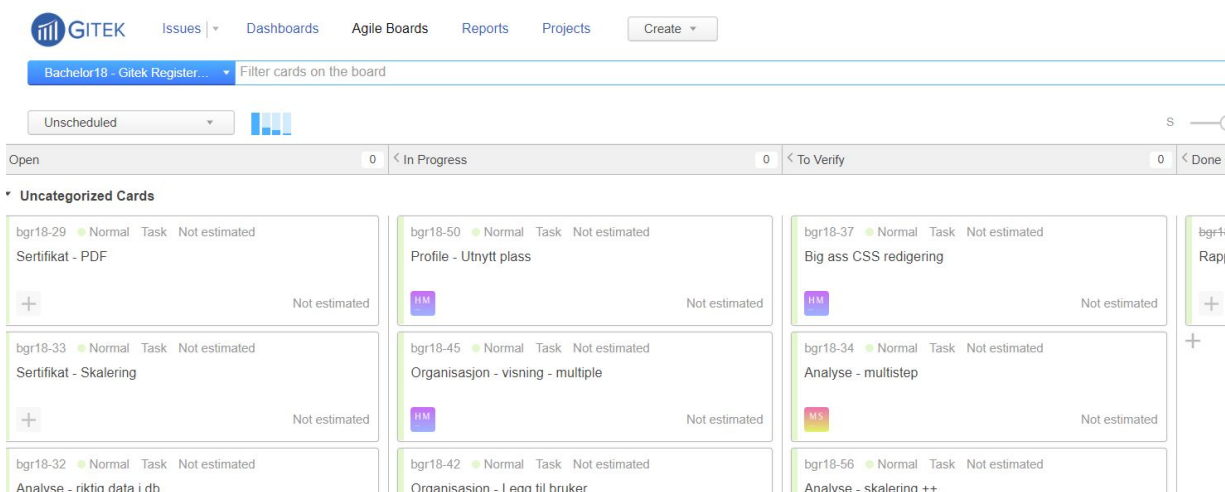
Smidig utviklingsmetodikk har svakheter som gruppen måtte være oppmerksom på. Ved smidig utvikling jobber man i mindre iterasjoner hvor dokumenter og modeller utvikles parallelt med

implementasjon. Dette kan fort skape gjentakning av funksjoner og dårlig struktur på arkitekturen som krever refaktorering. Svak dokumentasjon som et resultat av uformell kommunikasjon og skjulte vanskeligheter i løsningen kan skape større arbeidsmengder enn det som først er antatt. Leveransen til et smidig utviklingsprosjekt er vanskeligere å beregne da nye krav kan komme langt inn i prosessen.

3.2 Taskboard

Når det kom til taskboards så gruppen på alternativer som Wrike, Pivotal Tracker, Jira, Trello, og YouTrack. Trello og Pivotal Tracker var for lettevekter for gruppens behov, skulle gruppen benytte seg av Jira burde den ligge på skolens systemer. En viktig faktor i valget av taskboard var at gruppen allerede hadde bestemt seg for å benytte seg av PHPStorm, en JetBrains IDE. Derfor valgte gruppen å benytte seg av YouTrack da dette er et JetBrains produkt som inneholdt de funksjonalitetene gruppen ønsket. Etter å ha konsultert med oppdragsgiver fikk vi tilbud fra dem å bruke YouTrack, hvor de var vertskap for server for prosjektgruppen.

Gruppen valgte derfor å bruke YouTrack, oppsatt på oppdragsgivers server. YouTrack åpnet for at gruppen kunne lage scrumboards som kunne iterere over flere sprints. Disse boardsene var basert på backloger som gruppen laget, og viste til oppdragsgiver. Når backlogen var ferdig kunne gruppen dra inn issues til boardet og jobbe på dem. Boardene var delt inn i fire felt: *Open*, *In Progress*, *To Verify*, og *Done*. *Open* holdt de kortene som gruppen planla å jobbe med i løpet av den gjeldende sprinten. *In Progress* holdt de kortene som medlemmene av gruppen jobbet på. *To Verify* holdt kortene som et medlem hadde blitt ferdig med. Et annet medlem ville så gå over å se om kvaliteten på arbeidet var av høy nok kvalitet til å levere til oppdragsgiver. Hvis arbeidet var det så ble kortet flyttet til *Done*. Hvis ikke ble det flyttet tilbake til *In Progress* og jobbet mer på. Medlemmene kunne også sette seg opp på kortene slik at resten av gruppen kunne se hvem som jobbet på den oppgaven. Her er et utklipp fra YouTracken underveis i prosjektet.



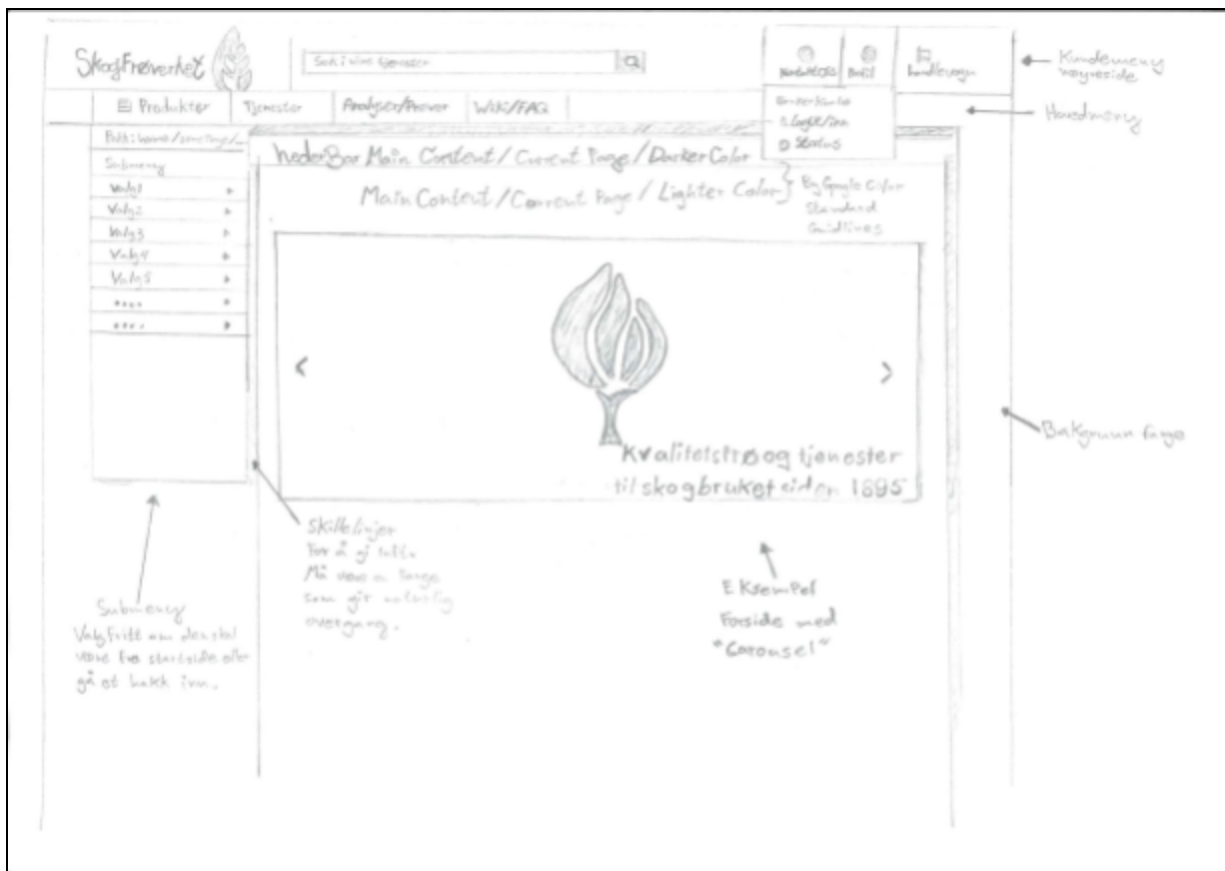
Figur 6: vårt YouTrack board

YouTrack har også en issues modul som lar brukerne se alle issues som de er knyttet til i en liste. Denne listen kan så sorteres til å vise mer spesifikke issues, for eksempel hvilke issues de har blitt satt opp til å løse.

3.3 Prototyper

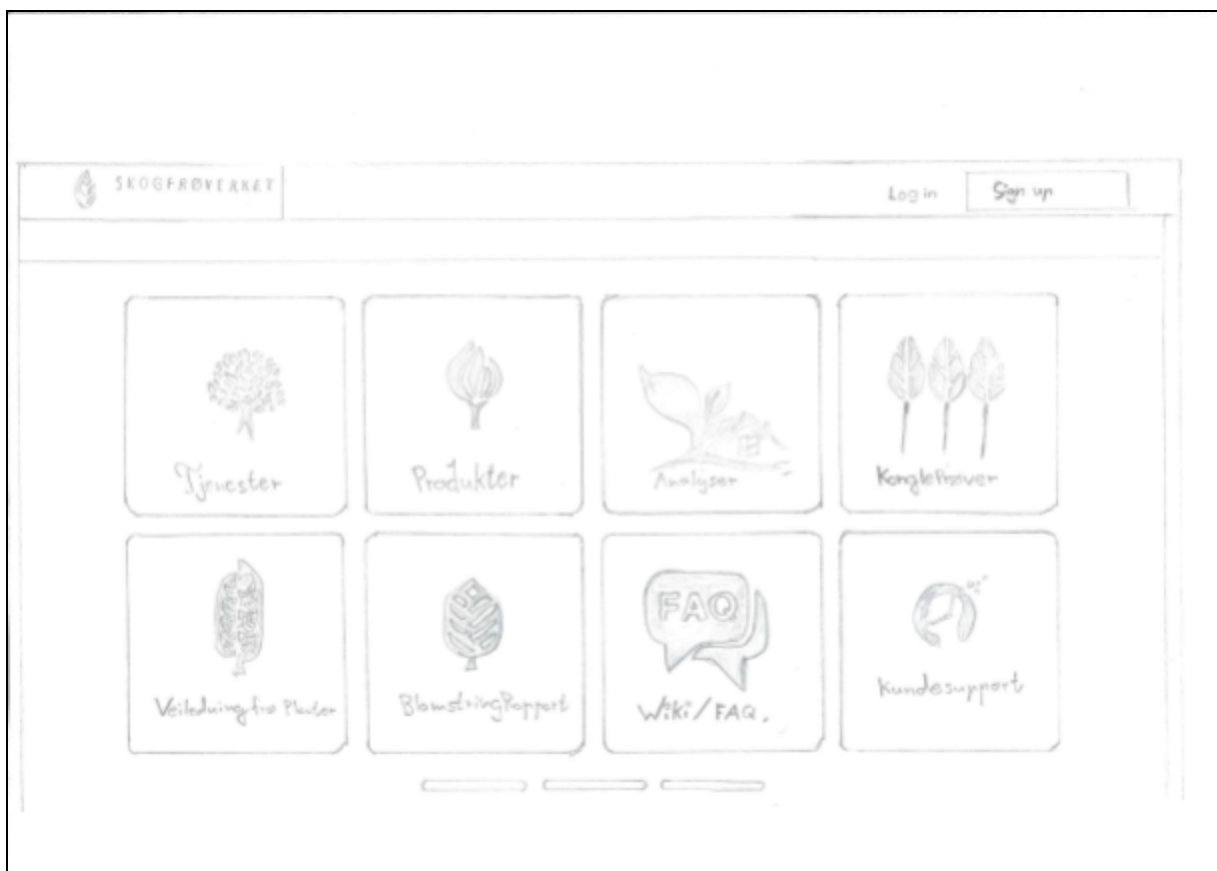
For å sørge for at gruppens medlemmer og oppdragsgiver hadde en tilnærmet lik konseptuell modell av hva som skal utvikles, tok gruppen i bruk prototyper. Disse hjalp også i diskusjoner internt i gruppen og med oppdragsgiver om hvordan grensesnittet skulle se ut. Gruppen startet prosessen med å la alle gruppens medlemmer lage sin egen prototype for så å møtes for å slå sammen de idéene som skulle føres videre. Senere i prosessen beveget gruppen seg fra spesifikke håndskrevne prototyper for hver enkelt løsning, over til mer generelle digitale prototyper. Dette skiftet i metode kom av målet gruppen har om gjenkjennbarhet i grensesnittet, ved å standardisere utseende og flyten til programmet. *(A4-størrelse på prototypene ligger i vedlegg B)*

Siden gruppen ønsket en modulbasert løsning ble prototypene utviklet parallelt med implementasjonen av løsningen, hvor prototyping var en del av planleggingen av neste sprint. Gruppen så samtidig fort at det å lage for detaljerte prototyper ville ta mer tid enn gevinsten de gav og vi bevegde oss mot mer generelle prototyper som beskrev et overordnet layout. I oppstartsfasen trengte gruppen prototyper for landingssiden til brukere. I en workshop hvor gruppen vurderte de forskjellige grovskissene, endte vi opp med tre prototyper for landingssiden til applikasjonen:



Figur 7: Prototype - forside pc

Den første skissen drar mye inspirasjon fra web-shop da dette er bruksområdet for eksterne brukere. Her er det en sidemeny hvor brukere kan spesifisere hvilken produkttype de ønsker å se. Headeren har en handlevogn som tar vare på hvilke produkter brukeren ønsker å kjøpe, en kontakt oss knapp og en profilside. Den har og en global søkefunksjon som vi senere gikk bort fra. I hovedvinduet ønsket vi i denne prototypen å ha en nyhets-karusell hvor interne brukere kunne definere innholdet. I møte med oppdragsgiver fant vi at denne prototypen var utenfor hva Gitek så for seg, men vi tok vare på konseptet om breadcrumbs og sidemeny.



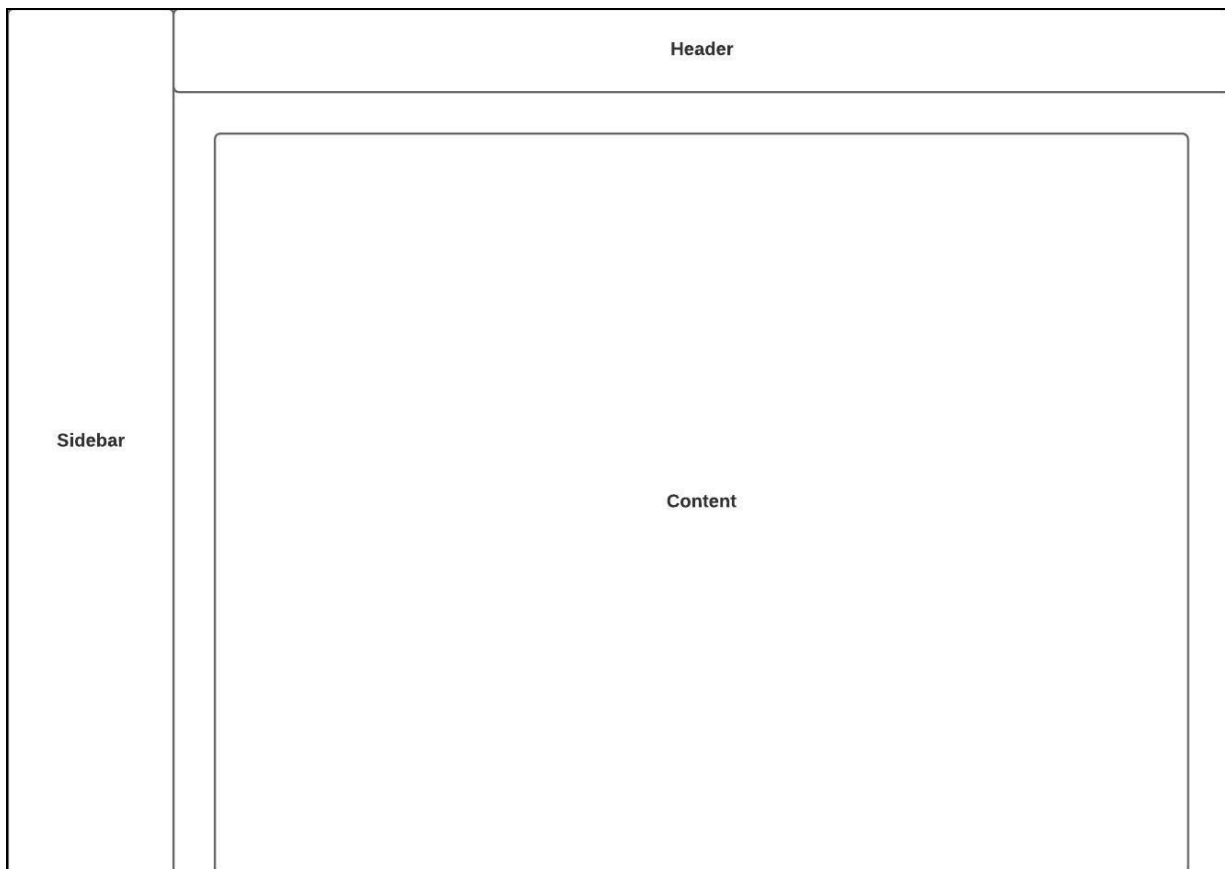
Figur 8: Prototype - forside tablet

Den andre prototypen var inspirert av hvordan nettbrett og telefoner presenterer applikasjoner for brukeren. Her presenterer landingssiden de forskjellige funksjonalitetene til applikasjonen i forskjellige bokser med ikoner. Denne prototypen dannet mye av grunnlaget for hvordan landingssiden endte opp.



Figur 9: Prototype - valg i header

Den tredje prototypen søkte å blande de to tidligere prototypene ved å presentere valgene en bruker har i headeren og samtidig ha en mindre header og sidebar som man ser i web-shop. Ved forsøkt implementasjon av denne prototypen fant gruppen at ikonene i headeren ville ta for mye av siden og vi valgte heller å legge disse inn i hovedvinduet og bruke sidebaren som en speiling av denne funksjonaliteten. Dermed vil brukeren lett kunne navigere seg rundt i applikasjonen.

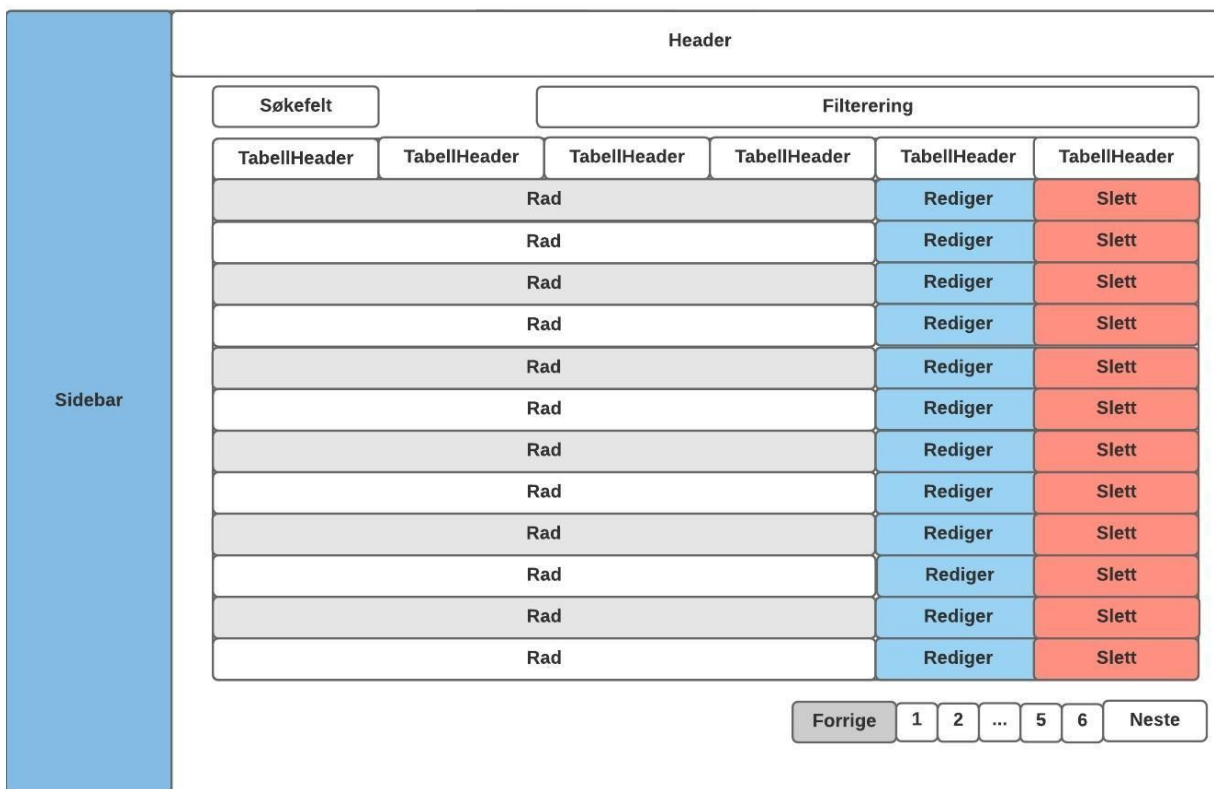


Figur 10: prototype - generell layout

Som en generell layout av sidene har gruppen bestemt seg for å alltid vise header og sidebar som er begge statiske i innhold. Det eneste som forandrer seg er highlighting av hvilket element i sidebaren som er aktivt for å hjelpe brukeren å navigere seg i løsningen. Denne highlightingen er en variant av UI-mønsteret navigation tabs, eller faner.⁴ I stedet for horisontal visning, har gruppen valgt å legge faner vertikalt inn i sidebaren for å utnytte bredde.

⁴ "Navigation Tabs design pattern - UI-Patterns.com." <http://ui-patterns.com/pattern/NavigationTabs>. Åpnet 14 mai. 2018.

PROTOTYPE: TABELLFREMVISNING

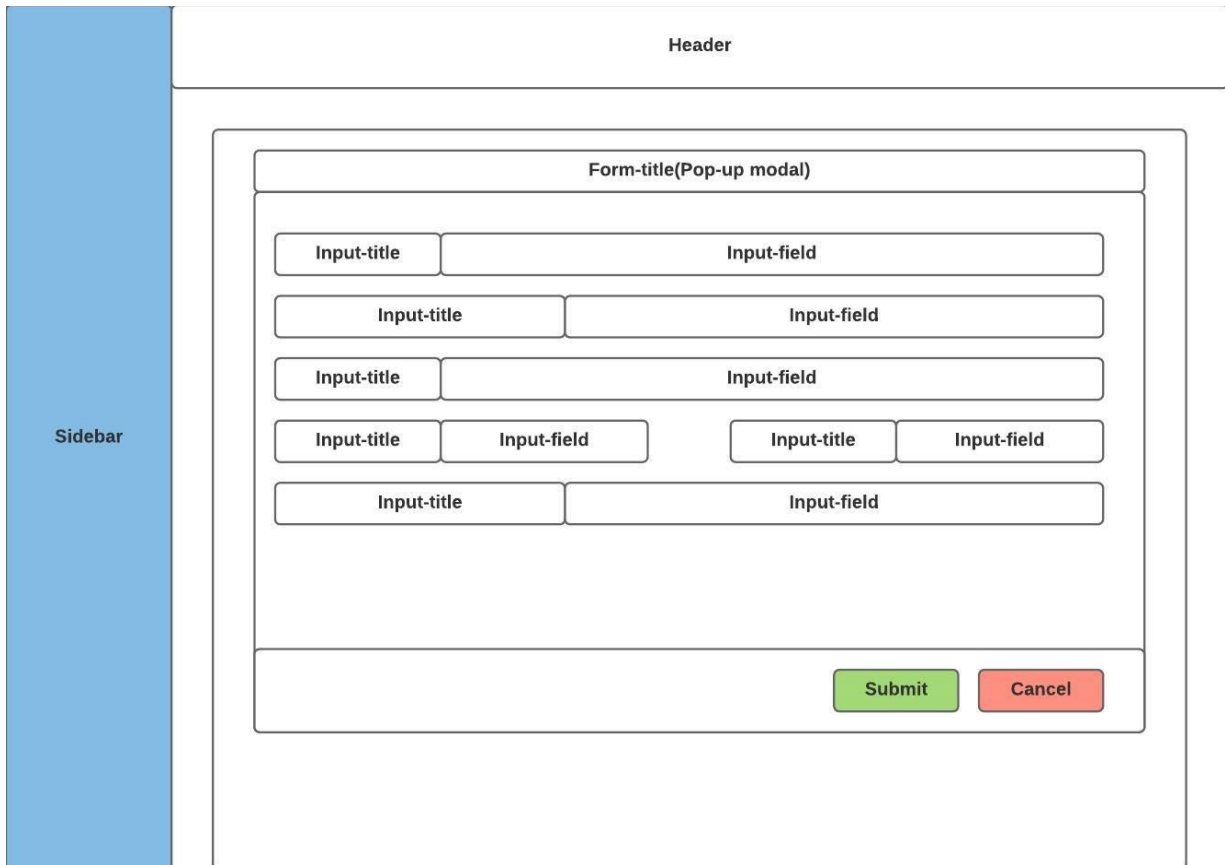


Figur 11: Prototype - tabellfremvisning

Gruppen valgte å legge ved en bakgrunnsfarge til sidebaren for å vise frem funksjonalitet i sidebaren, samtidig brøt det med den hittil svart/hvit fargepaletten. Ved fremvisning av større data benytter gruppen seg av tabeller. Tabellen i seg selv alternerer mellom hvit og grå for å hjelpe brukeren å skille mellom radene i henhold til UI-mønster om tabeller⁵. Over tabellen kan brukeren søke seg frem til data eller definere filtre å benytte seg av. Til høyre i tabellen ligger spesifikk funksjonalitet for rader. Da det er mye informasjon som vises har gruppen valgt å benytte seg av UI-mønsteret pagination⁶ som lar brukeren bla seg gjennom flere sider av informasjon.

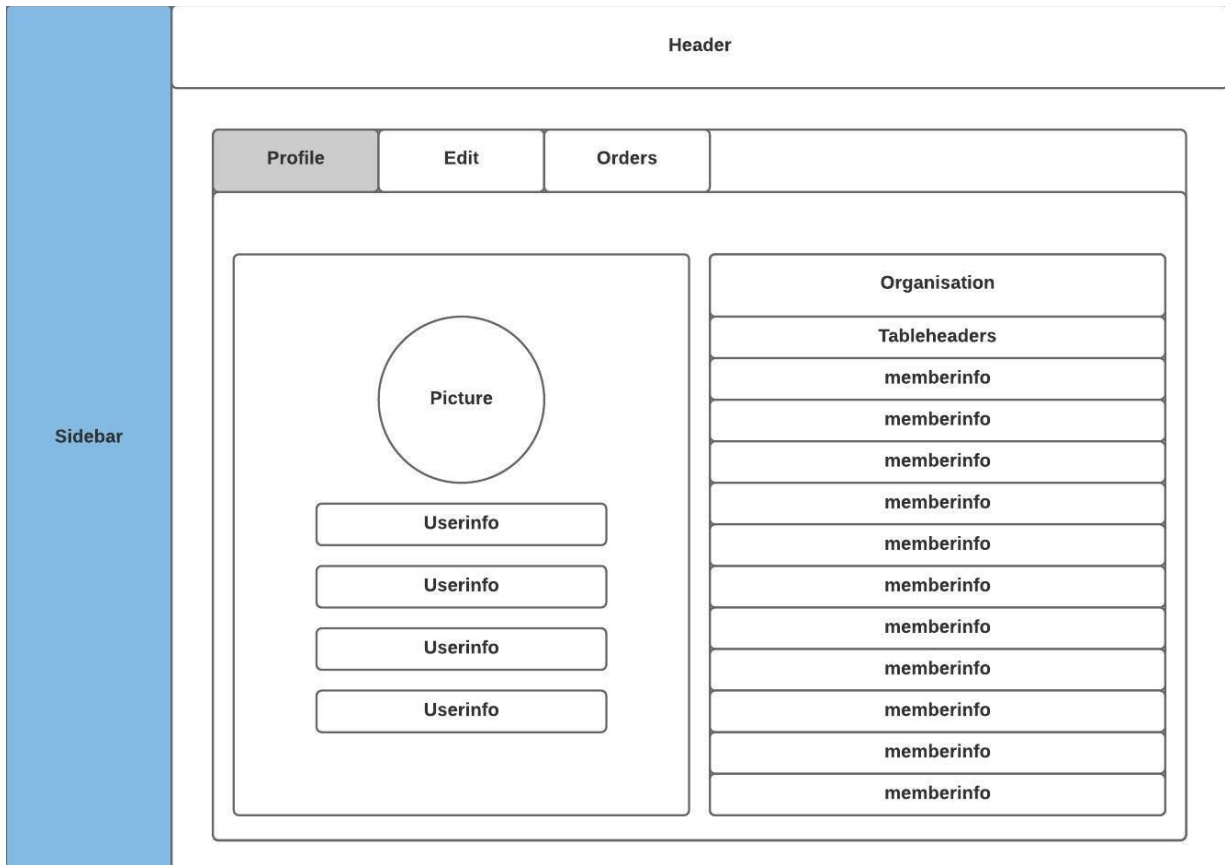
⁵ "Alternating Row Colors design pattern - UI-Patterns.com." <http://ui-patterns.com/pattern/AlternatingRowColors>. Åpnet 14 mai. 2018.

⁶ "Pagination design pattern - UI-Patterns.com." <http://ui-patterns.com/pattern/pagination>. Åpnet 14 mai. 2018.



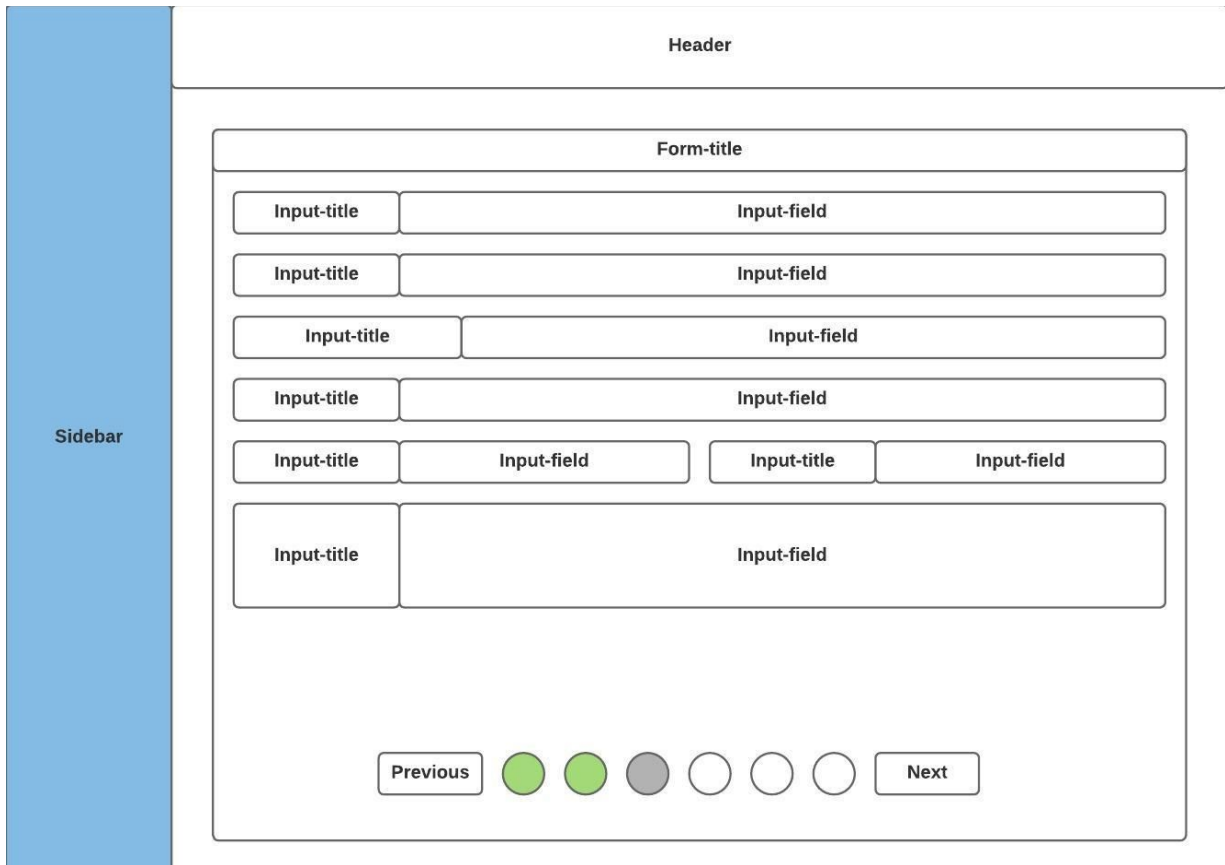
Figur 12: Prototype - endring av data i tabell

Dersom brukeren ønsker å endre på data i tabeller vil det komme et dialogvindu hvor brukeren kan endre på felter. Etter korrekt utfyllt form vil dialogen forsvinne og brukeren vil bli presentert med den oppdaterte siden.



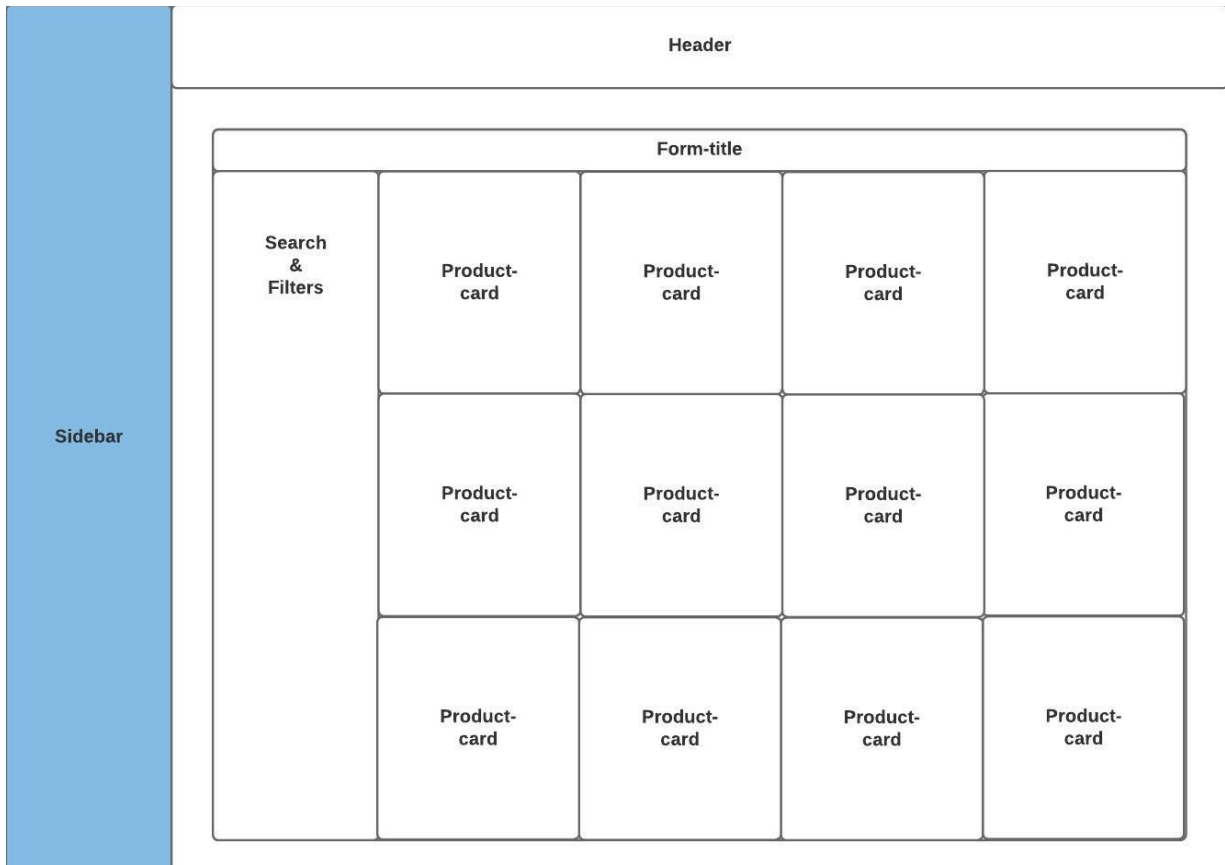
Figur 13: Prototype - min side

For min-side har gruppen valg å benytte seg av tabs som man vanligvis ser i nettlesere. Her knyttes og informasjon mellom bruker og hans organisasjon. Et eget felt for å redigere personlig informasjon hvor gruppen ønsker å vise informasjon som er skjult i profilsiden. Denne siden vil kreve rettigheter for å vise. Siste tab er ordre-siden hvor brukeren kan se sin ordrehistorikk og aktive ordre.



Figur 14: Prototype - pagination

Nesten all ny informasjon som skal inn i systemet har mange felter som skal fylles inn. Gruppen har derfor valgt å benytte seg av pagination for å vise brukeren hvilke "sider" av skjemaet han har besøkt, hvilke han har fylt ut og hvor han er nå. Grønn er sider som er ferdig utfylt, grå er sider som er besøkt. En fremhevet sirkel viser hvor brukeren er for øyeblikket og tomme sirkler er sider brukeren ikke har sett enda. Senere i utviklingsprosessen så gruppen seg nødt til å tillate brukeren å lagre enkelte skjemaer som ikke var ferdig utfylt da brukeren i sin arbeidsflyt ikke ville ha tilgang på all informasjon som skal registreres.



Figur 15: Prototype - fremvisning av produkter

For fremvisning av produkter valgte gruppen å legge til en web-shop inspirert visning, her vil brukeren få presentert produktene i et rutenett med bilde av produktet med tilhørende informasjon. Her vil gruppen igjen benytte seg av pagination for å vise brukeren store mengder med likt innhold.

4. Design og Arkitektur

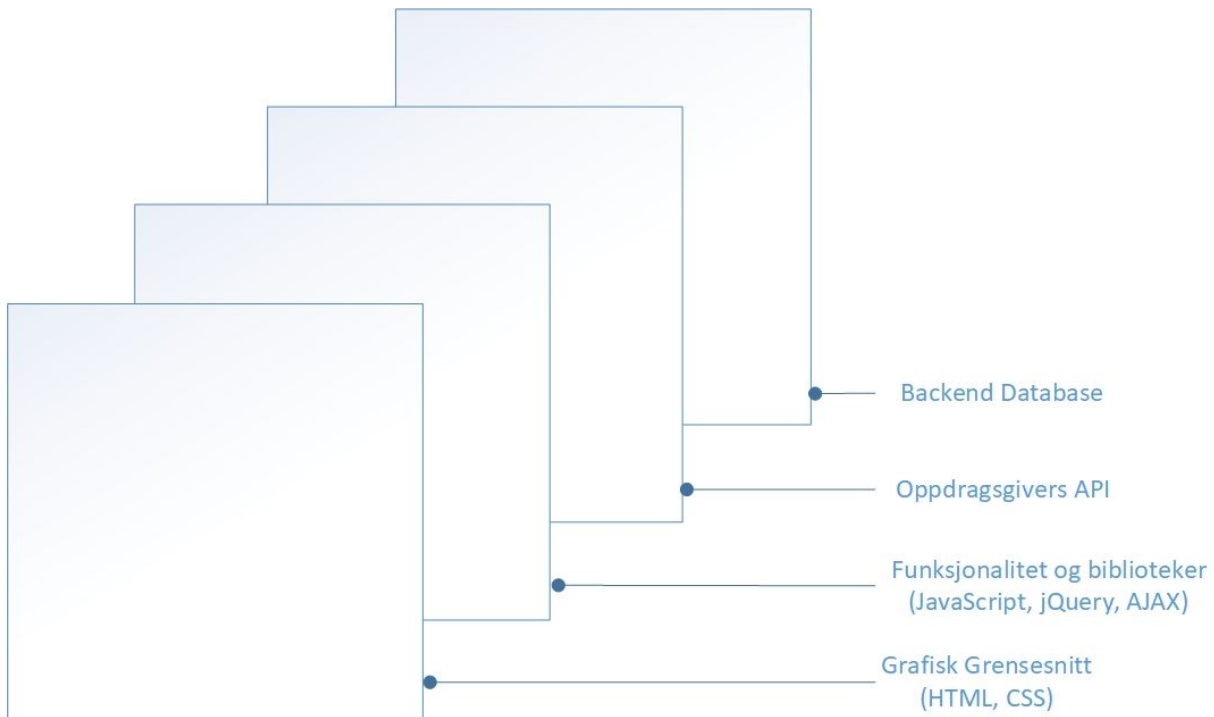
I dette kapitlet viser vi til hvordan strukturen på programvaren, og rammene rundt er satt opp, og argumenterer for forskjellige designvalg gjennom prosjektet. Det vises til forskjellige detaljer rundt utviklingen av kjernefunksjonalitet, hvordan dette blir presentert for brukeren når en ankommer siden, og de forskjellige funksjonene som utfører disse oppgavene. Vi diskuterer først valg rundt bakgrunns-strukturen og går så videre til å se på det grafiske grensesnittet som ligger på toppen.

4.1 Arkitektur og mønstre

For å forsikre om at alle gruppemedlemmene hadde tilsvarende konseptuelle modell om hvordan løsningen skulle struktureres, ble det utformet modeller og retningslinjer for arkitekturen. Dette var et tiltak for å passe på konsekvent arbeidsmetodikk og bedre gruppemedlemmenes forståelse av kodebasen.

4.1.1 Lagdelt arkitektur

Programvaren er lagdelt med HTML og CSS som styrer utseende og det grafiske grensesnittet som presenteres for brukeren. All funksjonalitet blir styrt av Javascript med jQuery og AJAX, dette laget fungerer som en controller mellom grafisk grensesnitt (View) og Oppdragsgivers API og database (Model).



Figur 16: lagdelt arkitektur

4.1.2 Repository struktur

Fra prosjektets oppstart benyttet gruppen det veletablerte Model View Controller mønsteret. På dette tidspunktet hadde vi ikke tilgang til informasjon om hvordan databasen og oppdragsgivers rammeverk så ut. Vi hadde heller ikke gjort oss kjent med oppdragsgivers servermiljø. Med disse variablene utenfor vurderingen, så gruppen heller på fordeler og ulemper, uavhengig av tekniske barrierer.

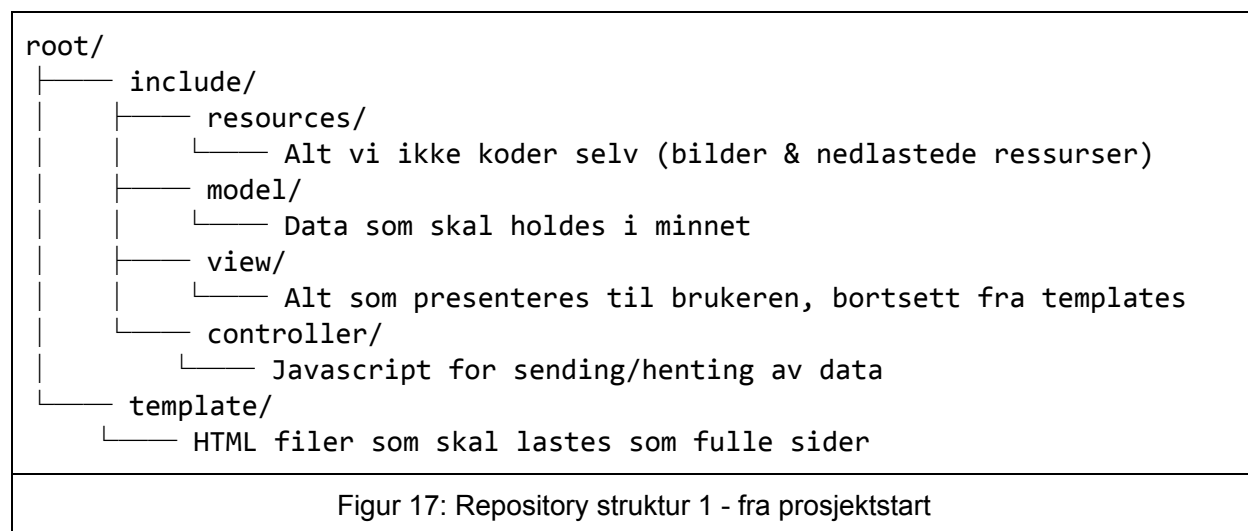
En veldig relevant faktor som skapte et sterkt argument for å velge MVC, var gruppestørrelsen på fire medlemmer. Utfordringen med en større gruppe er å få utnyttet arbeidskraften mest effektivt, og minimert bortkastet innsats. Ved å dele opp løsningen i tre sammenkoblede lag, tillater vi parallell utvikling på enkeltdeler av en modul. I den fiktive situasjonen som ble argumentert, kunne tre av medlemmene jobbe på hver sin del, samtidig med parprogrammering på den mest utfordrende delen.

Forbi dette, tillater lagdelingen oss å la enkelte gruppemedlemmer jobbe med ønsket fagområde, relevant til erfaring og egenskaper. Gruppen har tilegnet seg varierende nivå med kunnskap når det kommer til å designe brukergrensesnitt, kontra skrive skript som håndterer

database interaksjon. Med lagvis inndeling kan vi utnytte spisskompetanse, og ha en situasjon hvor gruppe medlemmene jobber med utfordringene de er best utstyrt til å håndtere.

En annen fordel som motvirker samme utfordring, er hvordan endringer i ét lag ikke påvirker resten av modellen. Med gruppestørrelsen i tankene, er det verdifullt å kunne gjøre kodeendringer på et sted, uten at det får konsekvenser for hele løsningen. Det er beleilig å kunne endre utseendet på en meny, uten at controlleren som er ansvarlig for business-logikken bak blir ødelagt. I tillegg vil strenge rammer på mappestrukturen hjelpe gruppen å holde en strukturert og oversiktlig kodebase, dermed minskes faren for duplisering av koden og det oppfordres til gjenbruk av eksisterende funksjoner.

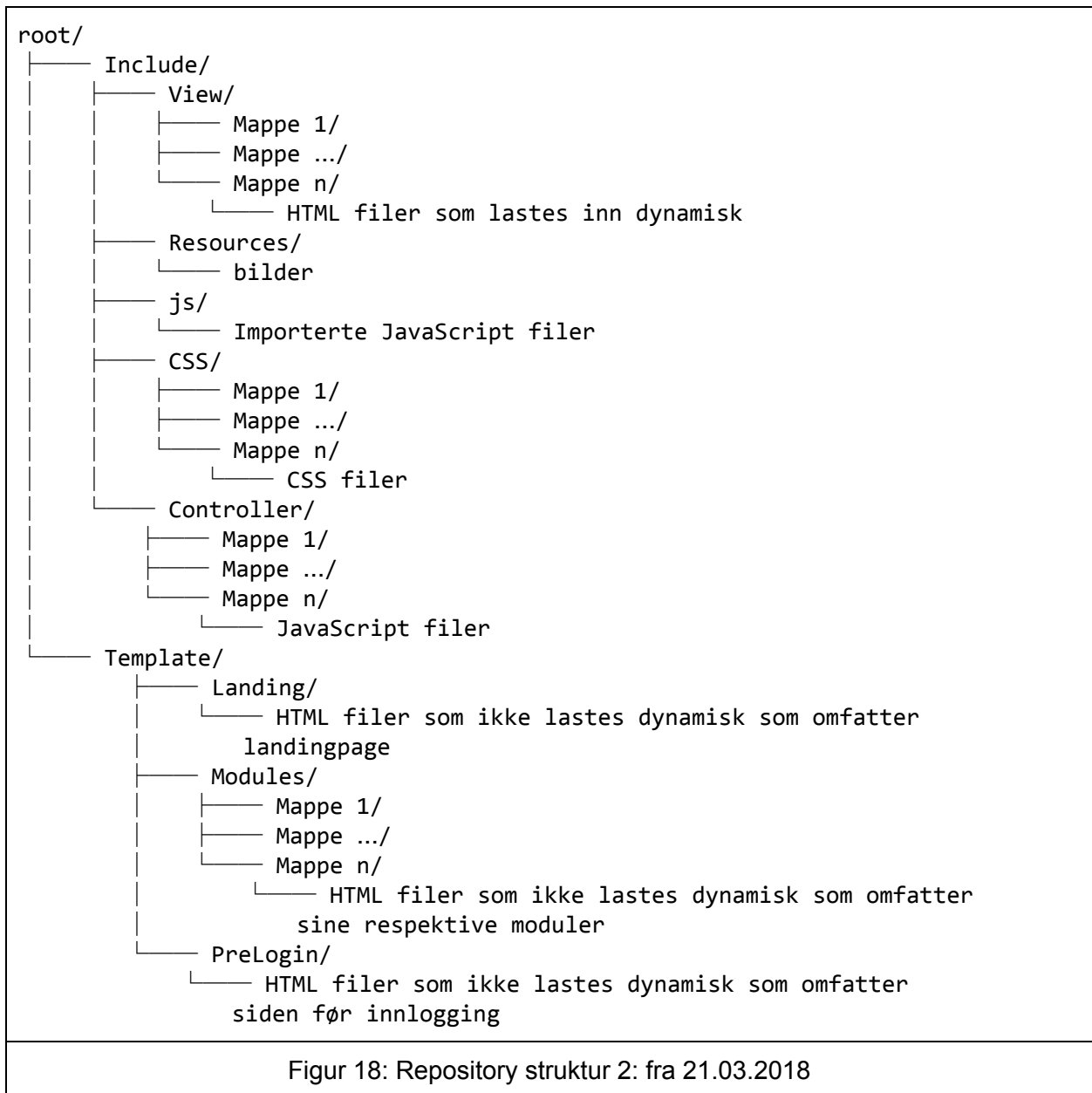
Strukturen på repositoret vårt hadde følgende struktur for den første fasen av prosjektet. Her har vi brukt det mest essensielle fra MVC, med respektive mapper for “Model”, “View”, og “Controller”.



Mot siste halvdel av mars fikk vi muligheten til å sette opp oppdragsgivers servermiljø på våre egne laptop, og trengte derfor ikke lenger å benytte oss av remote tilkobling til deres servere. Dette tillot oss å kunne gjenskape et miljø som simulerer hele situasjonen med koden vår i produksjon. Ved dette skiftet ble det nødvendig å gjøre noen endringer og omstruktureringer av repositoret for å forsikre kompatibilitet. Hver template HTML-fil behøvde nå sin dedikerte inf-fil som beskriver sidens navn i url-en. Vi ble også nødt til å sette opp ini-filer som inneholdt konfigurasjonsdata for å slå opp i database og paths til sentrale mapper.

I tillegg til nødvendige endringer i overgangen, ble det valgt å gå vekk fra å bruke “Model” laget fra MVC. Dette valget ble gjort for å kunne benytte oppdragsgivers back-end, i prosessen å integrere i deres systemer. Dette innebar at gruppen kom til å bruke deres eksisterende databaseløsning og API for AJAX interaksjon. I stedet beholdt vi “Views” og “Controller”, og gjennomførte alle oppdateringer av “Views”, gjennom “Controlleren”. Det ble også klart etter en stund at vi ikke kom til å få et behov for å vedlikeholde dynamisk innhold, som søkefunksjoner,

notifikasjoner eller nyheter. Fra uke 12 (19.- til 25. mars) til prosjektets slutt hadde vi følgende struktur.

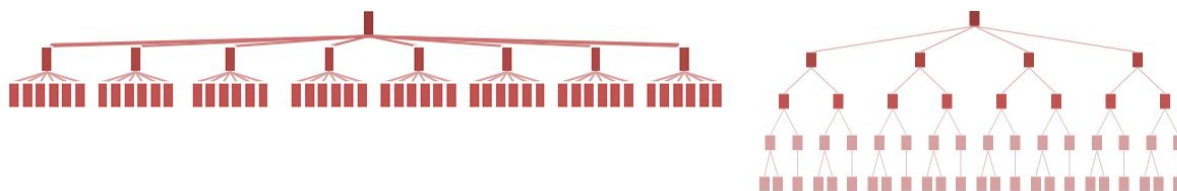


Det også gjort endringer for å appellere til ønsket om modularitet. Gruppen valgte å jobbe for at det skulle være mulig å eksportere komponenter som f.eks brukertoversikt, frøpartioversikt eller analyse. Målet var at ingen komponenter skulle ha avhengigheter som forhindrer at de kunne brukes i isolasjon. For å oppnå dette, valgte vi å lage undermapper for de sentrale mappene “View”, “CSS”, “Controller”. Hver komponent har følgende dedikerte filer i respektive mapper. Dette visualiseres i repository struktur 2 over med undermappe forholdet 1 .. n. Ved å gjøre komponentene i løsningen uavhengige oppnådde vi også gevinst for alle template-sidene. På

en hver side kunne vi laste inn komponenter i løsningen etter behov. Hvordan dette fungerer i praksis, beskrives under kapittel 5.3.3.

Controller inneholder nå all JavaScript-kode og View inneholder HTML-filer som ikke lastes direkte. Template inneholder hovedsakelig "skjelett" HTML-filer, hvor innholdet i filene lastes dynamisk fra View mappen ved hjelp av Giteks rammeverk og JavaScript. Alle filene er sortert i mapper etter hvilke modul de kommer under.

4.1.3 Flatt webløsning hierarki



Figur 19: Flatt vs dypt hierarki⁷

I avgjørelsene for informasjonsarkitekturen falt det viktigste valget på om løsningen skulle ha et flatt eller dypt hierarki. Her benyttet vi Nielsen Norman Group som viktigste kilde for å kartlegge styrker og svakheter. Det var to sentrale faktorer som ble tatt med i vurderingen under undersøkelsen av de to alternativenes egenskaper. Først og fremst skulle det tilrettelegges for at løsningen skal være brukervennlig på tablet. Den andre faktoren er hvordan primærmålgruppen allerede er velkjent med eksisterende løsning, samt fagspråk.

1. "Content is more **discoverable** when it's not buried under multiple intervening layers. All other things being equal, deep hierarchies are more difficult to use."⁸

At innholdet i løsningen skal være synlig og lett tilgjengelig uavhengig av hvor brukeren befinner seg i løsningen, er noe gruppen vurderte som attraktivt. Synlighet og oversikt er essensielt for at løsningen skal være en forbedring av den gamle.

2. "Categories that are **specific and do not overlap** are the easiest to understand. This cuts both ways: In deep hierarchies, when there are only a few categories on each level, they tend to be more generic and, thus, more confusing. A flatter hierarchy with more categories at each level usually has **more-specific labels** that are easier to understand; but in broad

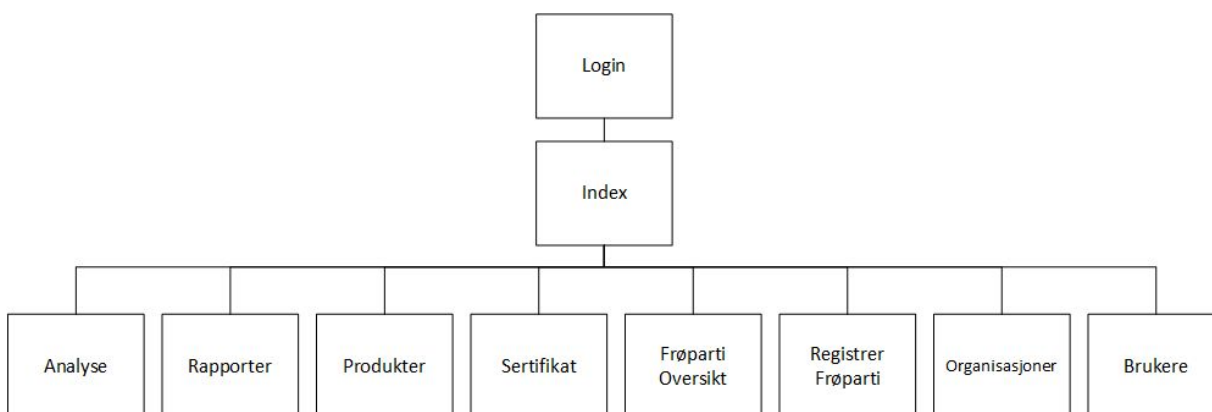
⁷ "role hierarchy – Customer Feedback for Discord - Discord Dream Land." 24 mar.. 2018, <https://feedback.discordapp.com/forums/326712-discord-dream-land/suggestions/33739801-role-hierarchy>. Åpnet 15 mai. 2018.

⁸ "Flat vs. Deep Website Hierarchies - Nielsen Norman Group." 10 nov.. 2013, <https://www.nngroup.com/articles/flat-vs-deep-hierarchy/>. Åpnet 9 mai. 2018.

hierarchies with a very large number of items, there is often some conceptual overlap between at least a few of the categories. Users can also become overwhelmed with long, cluttered menus.”⁹

På dette punktet veier faktoren om en erfaren målgruppe inn. Det at majoriteten av brukerne som vi appellerer til har kjennskap til faglige ord og uttrykk, gav rom for å lage mer spesifikke labels, i navigasjonsmenyene. Det hjelper også at vi ikke behøvde mange elementer tilgjengelig på samme tid. Dette er attributter som veier for valget av flat struktur.

Tidligere i planleggingen, ble diskutert et dypt hierarki, med bruk av breadcrumbs¹⁰. Argumentet ble gjort for å legge til rette for gruppens forventning om et større antall sider som måtte gjøres tilgjengelig for brukeren. Dette endte opp med å ikke være tilfelle, og vi bestemte oss for en flat struktur, visualisert under.



Figur 20: Prosjektets kategorier

Her har innholdet blitt separert i åtte kategorier. Det skulle i utgangspunktet være ni kategorier, men den siste falt bort grunnet datagrunnlag i ordremodulen. Mer om dette i kapittel 7.1. Målet var å begrense kategoriene til å være meningsbærende, men likevel i et begrenset antall. Uavhengig av hvor brukeren befinner seg i løsningen etter innlogging, vil hierarkiet med utgangspunkt i index være gyldig. Dette oppnås med å ha kategoriene synlige i navigasjonsbaren til alle tider som en statisk informasjonsbærer. Hver kategori er tenkt for å løse en spesifikk arbeidsoppgave så en bruker vil typisk benytte seg av en eller to kategorier per besøk av siden.

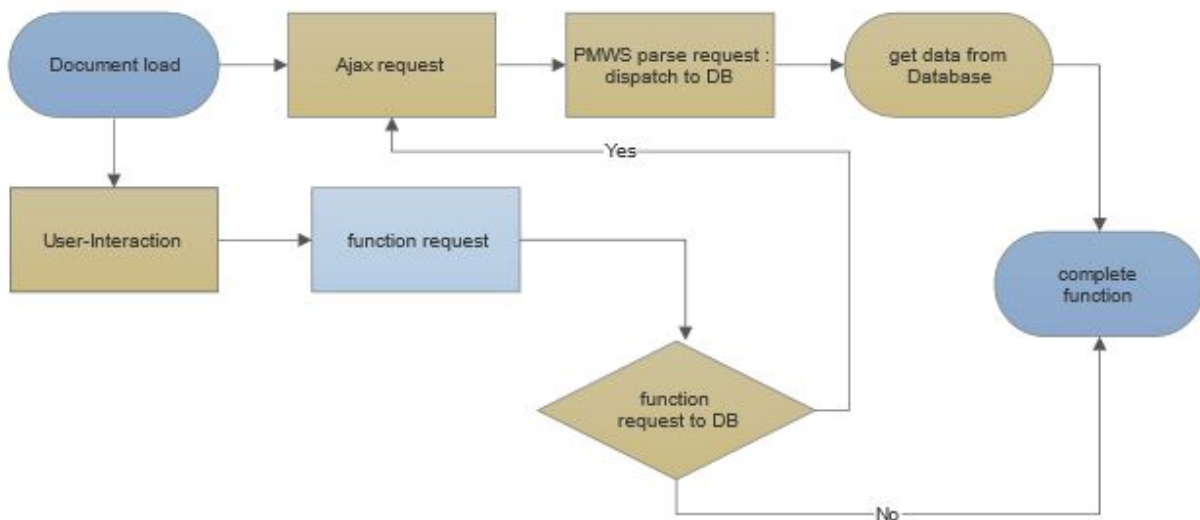
⁹ "Flat vs. Deep Website Hierarchies - Nielsen Norman Group." 10 nov.. 2013, <https://www.nngroup.com/articles/flat-vs-deep-hierarchy/>. Åpnet 9 mai. 2018.

¹⁰ "Breadcrumb Navigation Increasingly Useful - Nielsen Norman Group." 10 apr.. 2007, <https://www.nngroup.com/articles/breadcrumb-navigation-useful/>. Åpnet 9 mai. 2018.

4.2 Oppdragsgivers Rammeverk

Som nevnt tidligere har prosjektgruppen valgt å integrere løsningen med Giteks rammeverk og systemer. Som en følge av dette vil gruppen ikke kunne utvikle server-funksjonalitet eller konstruere direkte spørringer mot databasen.

Strukturen rundt er bygget opp slik at oppdragsgivers API sammenknytter gruppens programvareløsning sammen med kundens database. Dette skjer ved at programvaren sender forskjellig type ajax-baserte kall opp mot oppdragsgiver API som parser koden opp mot datauthenting fra databasen og sender den tilbake til programvaren.



Figur 21: Ajax kall

4.3 Brukergrensesnitt

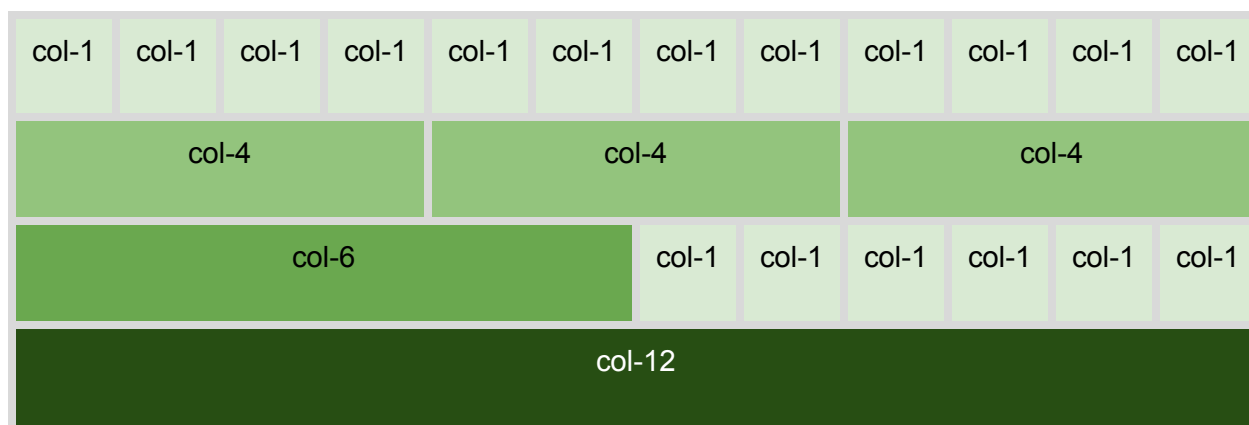
Da en av de sentrale utfordringene i oppgaven var å gjenskape en utdatert løsning i en webbløsning, var det viktig for gruppen å støtte seg på gode prinsipper innen utforming og brukeropplevelse. Dette delkapittelet prøver å argumentere for valg som er tatt i forbindelse med utseendet og dynamikk løsningen.

4.3.1 Responsivt design

Ønsket om å få til et godt responsivt design kom som en følge av oppdragsgivers ønske om tilgjengelighet på både desktop og tablet. Det er essensielt at løsningen er velfungerende på tablet, da det tillater brukeren å være mobile mens de bruker løsningen. Det var også spesifisert at et intuitivt grensesnitt var ønskelig.

For å garantere responsivitet har en rekke hjelpemidler blitt anvendt, for å slippe å bruke store mengder ressurser og tid på å lage egne biblioteker. Fremfor å lage enkeltløsninger for hver plattform, bestemte vi oss for å bruke Bootstrap. Gruppen har gjennom vårt studie blitt introdusert for Bootstrap 3.0, men valgte å anvende den nylanserte versjon 4. Dette valget blir videre argumentert i kapittel 5.2.3.1. Rammeverket tilbyr en rekke komponenter bestående av forhåndsdefinert HTML og CSS, som enkelt kan settes rett inn i løsningen. Dette sparte gruppen for å måtte egenutvikle grunnleggende deler av grensesnittet, som knapper, dropdowns og former.

Bootstrap sine komponenter blir umiddelbart enda mer attraktive når man introduserer deres responsive rammeverk. Gridsystemet deres gir muligheten til å organisere gui-deler i kolonner og rader med 12 celler i bredden. I praksis betyr dette at innholdets utseende omorganiseres avhengig av skjermstørrelsen. En rad fordeler bredden til enheten i 12, og tildeler plass som definert gjennom egne klasser. Klassene settes rett inn i HTML-elementene, og defineres enten med *row*, eller med *col* etterfulgt av ønsket antall celler som skal dekkes. Eksemplifisert under.



Figur 22: Bootstrap gridsystemet

I tillegg til gridsystemet tilbyr Bootstrap muligheten til å styre layouten basert på skjermstørrelsen. Dette gjøres med fem forhåndsdefinerte gridklasser for å tilby forskjellige opplevelser på alle skjermstørrelser. I løsningen brukte vi dette for å endre størrelsen på kolonnene, og hvordan innhold ligger i relasjon til hverandre.

Klassenavn	Breakpoint	Egnet plattform
xs (<i>extra small</i>)	0 px	Phone
sm (<i>small</i>)	576 px	Phone
md (<i>medium</i>)	768 px	Tablet
lg (<i>large</i>)	992 px	Desktop
xl (<i>extra large</i>)	1200 px	Wide desktop

Forbi Bootstrap, har gruppen forsøkt å benytte moderne prinsipper for responsivitet der det lar seg gjøre. I alle dokumentene er det benyttet HTML5 meta tags med content *width* satt til *device-width* (`<meta name="viewport" content="width=device-width, initial-scale=1.0">`). Dette hjelper på skalering, og lar innholdets bredde bli satt til enhetens bredde. Responsiv *width* bekjemper problemet med zoomet ut, og uleselig innhold på mindre enheter. Det hjelper også med presentasjon av innholdet på mobile enheter, siden visningsmodus ofte skiftes mellom portrett til landskap.

Da løsningen tar i bruk tabeller for å vise frem data er disse implementert skalerbare og håndterer de forskjellige visningsformatene. Tabellene justerer størrelsen sin ved å tilpasse bredden på hvert felt. Dersom kolonnene ikke får plass vil de bli lagt til under en rullegardin. Det vil da fremkomme et plusstegn i første kolonne som brukeren kan trykke på for at denne skal droppe ned under raden.

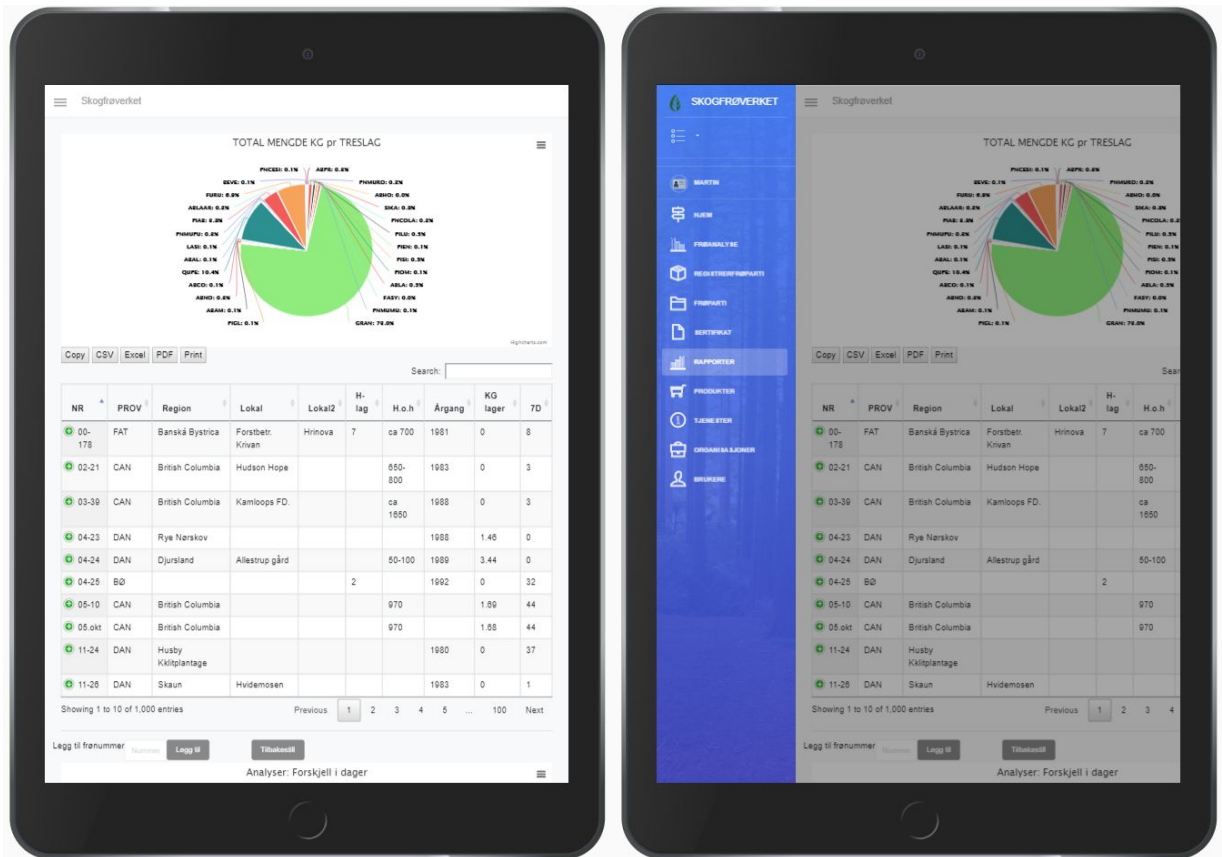
Det er også benyttet relative enheter for å videre skape skalerbarhet. Der det anses som fornuftig har gruppen benyttet *rem*-enheten istedenfor px i CSS-dokumenter. Enheten beskrives slik: *"Represents the font-size of the root element (typically <html>). When used within the root element font-size, it represents its initial value (a common browser default is 16px, but user-defined preferences may modify this)."*¹¹

En annen gevinst med anvendelse DataTables, er muligheten til å gjøre tabeller mer responsive. Dette vedlikeholdt ønsket om et velfungerende design, også på mobile plattformer. Alle kolonner som ikke får plass på nåværende oppløsning, kollapses og legges til under en ekspanderbar dropdown meny tilhørende hver rad. Dette hjalp med å passe på at informasjon aldri ble skjult for brukeren. Tabellene laget ble satt opp til å endre utforming dersom nettleseren endrer størrelse, slik at innholdet til en hver tid skal presenteres som leselig. Se mer om DataTables i kapittel 5.2.3.2.

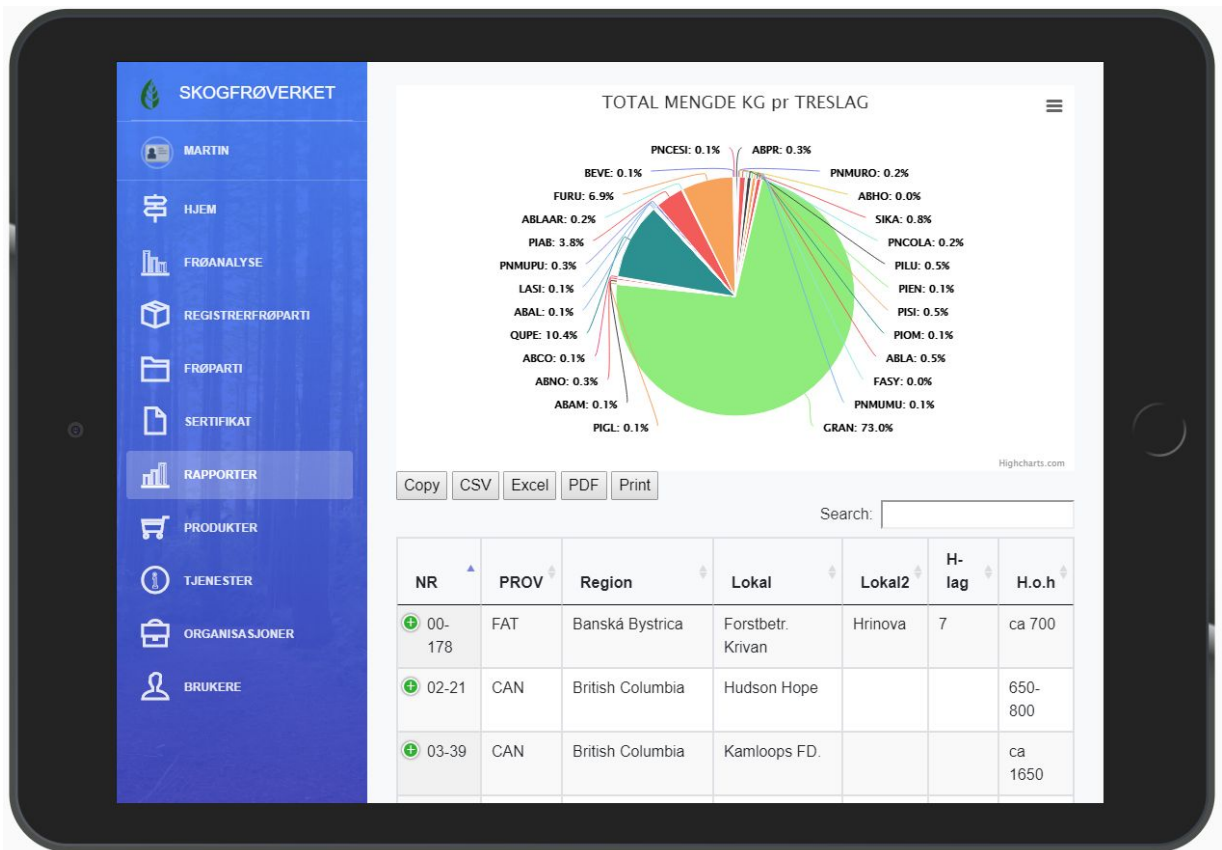
¹¹ "<length> - CSS: Cascading Style Sheets | MDN - Mozilla." 26 feb.. 2018, <https://developer.mozilla.org/en-US/docs/Web/CSS/length>. Åpnet 13 mai. 2018.

4.3.1.1 Forskjellige plattformer

Dette underkapittelet inneholder en bildeserie som viser frem eksempler på responsivitet på tablet og mobil, i portrett- og landskapsvisning.



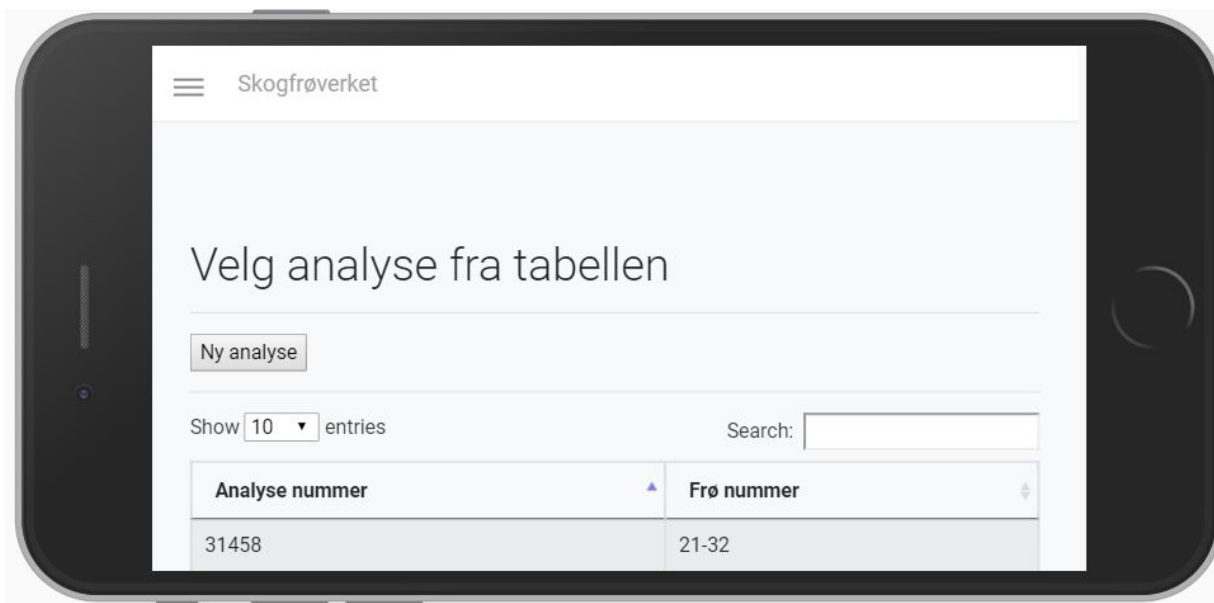
Figur 23: Eksempel tablet portrait



Figur 24: Eksempel tablet landscape



Figur 25: Eksempel mobil portraitt



Figur 26: Eksempel mobile landscape

4.3.2 Forms

I løsningen er det mange skjemaer som skal fylles ut av både interne og eksterne brukere. Å gjøre forms brukervennlige ble derfor essensielt for opplevelsen av hele web-løsningen. Vi har på dette området tatt i bruk utforming basert på best practices fra boken *Web Form Design: Filling in the Blanks*¹², skrevet av Luke Wroblewski.

Et av flere eksempler på dette er bruken av toppjusterte labels på smalere visninger, for å bedre utnytte den vertikale plassen. Her plasseres labels nærmere sitt eget felt, for å bekjempe misforståelsen om hvilket felt den faktisk tilhører. Toppjusterte labels er også assosiert med kortere utfyllingstid. Ved bredere visning, er det benyttet venstrestilte labels for å kunne utnytte horisontal plass. I dette tilfelle benyttes høyrejusterte labels for å skape tydeligere assosiasjon til tilhørende felt. Valget styrkes videre av påvist raskere utfyllingstid en venstrejusterte labels.

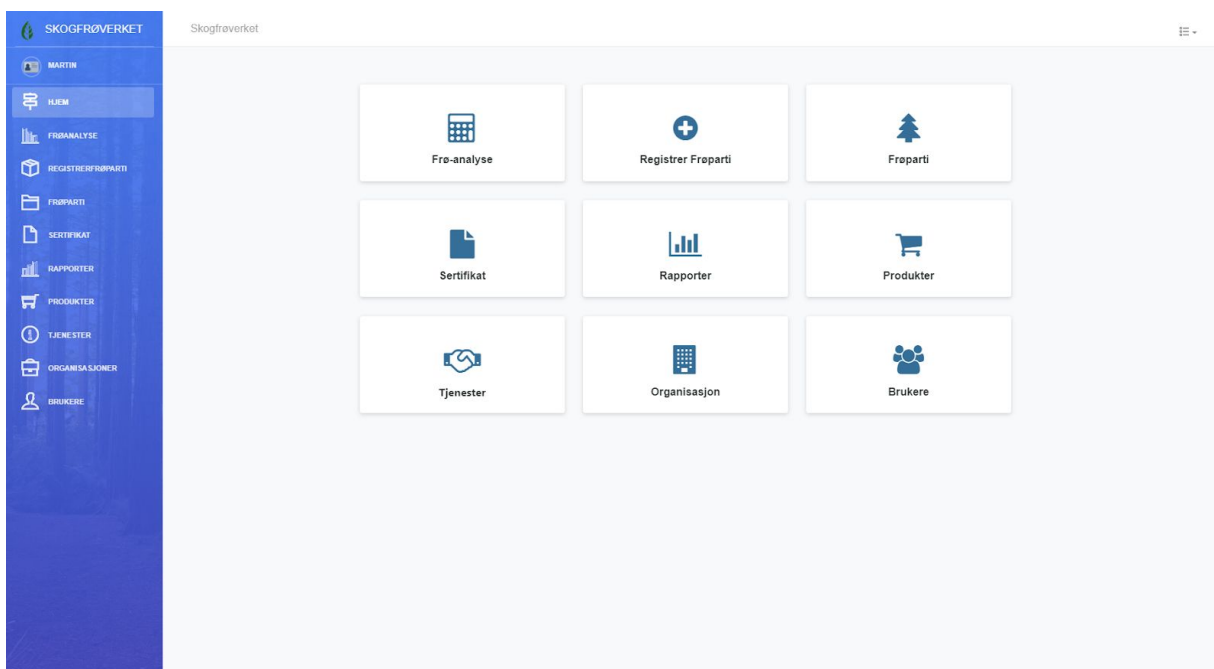
For innholdet på felter er bredden justert som en affordance, for å tydeliggjøre forventninger for brukerinput. Eksempler på dette er postnummer og telefonnummer som vanligvis har et fast antall tegn. På felter som ikke har noen naturlig størrelse benyttes nok rom til å garantere plass. Videre grupperes felter som har noen form for relasjon, og isoleres ved bruk av horisontale

¹² "LukeW | Web Form Design: Filling in the Blanks -a Web design"
https://www.lukew.com/resources/web_form_design.asp. Åpnet 30 apr.. 2018.

linjer. Eksempler på grupperinger er personlig informasjon som navn, epost og passord, eller informasjon knyttet til plassering, som adresse, postnummer og poststed.

En annen inspirasjon er boken *Forms that Work: Designing Web Forms for Usability*[8], skrevet av Caroline Jarrett og Gerry Gaffney. form og survey spesialisten Caroline Jarrett. Særlig kapittel 8 som omfatter organisering, gruppering av innhold, og ryddig bruk av grids.

4.3.3 Navigasjon og fremside



Figur 27: Fremside

Prosjektgruppen hadde som tidligere nevnt som mål å ha et brukervennlig grensesnitt og unngå for mye funksjonalitet i hvert vindu. Med et for minimalistisk design så gruppen at løsningen ville bli for steril og minne mer om en standardløsning. Gruppen har lagt ved et bakgrunnsbilde i sidemenyen av trær for å gi løsningen personlighet og knytning til skogfrøverket . Det er balansert slik at bakgrunnen ikke forstyrrer klarheten på menyteksten, men som fortsatt er synlig for brukeren.

Som nevnt tidligere har løsningen åtte kategorier som løsningen omfatter. Disse kategoriene er representert med ni bokser hvor den siste boksen ikke er lenket til noen funksjonalitet og representerer den manglende kategorien.

Donald Norman nevner “signifiers”, “affordance”, “discoverability” og “feedback” som viktige prinsipper når det kommer til brukerens interaksjon med et system [9]. “Affordance” er

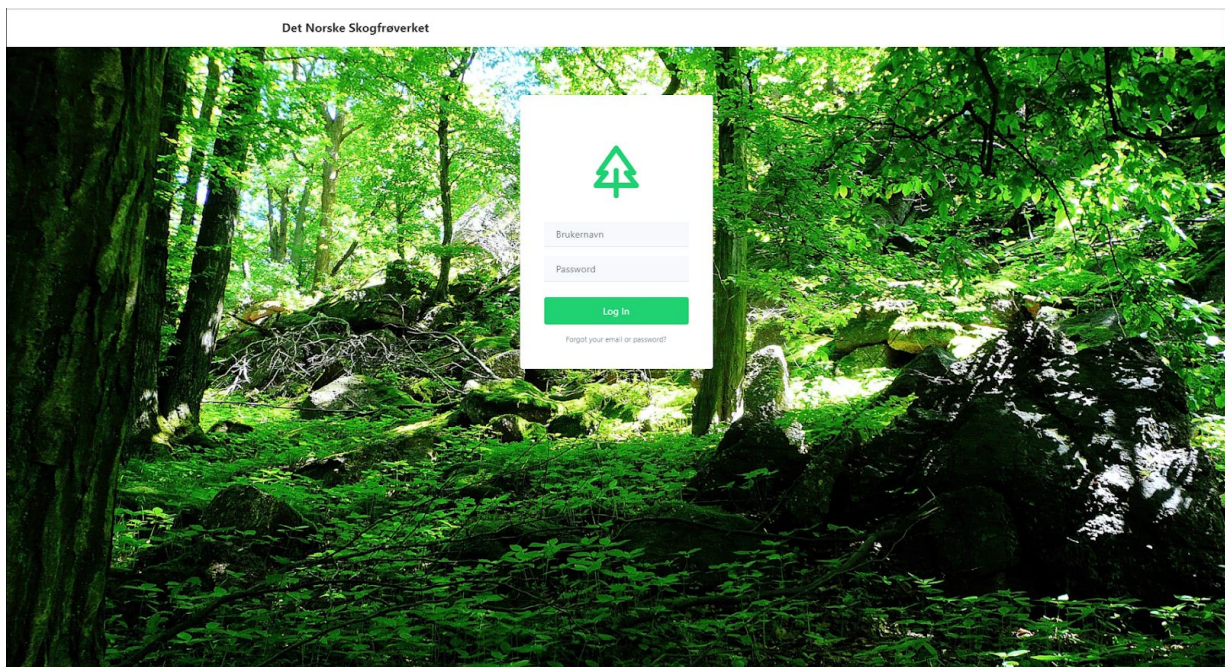
muligheten til å utføre en handling og “discoverability” er muligheten for å oppdage “affordances”. “Signifiers” hjelper brukeren med å oppdage funksjonalitet, og “feedback” gir brukeren signal om at dens interaksjon er registrert og blir behandlet av systemet. Prosjektgruppen har valgt å legge vekt på disse prinsippene ved utformingen av designet. På forsiden ble det lagt på ikoner som samsvarer med titlene på menyvalgene, og som skal bidra til å heve designet, og dermed gjøre det enklere for brukere få øye på ønsket funksjonalitet. Boksene har en skyggeeffekt for å gi inntrykk av at de kan trykkes på, musepeker forandres innenfor rammene til boksen og selve boksen skifter farge for å hjelpe brukeren å oppdage boksens “affordance”.



Figur 28: Eksempel på affordance

4.3.4 Login

Når prosjektgruppen implementerte login funksjonalitet ville vi at designvalget på login siden skulle reflektere temaet til organisasjonen som er skog og natur. Derfor var det stor enighet om å velge noe med grønne fargetoner.



Figur 29: Login siden

Vi forsøkte å finne en fargenyanse, som også gikk i grønn på fremsiden. Etter å ha testet ut flere forskjellige fargetoner, deriblant blå, sort, grønn, gul og rød fant vi ut at den blå fargepaletten passet best på startsiden, og den virket å passe bedre sammen forskjellige komponenter. Prosjektgruppen så også at dette samsvarte mer med en *enterprise*-løsning.

4.3.5 CRM-modulen

I CRM-modulen har vi delt opp designet i to deler. Den første delen er en brukertabell for administrasjon hvor internbrukere kan inspisere og redigere brukerinformasjon, og deaktivere brukerkontoer etter behov. Tabellen kommer med et responsivt søkefelt hvor en lett kan søke frem personer.

Skogfrøverket

Brukeradministrasjon

Show 10 entries

#	Fornavn	Ettennavn	organisasjon	Telefon	emailAdr	Adresse	active	valg	bruker
1	fornavn	etternavn	org	tlf	epost		enabled	Valg	deaktiv
2	fornavn	etternavn	org	tlf	epost		enabled	Valg	deaktiv
3	fornavn	etternavn	org		epost		enabled	Valg	deaktiv
4	fornavn	etternavn	org		epost		enabled	Valg	deaktiv
5	fornavn	etternavn	org	tlf	epost		enabled	Valg	deaktiv
6	fornavn	etternavn	org		epost		enabled	Valg	deaktiv
7	fornavn	etternavn	org		epost		enabled	Valg	deaktiv
8	fornavn	etternavn	org	tlf	epost		enabled	Valg	deaktiv
9	fornavn	etternavn	org	tlf	epost		enabled	Valg	deaktiv
10	fornavn	etternavn	org	tlf	epost		enabled	Valg	deaktiv

Showing 1 to 10 of 664 entries

Previous 1 2 3 4 5 ... 67 Next

Figur 30: Brukeradministrasjon

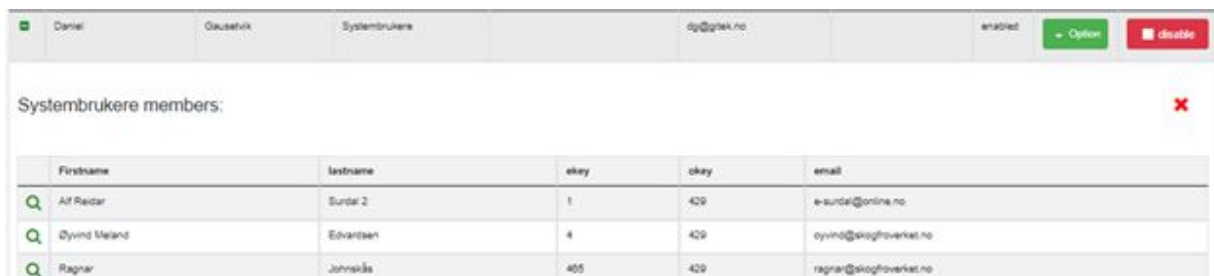
Ved å trykke på “rediger bruker” bruker vil det bli fremvist en redigerings-modal med utvidet brukerinformasjon. Prosjektgruppen, eksperimenterte med forskjellige alternativer for redigering av forskjellige datasett. Etter konsultasjon med oppdragsgiver, presiserte de ønsket om at gruppen skulle benytte seg av modal for redigering av brukerne.

Ytterst til høyre finner vi kolonnen “Bruker”. Denne knappen viser enten “deaktiver” eller “aktiver”. Alle aktive bruker får knappen “deaktiver”, som gir muligheten å deaktivere kontoen. Alle deaktiverte kontoer får knappen “aktiver”, som lar en administrator aktivere kontoen.



Figur 31: Detaljert brukerinformasjon

Ved valg av “user-organisation” vil det gli ned et fremvisningsfelt mellom radene. Dette viser alle medlemmene som tilhører samme organisasjon. Disse er lagt på “click-Listener” slik at ved å trykke på én av medlemmene, vil de resterende medlemmene også dukke opp i redigerings-modalen.



Figur 32: Visning av medlemmene i en organisasjon

Disse funksjonene kunne blitt spredt over flere sider, hvor eksempelvis redigering av en bruker ville tatt brukeren til en ny side hvor operasjonen kunne utføres. Dette var noe som ble vurdert flere steder. Det ville gitt brukerne opplevelsen av en programvare med mer volum. Argumentet imot var ønsket om kraftfulle tabeller, med gruppering av relatert innhold. Videre kan

forandringer opp mot databasen skje raskt uten at brukeren må navigere seg frem og tilbake mellom de forskjellige sidene.

Ved bruk av DataTables, et plug-in til jQuery, kan tabellene å bli sortert i de forskjellige kolonnene ved å trykke på dem. Man kan velge antall rad-fremvisninger per side, og søke gjennom tabellen etter regulære uttrykk.

På den andre delen av CRM-modulen finner vi innlogging og utlogging, samt personlige profilsider hvor brukere kan skaffe seg oversikt over egen brukerinformasjon, redigere brukeren, eller se på ordre og ordrehistorikk. Brukerprofilen er delt opp i tre faner for å få en ryddig og oversiktlig navigasjon på siden. Tab menyen består av "profil", "redigere" og "ordre".

Den siste fanen for ordreinformasjon endte opp med å være tom i siste versjon av prosjektet, grunnet manglende datasett i databasen. Se kapittel 7.1.


The screenshot shows a user profile page in a CRM system. The page is titled "bruker info" and "Organisasjon - Systembrukere". It features a user profile for Håvard Myrvold Johannessen, including a profile picture, name, and contact information. To the right, there is a table listing other system users.

Fornavn	Etternavn	organisasjon	Telefon	emailAdr	Adresse
Batmann	BruceWayn	Systembrukere	9977777	e-JK@online.no	nullsvingen 0
Supermann	Kent	Systembrukere		bais@gitek.no	
Håvard Myrvold	Johannessen	Systembrukere	34556734	kongen@gitek.no	Øgardsbakka 38A
JanTeigen	Pålåven	Systembrukere		fanBOY@genion.de	Skogv 6
Joker	JK	Systembrukere		kvn@gitek.no	
Henrik	Wergeland	Systembrukere	92 34 00 93	rggar@skogfroverket.no	
IronMan	Stark	Systembrukere		tttt@gitek.no	
Hulken	Banner	Systembrukere		Mestern@skogfroverket.no	
Justin	Bieber	Systembrukere		tullmail@skogfroverket.no	

Figur 33: Brukerprofil

I bruker-redigeringen satte prosjektgruppen opp et forstørret "card-view" hvor brukeren får opp informasjonen som kan redigeres, og hvor det fremkommer mulighet for å legge til eller bytte profilbilde. Etter konsultasjon med oppdragsgiver skulle ikke eksterne brukere ha mulighet å redigere noen andre personlige data enn profilbilde. Derfor ble det deaktivert i html strukturen, og lagt til en sjekk på tilgangsnivå for å kontrollere om brukeren har rettigheter til å endre selv, for så å åpne opp hvis dette er tilfelle.

Rediger Profil



bytt bilde angre

ORG / PLANTESKOLE BRUKERNAVN EMAIL ADRESSE

Org / Planteskole hajo mi@gitek.no

FORNAVN ETTERNAVN

Håvard Myrvold Johannessen

ADRESSE

Ølgardsbakka 38A

STED NASJONALITET POSTKODE

Gran Norsk 2750

TLF-NUMMER

95034807

Lagre Innhold

Figur 34: Rediger profil

Den en siste delen av CRM modulen handler om administrering av planteskoler / organisasjoner. Fordi datasettet på de forskjellige tabellene består av veldig mange rader, så har vi valgt å bruke tabeller. Datagrunnlaget for tabellene som brukes gjennom løsningen er forskjellig, både i omfang og relevans, og er satt opp slik at de best mulig skal kunne gi brukeren en god oversikt over det relevante datasettet.

Skogfrøverket

Organisasjoner

Show [10] entries

Search:

Organisasjon	orgkode	email	address	antallansatte
Systembrukere	SYS	oyvind@skogfroverket.no	Skogfrøverket	10

[add user to organisation](#) | [remove user from organisation](#)

Employee

navn	etternavn	mobil	adresse	email
fornavn	etternavn	ttf		epost
fornavn	etternavn			epost
fornavn	etternavn	ttf		epost
fornavn	etternavn			epost
fornavn	etternavn	ttf		epost
Khai	Ngo			
Martin	Harestad			
Martin	Stenen			mastene@stud.ntnu.no
Vette	Lundsvoll			
Håvard	Johannessen			

Organisation Info

Orders

Organisasjon	orgkode	email	address	antallansatte
--------------	---------	-------	---------	---------------

Showing 1 to 1 of 1 entries (filtered from 526 total entries)

Previous Next

Figur 35: Organisasjonstabell

Organisasjonstabellen er bygd opp av fem kolonner, som viser navn, e-post, adresse og antall ansatte registrert ved bedriften. Til venstre i tabellen, på hver rad, finner vi et ekspansjons ikon, som informerer brukeren at han kan trykke på ikonet for å få utvidet informasjon.

Prosjektgruppen erfarte at det ville være lettere for brukeren dersom han kunne trykke hvor som helst på raden for å få frem utvidet informasjon om organisasjonen.

Strukturen rundt en organisasjon som kan inngå i det utvidede vinduet kan deles opp i fire deler:

1. Detaljert informasjon om organisasjonen
2. Alle de ansatte som jobber i organisasjonen
3. Ordre og ordrehistorikk som organisasjonen har rundt kjøpt av frø
4. Legge til og fjerne brukere fra organisasjonen

De tre første delene, som vil ha mest bruk, ble lagt inn i en “accordion” for å samle denne funksjonaliteten. Det siste elementet som innebærer å legge til eller fjerne brukere fra en organisasjon ble lagt til som en ekstra knapp oppe i venstrehjørnet rett under organisasjon kolonnen. Ved å trykke på en rad dropper det ned et utvidet informasjonsfelt som er pakket rundt “accordion”-elementet og har tre faner. Den første fanen viser de ansatte som er oppført i organisasjonen. Den andre fanen viser utvidet informasjon om bedriften, hvor det også er lagt til editerings muligheter. Den tredje og siste fanen viser aktive ordre og ordrehistorikken fra perioden før salg.

Når en rad blir trykket på blir raden raden i en mørk fargetone for å vise til at den er aktiv. Det er også lagt til actionlistener på ansatt radene i undertabellen slik at de også kan redigeres her etter behov. Dette ble først lagt inn i en trestruktur som ble ekspanderte inn. Dette fant oppdragsgiver mindre oversiktlig, og vi valgte derfor å gå over til at redigeringsmuligheter vises frem i en “modal”, på samme måte som i brukertabellen.

The screenshot shows the 'Organisasjoner' page in the Skogfrøverket system. The page has a blue sidebar with navigation options: MARTIN, HJEM, NYHETER, FRØANALYSE, REGISTRERT FORPARTI, FORPARTI, SERTIFIKAT, RAPPORTER, PRODUKTER, TJENESTER, ORGANISASJONER, and BRUKERE. The main content area is titled 'Organisasjoner' and shows a table of organizations. The table has columns for 'Organisasjon', 'orgkode', 'email', 'address', and 'antallansatte'. The first row is 'Systembrukere' with orgkode 'SYS', email 'oyvind@skogfroverket.no', address 'Skogfrøverket', and antallansatte '10'. Below the table, there are two buttons: 'add user to organisation' and 'remove user from organisation'. Below the buttons, there is a section for 'Employee' and 'Organisation Info'. The 'Employee' section shows a table with columns for 'empcount' (10) and 'fax'. The 'Organisation Info' section shows a table with columns for 'level' (0), 'mailaddress' (Skogfrøverket), 'mailpostaladdress' (Hamar), 'mailpostalcode' (2301), 'notes', 'orgcode' (SYS), 'orgleader', 'orgleaderkey' (0), 'orgname' (Systembrukere), 'orgstatus' (1), 'phone' (tlf), 'stamp' (2008-12-16T15:07:33.000), 'treepath' (Systembrukere), and 'webaddress'.

Figur 36: Organisasjonsinfo

Fanene er designet slik at hvis du trykker på en uåpnet fane, vil den allerede åpnede lukke seg i en sammenhengende animasjon. Dette valget ble gjort ut i fra vurderingen om at det er viktig å holde oversikten mest mulig ryddig, og unngå å overvelde brukeren. Det er lagt til “signifiers” i form av piler på alle fanene, som roterer 180 grader ved åpning og lukking av fanene, for å hjelpe illustrere hvilken fane som er aktiv.



Figur 37: Signifiers for åpning og lukking av faner

4.3.6 Produktregister-modulen

Ved flere av modulene valgte gruppen å gå for gjenkjennelighet når det kommer til designet og utformingen av løsningen. I prosjektets planleggingsfase fikk gruppen tilgang til den gamle IT-løsningen kunden benytter seg av. På grunn av den konfidensielle avtalen som prosjektgruppen har underskrevet har vi ikke tillatelse til å vise bilder av denne IT-løsningen, i den hensikt å vise til similaritet av designet.

Det å registrere nytt frøparti har blitt løst med et fire stegs utfyllingsskjema. Valget å ha flere trinn i formen, ble gjort fordi antall felter som skulle fylles ut ble overveldende. Det starter med å velge et treslag fra databasen som befinner seg i denne kategorien. Deretter må brukeren fylle inn land, region, lokasjon og høydemeter som frøpartiet kommer fra.

Registrer Frøparti:

Generell informasjon

Velg treslag... ▾

Land

Land

Region

Region

Lokasjon

Lokasjon

Høydemeter

Høydemeter

1 2 3 4 Next

Figur 38: Bilde fra registrerings formen for Frøpartier

Ikke alle felt i skjemaet må være utfyllt før brukeren kan gå videre i prosessen. Enkelte av feltene er opsjonale grunnet hvordan vi har fått beskrevet registreringsprosessen. Det er vanlig at ikke alle data er tilgjengelig ved første registrering, og at informasjon fylles inn i etterkant. Det er lagt inn et sjekkpunkt på hvert obligatoriske felt slik at man ikke kan gå videre uten at de er utfyllt. Dersom man likevel forsøker å gå videre vil feltene lyse opp i rødt og skjemaet forblir stående på det samme steget.

Brukerne av systemet kan registrere og vedlikeholde frøpartier, registrere spireevne og historikk, registrere og vedlikeholde geografisk lokasjon til frøpartier. Under planleggingsfasen til prosjektgruppen ble det sett på hvordan dette kunne struktureres best mulig. Dette ble gjort i

samråd med oppdragsgiver, slik at arkitektur og struktur skulle være i trå med oppdragsgiver ønske og forventninger.

Etter å ha analysert de forskjellige modulene så prosjektgruppen det mest hensiktsmessig å flytte registrering av spireevne og historikk inn i analyse-modulen, og flytte registrering av frøparti inn i registrerings-modulen, som opprinnelig kun var bestående av å laste inn eksisterende sertifikater og gamle vedlegg.

Med dette vedtaket stod prosjektgruppen med nåværende struktur rundt løsningen, og produktregister-modulen stod igjen med oppgaven å gi overblikk i produktsortimentet, filtrere, og redigere frøpartiene i databasen.

Den gamle løsningen er strukturert slik at produktfremvisning er delt inn i sider som brukeren kan navigere seg mellom, og for hver side vises kun ét frøparti. Dette gir brukeren ingen form for overblikk i det relativt store datasettet. Dette var en område med stort forbedringspotensiale, som gruppen ønsket å forbedre i den nye løsningen.

Et av de mest essensielle elementene etter å ha analysert kravspesifikasjonen gitt av oppdragsgiver, var å strukturere datasettene gjennom flittig bruk av tabeller. Siden fagfeltet var relativt ukjent for gruppen, ble det valgt å ikke implementere statiske filtrere. Vi er ikke sikre på hvilke faktorer som er interessante for brukerne i valg av frøparti, og lot heller brukerne definere egne filtre.

Det ble brukt slidere for tallverdier det virket naturlig å ha en øvre og nedre grense på. Siden det var usikkert hva som var naturlige øvre grense for verdier som tusenkornvekt, og det fantes vanninnhold i databasen over 100%, ble det satt på premisset "og over" for maksverdiene. Det ble lagt inn at hver gang det skjer forandringer i sidemenyen oppdateres tabellen slik at data som fremvises korresponderer med valgte filtre.

The image shows two side-by-side panels of a software interface. The left panel contains three sliders for numerical values:

- Treslag:** A dropdown menu with the text "Velg treslag...".
- Minimum antall kilo:** A text input field containing "kg".
- Renhet: 0% - 100% og over:** A slider bar.
- Vanninnhold: 0% - 100% og over:** A slider bar.
- Tusenkorvekt: 0g - 1000g og over:** A slider bar.

The right panel shows a dropdown menu for tree species:

- Treslag:** A dropdown menu with the text "Velg treslag...".
- A list of tree species: 123, Ajangran, Alaskapoppel, Alaskasyppress, Alm, Ask, Balsamedelgran, Banksfuru, Barlind, Bergfuru, Blågran, Blågran (sølvgran), Brewergran (Slørgran), Buksbum, Buskfuru var. frutescens, Bøk, Cembrafuru, Coloradoedelgran, and Contortafuru (innl.form).

Figur 39: Sliders for tallverdier

Treslag: Show 10 entries Search:

Velg treslag...

Minimum antall kilo:
kg

Renhet: 3% - 100% og over

Vanninnhold:
0% - 100% og over

Tusenkorvekt:
0g - 1000g og over

#	Norsk navn	Kg lager	Renhet	Vanninnhold	Tusenkorvekt	Rediger	Slett
	Gran (importert)	48.6	99.2	7.3	7.01	rediger	slett
	Gran (importert)	89.54	97.8	5.7	6.7	rediger	slett
	Gran (importert)	55	98.7	5.6	8.77	rediger	slett
	Gran (importert)	10.02	99.1	4.8	8.68	rediger	slett
	Gran (importert)	22.99	99.3	6.1	6.53	rediger	slett
	Gran (importert)	7.7	99	7.6	7.77	rediger	slett
	Gran (importert)	100.96	99	5.1	7.12	rediger	slett
	Gran (importert)	19.98	99	4.6	6.91	rediger	slett

Figur 40: Sliders for tallverdier 2

4.3.7 Analyse-modulen

I analyse-modulen ble det sett på flere måter å presentere analyser utført på frøpartier til brukerne av systemet. Et av designmålene var å skape en fungerende balanse mellom moderne prinsipper, og samtidig beholde gjenkjennbarhet. Den gamle metoden for å gjennomføre analyser, var å manuelt fylle inn analyse-kort i papir som var blitt utskrevet. Den nye løsningen skulle være en web-basert portal, tilgjengelig for de ansatte til å kunne logge seg inn og utføre prosessen digitalt.

Siden det papirformaterte analysekortet var en såpass etablert ressurs i oppgaven, valgte gruppen å kopiere av det gamle designet på over til den nye løsningen. Ved å navigere seg inn på analyse via navigasjonsmenyen blir brukeren av systemet møtt med to valg. Det første valget er å opprette et nytt analyse-kort. Dette valget sender brukeren videre til analysekortene som er delt opp i et fler-trinns -utfyllingsform, etter samme prinsipper som produktregistreringen. Her er det særlig tydelig at det er valgt gjenkjennbarhet over det gruppen vurderer som best practise. Hver av trinnene har et oppsett som er en kopi av et segment fra analysekortet. Den andre valgmuligheten brukeren har er å oppdatere et gammelt analysesett. Da velger brukeren et frø- og analysenummer fra tabellen som korresponderer med det analysesettet som skal oppdateres.

Velg analyse fra tabellen

Ny analyse

Show 10 entries Search:

Analyse nummer	Frø nummer
31458	21-32
31523	28-17
32700	41-10
33051	21-32
34012	16-30
34348	218-11
34349	218-11
34369	5776
34635	27-30
34867	27-32

Showing 1 to 10 of 100 entries Previous 1 2 3 4 5 ... 10 Next

Figur 41: valg av analyse

I forbindelse med oppretting av en ny analyseform vil alle feltene være tomme, og alt må fylles inn på nytt i motsetning til det å bare redigere en eksisterende analyse som har behov for å bli oppdatert på ett eller flere punkter. Dersom man velger å bare redigere en gammel analyseform vil hele datasettet som tidligere har blitt fylt inn, bli lastet inn i feltene i analyseskjemaet.

Analysekort

Frøparti nummer

Treslag

Analysenummer

Provins

Årgang

Opprinnelse

Garantianalyse
 Vegledningsanalyse

Spirehastighet	Spireevne	Renhet %	Tusenkornerket	Frø pr. KG	Vanninnhold
Spirehastighet (%)	Spireevne (%)	98.2	26.7	36780	6.4

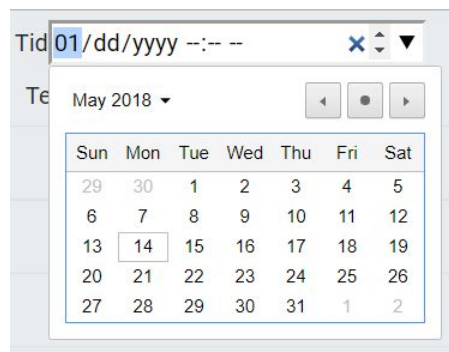
● ● ● ● ● ●

[Neste](#)

Figur 42: Analysekort

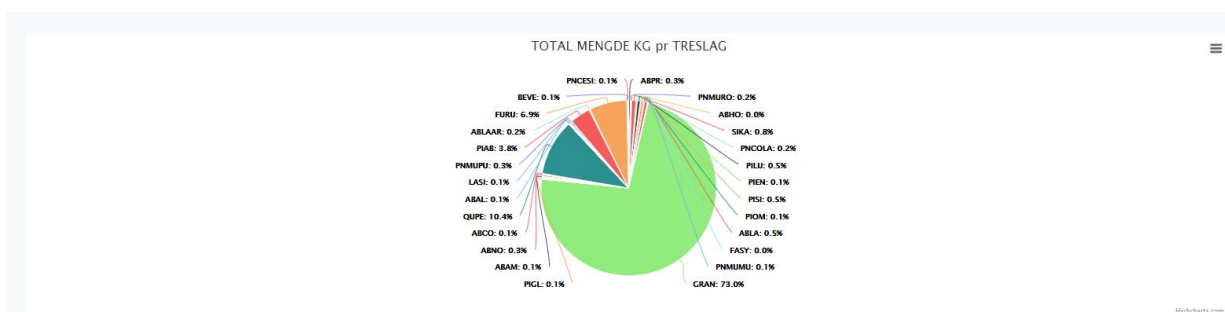
Gruppen undersøkte den gamle løsningen kunden brukte på å løse forskjellige arbeidsoppgaver, som registrering av data. Prosessen består delvis av en IT-løsning, kombinert med notatblokker, og utskrevne analysekort. Det ble kartlagt alt av datagrunnlag som må bli tatt med i den nye løsningen for å gi brukerne av løsningen tilstrekkelig med informasjon. Formen er delt opp i flere steg hvor hvert steg inneholder informasjon som blir fylt ut samtidig.

Ved valg av dato er dette designet slik at det vil dukke opp en kalender for input som formaterer dato, uten at brukeren trenger å tenke på dette. Det bidrar til å skape robuste input områder som sender inn riktig format og som databasen aksepterer.



Figur 43: valg av dato

4.3.8 Rapport-modulen



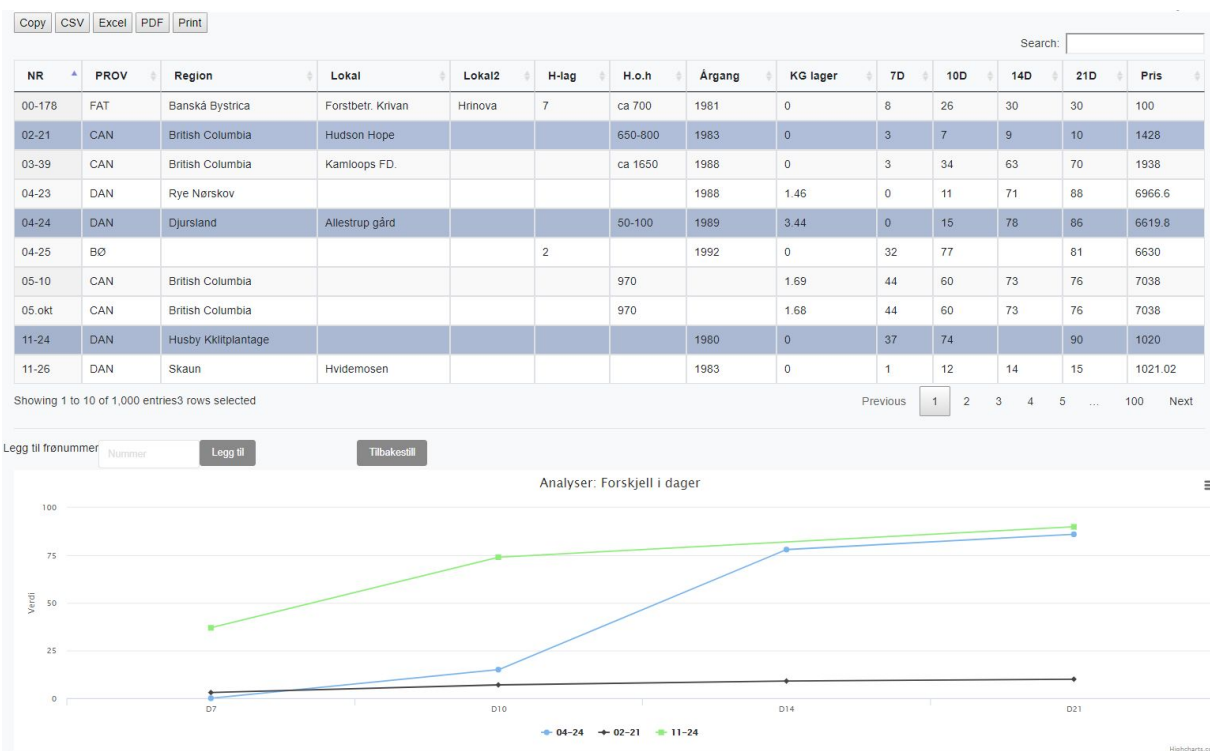
Figur 44: Rapport illustrasjon 1

På toppen er det et interaktivt sektordiagram som ved berøring registrerer den delen brukeren holder musepekeren over, og "highlighter" den. Diagrammet viser brukerne fordelingen av de

forskjellige treslagene som finnes i databasen. Vi kan eksempelvis se at sortimentet til organisasjonen er majoriteten av frøpartiene gran, det utgjør 72 prosent av alle varene på lager.

Det er implementerings knapper som eksportere informasjonen som fremvises. Med disse menyvalgene kan brukeren kopiere dataen, generere CSV-formatert fil, Excel-formatert fil, PDF-formatert fil eller skrive data direkte ut. Dette gir organisasjonen mulighet til å overføre data til andre avdelinger eller eksterne aktører uten noen form for direkte integrasjon til systemet. Det er designet slik at dersom brukeren filtrerer søket på forhånd, vil kun de egnede resultatene bli med ved en eksportering.

Videre ned på siden finner brukeren av programvaren den dynamisk graf. Denne er ved lasting av siden tom. Ved å trykke på en av radene som består av et frøparti, vil raden markeres, og deretter vil grafen tegnes opp for brukeren i samtid.



Figur 45: Rapport illustrasjon 2

Brukeren kan da holde musepekeren over punktene på grafen, og får opp da en "tool-tip" boks som viser id til grafen og verdien grafen har ved punktet. Grafen viser data om spireevnen til forskjellige frøpartier. Det vil si antall spirede frø partiet produserer etter fire forskjellige konstanter, på 7-, 10-, 14-, og 21-dager. Dette forteller frø-edlere reproduserbarheten til frøpartiet. Det ble sett på forskjellige typer statistikk som kunne legges til. Rapporter som


baserer seg på datagrunnlag fra ordremodulen ble ikke implementert da ordremodulen ikke er implementert.

Over grafen er det alternativer for manuelt skrive inn frøpartinummer som skal fylles inn i datasettet, og for å tilbake stille grafen. Ved å trykke tilbake still, fjernes datasettet til grafen og de markerte radene i tabelloversikten.

4.3.9 Ordre-modulen.

Ved kjøp av et frøparti følger det med et sertifikat som beskriver aspekter ved frøpartiet. Dette sertifikatet inneholder mye informasjon som trekkes fra forskjellige datasett. Sertifikatet har som hensikt å bli sendt sammen med en bekreftelsesordre kunder får via e-post, etter å ha gjort et kjøp via løsningen. Selv om sertifikat var planlagt å gå under produktregister-modulen måtte gruppen endre dette til ordre-modulen under implementasjon da det ble klart at sertifikatet kommer som en følge av en ordre.

I den gamle løsningen blir det generert et sertifikat og oppdragsgiver ønske var at designet på denne modulen skulle gjenspeile dette sertifikat, derfor var alle designvalg rundt dette punktet fastsatt fra starten.



SKOGFRØVERKET
Stiftelsen Det norske Skogfrøverk
Boks 118, N-2301 HAMAR

Stambrev

(FRØSERTIFIKAT)

Ordre-nr: B17-002

Mottager Trysil planteskole A/S		Frøparti (sertifikat) nr F06-037	Mengde kg 6.000
Treslag Picea Abies		Proveniensa/Sankeområde FPL/S Sanderud	Årgang 2006
Opprinnelse	Kategori	Frøklode	
STEDEGEN <input type="checkbox"/>	LOKALITETSBESTEMT <input type="checkbox"/>	BESTAND <input checked="" type="checkbox"/>	
UVISS <input checked="" type="checkbox"/>	UTVALG <input type="checkbox"/>	UTVALGT BESTAND <input type="checkbox"/>	
IKKE STEDEGEN <input type="checkbox"/>	KVALIFISERT <input checked="" type="checkbox"/>	FRØPLANTASJE <input type="checkbox"/>	
	TESTET <input type="checkbox"/>	FPL Åvisprogram <input type="checkbox"/>	
		KLON <input type="checkbox"/>	
		PARKRYSNING <input type="checkbox"/>	

Lokaliteter innen frøparti							(Desimalgrader)		Innsamlingssted (ved frøimport)	
Prov.	H.lag	Kommune/FPL	Beskrivelse	Best nr.	Bredde	Lengde	H.o.h	Land :		
FLP	S	Sanderud	FP 1114		60,29		218	Stat/Distrikt :		
FLP	S	Sanderud	FP 1114		60,29		218	Sankeområde :		
								Lokalitet :		
								Breddegrad :		
								Lengdegrad :		
								H. o. h		

ANALYSERESULTATER												
Spirehastighet		Spirt	Spireevne		Friske ikke spirt		Abnorme spirer	Døde matede	Tomme frø	Antall levende frø pr kg	Analysen avsluttet (dato)	Analyse nr.
7d	10d	14d	21d	28d	%	%	%	%	%			
57	86	98	98					2		188500	21.12.2015	47041 A

Merknader til analysene

A:	Test text1	Test text2	Test text3
B:	Test text1	Test text2	Test text3

Renhet (%)	Tusenkomvekt (g)	Antall frø pr. kg	vanninnhold (%)
99.9	5.30	188500	7.2

Andre opplysninger :

22.02.2017
Date

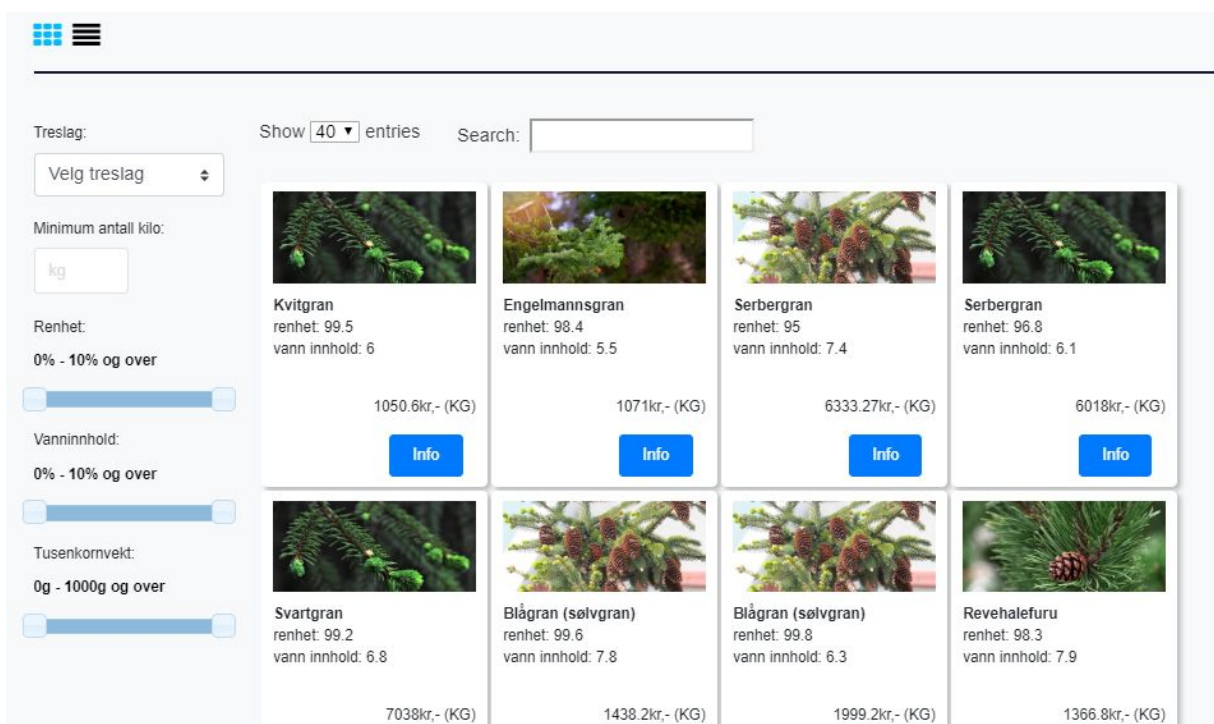
Underskrift

Figur 46: Frøsertifikat

Som nevnt tidligere har gruppen støtt på problemer med ordre-modulen som har ført til at denne ikke har blitt ferdig implementert. Et valg gruppen gjorde rundt ordre-modulen var at ved innlogging brukeren bli sendt til siden for interne brukere eller en web-shop, avhengig av brukertype. Prosjektgruppen hadde i produkt planleggingsfasen fått innblikk i oppdragsgivers

API, der vi fant ut at det eksisterte funksjonalitet som håndterer produktbestilling med, fakturering og ordredata. Vi så det som hensiktsmessig å integrere dette inn i kjøp for å fullføre salgsprosesser.

Selv om denne modulen ikke ble ferdigstilt før produksjonsfasen nådde slutten, ble det utviklet produkt fremvisning, slik gruppen ønsket å fremvise det for kundene i webportalen.



Figur 47: frontend for ordre modulen 1

Fremvisning av produktene har tatt inspirasjon på utformingen fra andre veletablerte nettbutikker der produktene er formet som kort med informasjon og bilde. Det er også lagt på en info knapp som skal ta brukeren videre til frøpartiet sin produktside, med mer detaljert informasjon og bestillingsmuligheter.

I ytterste kolonne til venstre er det lagt til filterings muligheter som er designet på lik måte som i produktregister-modulen, slik at kundebasen til organisasjonen skal få like filtreringsmuligheter som intern brukerne. Resultatene som vises er filterert for å sikre at frøpartiet har oppført pris og fortsatt finnes på lager. Øverst i venstre hjørne er det lagt til to fremvisningsikoner slik at brukeren kan velge hvordan produktene skal bli presentert.

Treslag: Show 40 entries Search:





Velg treslag

Minimum antall kilo:

Renhet: 0% - 10% og over

Vanninnhold: 0% - 10% og over

Tusenkorntekt: 0g - 1000g og over

	Kvitgran renhet: 99.5 vann innhold: 6	1050.6kr,- (KG)	Info
	Engelmannsgran renhet: 98.4 vann innhold: 5.5	1071kr,- (KG)	Info
	Serbergran renhet: 95 vann innhold: 7.4	6333.27kr,- (KG)	Info
	Serbergran renhet: 96.8 vann innhold: 6.1	6018kr,- (KG)	

Figur 48: frontend for ordre modulen 2

Ved å trykke på det andre ikonet vil sidefremvisningen bli endret til linjerte rader. Slik at kunden kan etter preferanse kan velge hvordan produktene presenteres. Dette har som mål å øke brukeropplevelsen ved anvendelse av siden.

5. Implementasjon og koding

Dette kapittelet beskriver gruppens utviklingsmiljø, og programmering av løsningen som er beskrevet i de tidligere kapitlene. Kapittelet kommer til å gå inn på hvilke redskaper gruppen brukte, programmering for web, og bruken av oppdragsgivers moduler som ble integrert inn i applikasjonen.

5.1 Utviklingsmiljø

Det er lett å miste oversikten i et voksende prosjekt. For å kunne effektivt lage en god applikasjon så var det viktig at gruppen satte opp et solid utviklingsmiljø som kunne ligge i bunnen av alt og støtte utviklere i løpet av utviklingsprosessen.

5.1.1 Server

Fra starten av prosjektet jobbet gruppen på egne maskiner i isolasjon. Den originale planen var at gruppen skulle jobbe opp imot en av Gitek sine servere, som avgjort etter oppstartsmøte i januar. Hensikten var å programmere lokalt, og sette opp et produksjonsmiljø på serveren. Oppdragsgiver ble nødt til å sette opp et eget servermiljø for oss, og vi måtte vente på dette før planen ble realisert.

Det ble klart at vi ikke hadde mulighet til å benytte oss av oppdragsgivers biblioteker på egne maskiner, og at disse var avhengige av serveren. Vi hadde heller ikke tilgang til databasen i perioden før servermiljøet var klart. Siden dette ble en større hindring for progresjon, jobbet gruppen heller for å utbedre et godt brukergrensesnitt i denne perioden. Vi fikk tilgang til en ferdig konfigurert server i uke 7 (12.- 18 februar).

Serveren var satt opp som et eget testmiljø som replikerte Gitek's egentlige servere. Dette betydde at gruppen jobbet i relativ isolasjon, uten at gruppens handlinger kunne påvirke Gitek's reelle systemer eller daværende utviklingsprosesser. For å jobbe med serveren benyttet vi Windows sitt "Remote Desktop Connection", og jobbet på eksternt skrivebord. Her var det en uungåelig begrensning som ikke kunne håndteres. Grunnet begrensninger på serveren, var det kun mulig å ha to tilkoblede brukere samtidig. Dette var en stor flaskehals for gruppens del, med en fire medlemmer. For videre arbeid betydde dette at kun to av medlemmene hadde mulighet til å jobbe med nødvendige biblioteker av gangen. I kontrast med den opprinnelige hensikten, ble det behov for å bruke produksjonsserveren som testmiljø da dette var det eneste stedet koden vår kunne kjøre. Dette vurderer gruppen naturligvis ikke som best-practise, men ble nødvendig for å ikke halte fremgang i prosjektet. I dette tilfellet skulle ikke løsningen være distribuert til en brukerbasis under utvikling, så dette virket som det beste alternativet.

Arbeidet i denne perioden skjedde gjennom at to gruppemedlemmer koblet til, og jobbet skrev kode i tilsvarende IDE på serveren, med egnede biblioteker og databasetilgang. De to resterende gruppemedlemmene jobbet i mellomtiden med utfordringer som kunne utføres i isolert fra serveren. Det ble en god del venting på tur, og rotering av hvem som hadde tilgang til serveren. Denne perioden var preget av mye parprogrammering for å maksimalisere tilgangen gruppen hadde til serveren.

Etter samtaler med oppdragsgiver, ble gruppen tilbudt å få hjelp til å sette opp lokale kopier av servermiljøet. I uke 10 (5.- 11 Mars) var gruppen innom oppdragsgiver og fikk hjelp til å sette opp nødvendige programmer og verktøy for å kunne kjøre serveren på egne laptopper. Dette var en kompleks og tidkrevende prosess, men definitivt verdt det. Alle på gruppen kunne da skrive, teste og kjøre kode lokalt, før den ble pushet til repositoriet. Fra dette tidspunktet til slutten av prosjektet lot gruppen være å benytte oppdragsgivers server.

5.1.2 IDE

Når gruppen var i oppstartsperioden av prosjektet, ble det drøftet hvilke IDE valg som ville være best egnet for å løse oppgaven. Det ble enighet om en felles bestemmelse, da dette ville gjøre det lettere å hjelpe til med feilsøking relatert til utviklingsmiljø, hurtigtast-layout og IDE-spesifikke feilmeldinger.

Det sto mellom to valg: Visual Studio og PhpStorm. Visual Studio hadde majoriteten av gruppen allerede gjort seg kjent med. Et annet argument var at både Visual studio og deler av back-end løsningen er Microsoft produkter, og kunne potensielt skape rom for enklere kompatibilitet.

Likevel valgte gruppen å gå for PhpStorm. IDE-en er spesifikk for webutvikling, med gode hjelpeverktøy for HTML og JavaScript. PhpStorm er et JetBrains produkt, og har samme brukergrensesnitt som IDE-en gruppen benyttet i mobilutviklingsfaget parallelt med bacheloroppgaven. Som en bonus nevnte oppdragsgiver at han hadde erfaring med JetBrains produkter og kunne være til noe assistanse om det skulle trenge.

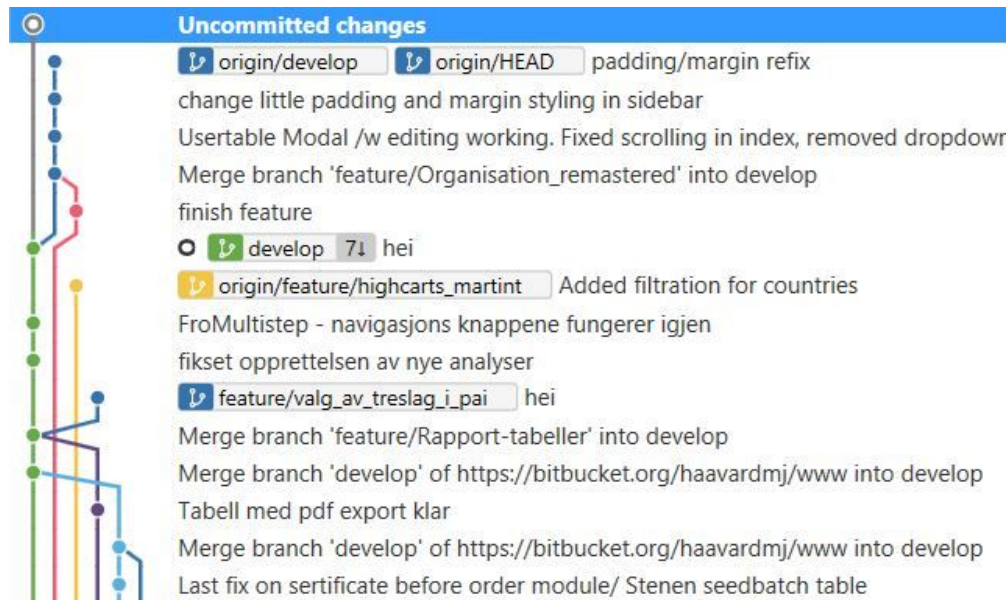
5.1.3 Repository

Gruppen valgte å bruke BitBucket som vårt repository da gruppen allerede hadde god erfaring med BitBucket i fra diverse prosjekter i løpet av studiet. Av BitBuckets funksjonaliteter var det hovedsaklig versjonskontrollen og de grunnleggende repository funksjonalitetene som ble brukt. Men BitBuckets funksjonaliteter som Wiki og Issue oversikt ble også vurdert. I forhold til den naturlige konkurrenten GitHub, tilbyr BitBucket også gratis konfigurasjon for private repositories.

5.1.3.1 SourceTree og GitFlow

Håndteringen av repositoriet lokalt på gruppemedlemmenes maskiner ble håndtert gjennom SourceTree. SourceTree gir et grafisk brukergrensesnitt til Git, noe som gjør Gits mer avanserte funksjonaliteter lettere å bruke.

Gjennom SourceTree sitt brukergrensesnitt kunne gruppen lett holde styr på commits, branches og den generelle fremgangen i prosjektet. Om gruppen for eksempel skulle ønske å gå tilbake til en eldre commit så kunne dette gjøres gjennom prosjektoversikten.



Figur 49: SourceTree branch tracker

SourceTree viser også statusen på de lokale filene sammenlignet med de samme filene som ligger i repositoret. På denne måten kan man enkelt se hvilke filer som har blitt endret og må pushes, eller hvilke filer som har merge konflikter i seg og må fikses.

SourceTree kommer også integrert med GitFlow. GitFlow er en modell for å håndtere og bruke forskjellige branches [6]. Valget ble gjort siden GitFlow gjør det enkelt å lage forskjellige midlertidige branches basert på den typen jobb du ønsker å gjøre. Hvis man for eksempel skal legge til en ny funksjonalitet så lager man en feature-branch og jobber i den til man er ferdig. Gjennom SourceTree tar man da så å avslutter feature-branchen og lar SourceTree ta seg av å merge feature-branchen inn i utvikler-branchen.

Siden mergingen av branches gjøres automatisk, samt at man kan lage nye branches når man trenger dem, så er GitFlow et utmerket verktøy i gruppe prosjekter da man kan unngå merge konflikter og splitte ut uferdig kode til egne branches som gruppe medlemmene kan jobbe på uten å måtte endre deres eksisterende kode.

5.2 Web-løsning

Siden oppgaven gikk ut på å lage en web applikasjon så brukte gruppen en rekke web relaterte teknologier. Bruken av disse teknologiene er diskuteres under.

5.2.1 HTML5

Valget å benytte HTML5 var et av de enklere. Det var vanskeligere å finne argumenter for å velge noe annet. Som nyeste versjon av Hypertext Markup Language, er det den definitive standarden for å beskrive innhold og utseende på nettsider. Til tross for at vi kun benytter en liten del av funksjonaliteten som skiller HTML5 fra sine eldre versjoner, var ryss-kompatibiliteten den tilbyr blant nettlesere et overveldende argument. Siden HTML5 har vært ute siden 2014, har det utvilsomt etablert seg som et essensielt språk for webutvikling.

5.2.2 Javascript og jQuery

Det kom naturlig for gruppen å bruke JavaScript for å lage funksjonalitet i web-løsningen da gruppen ikke kunne bruke PHP. JavaScript var allerede et web programmeringsspråk som gruppen hadde noe erfaring med å bruke. Siden det var behov for et programmeringsspråk i stand til å legge til ekstra interaktivitet på HTML sidene, var det vanskelig å finne et mer egnet språk enn JavaScript.

JavaScript i seg selv ble egentlig lite brukt av gruppen som et programmeringsspråk, men filtypen ble flittig brukt. Dette kom av at gruppen for det meste skrev JavaScript biblioteket jQuery. jQuery biblioteket tilbyr et API som gjør ting som dokument traversering, animasjoner, og eventer lettere å både bruke og å håndtere [7].

5.2.3 Tredjeparts-biblioteker

For mange av oppgavene gruppen skulle løse, holdt det ikke å benytte seg av oppdragsgivers biblioteker og standardfunksjonalitet fra anvendte programmeringsspråk. For disse utfordringene, hadde gruppen to valg: Enten egenutvikling med lage all koden selv, eller å finne biblioteker som allerede har en løsning på problemet. Gruppen prioriterte å bruke tredjeparts biblioteker, der det var mulig. Dette var gitt at de ikke hadde lisenser som hindret fri bruk. Valget av å bruke tredjeparts biblioteker sparte gruppen for en del tid og holdt gruppen unna anti-patternet "Gjenoppfinne hjulet" [4], da majoriteten av løsningen er bestående av velkjente komponenter.

5.2.3.1 Bootstrap

En av de største vurderingene gruppen måtte ta som omhandler Bootstrap var at alpha-versjon 4.0 ble lansert rett i forkant av oppstarten på prosjektet. Dette betydde at gruppen kunne velge å gå over til den nye versjonen i fra versjon 3. Men siden det var en helt ny versjon så hadde

både arbeidsgiver og gruppen noen bekymringer rundt potensielle bugs og enkelte funksjoner som ikke nødvendigvis ville funke i vår applikasjon.

Gruppen valgte likevel å oppgradere for å få erfaring med noen av det nye og attraktive egenskapene som versjonen tilbyr. Først og fremst tillot den nye versjonen oss å bruke størrelsen *xl*, for de fem forhåndsdefinerte media-query størrelsene beskrevet i design-kapittelet. En annen mindre, men spennende endring, er overgangen fra bruk av piksler til rem som primær CSS-enhet. I tillegg til dette var det en mulighet for arbeidsgiver å få testet hvordan Bootstrap 4.0 fungerte opp i mot deres systemer. Overgangen gikk for det meste feilfritt, med unntak av sertifikat-modulen som brukte biblioteker som ikke var oppdatert til å være kompatibel med Bootstrap 4.0. Gruppen måtte derfor degradere til versjon 3.3.7 for denne spesifikke modulen.

5.2.3.2 DataTables

DataTable ble brukt for å tilby bedre og mer funksjonalitet til tabellene som ble laget. Biblioteket inneholder alle ønskede funksjoner relatert til tabeller som ren bootstrap ikke tilbyr. Siden tabeller er en såpass sentral del av løsningen, var datafiltrering essensielt. DataTables inneholder en rekke verktøy som automatisk settes opp, eller kan legges til under initialisering av tabeller. Vi valgte å benytte oss av pagination for inndeling, umiddelbare søk på tvers av kolonner, og sortering på kolonnene.

DataTables tillater også å sette opp tabellene ut i fra et ajax objekt, noe som passet utmerket for vår løsning. Det gjorde jobben å fylle inn vilkårlig data mye enklere. Biblioteket tillater også eksportering av tabellinnholdet til flere formater, noe som ellers ville vært en omfattende jobb å implementere.

```

var myTable = table.dataTable({
  "data": data.value.records,
  autoWidth: false,
  select: true,
  responsive: true,
  "fixedHeader": {
    "header": true,
    "footer": true
  },
  "columns": [
    {"data": "nr"},
    {"data": "treslag"},
    {"data": "norsk_navn"},
  ],
});

window.onresize = function() {
  myTable.columns.adjust().responsive.recalc();
}

```

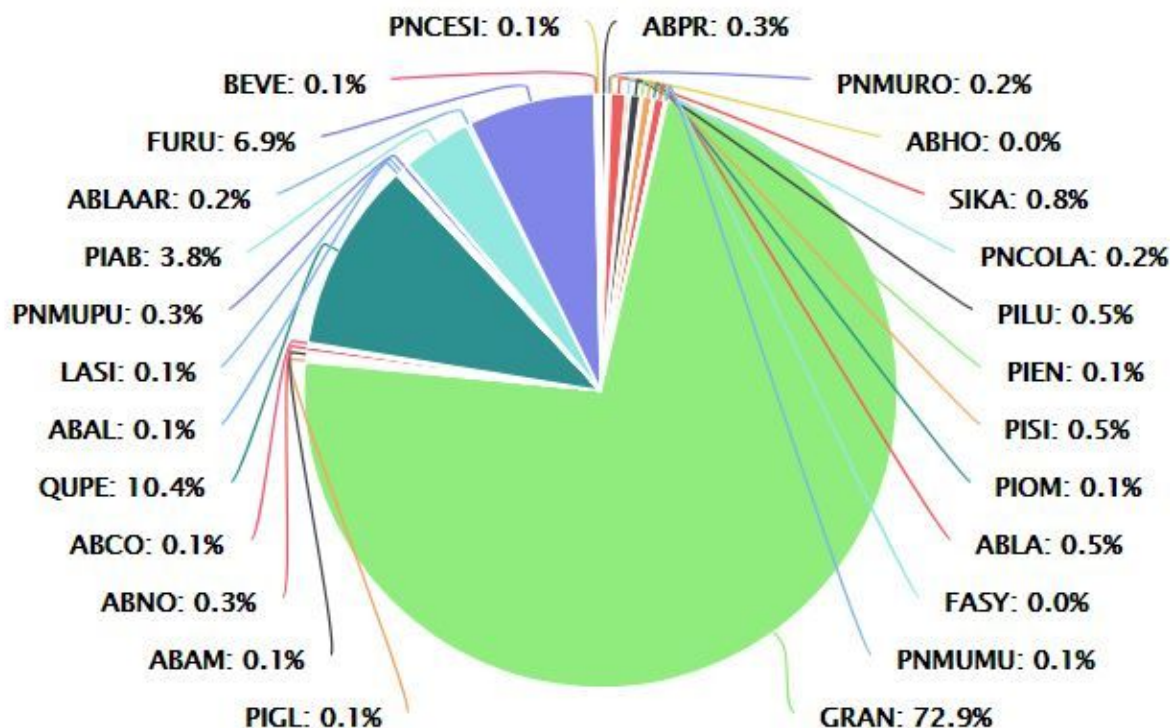
Figur 50: DataTable kode eksempel

Figur 49 viser hvordan et DataTable har blitt initialisert i løsningen (Noe kode er tatt bort for å skape en tydeligere presentasjon). Data fylles inn fra databaseresponsen `data.value.records`, og ønsket funksjonalitet settes til `true`. Kolonnene blir tildelt verdier ut i fra navnet på feltene i databaseresponsen. Til slutt settes tabellen til å endre størrelse sammen med vinduet.

5.2.3.3 HighCharts

I rapport-modulen var det nødvendig å vise frem representasjoner av data slik at sluttbrukeren kan lettere forstå dem. Oppdragsgiver foreslo derfor at gruppen kunne bruke HighCharts for å gjøre den jobben. HighCharts er et JavaScript bibliotek som lar brukeren lage interaktive grafer og diagrammer av forskjellige slag. Gruppen brukte for eksempel HighChart til å lage et sektordiagram som viser hvor mange kg frø totalt det er på lager av hver frøtype.

TOTAL MENGDE KG pr TRESLAG



Figur 51: Sektordiagram som viser prosentvis totalmengden av hvert treslag i kilogram.

I HTML-filen ble det laget en tom div med en unik id. Inne i tilhørende JavaScript-dokument brukte vi så jQuery til å hente den spesifikke div'en basert på id. Et nytt HighCharts diagram ble så laget basert på data i fra databasen og satt inn i div'en. HighCharts grafene har muligheten til å ta en rekke parametere som er med på å bestemme utseendet og funksjonaliteten til grafen. Eksempler på dette er fargene som brukes i diagrammet, typen diagram, og hvilke verdier som skal vises hvor. Kodeeksemplet under viser hvordan HighCharts lages og fylles med data i løsningen. Eksemplet er delt opp i to for å vise begge prosessene.


```

var circle = Highcharts.chart('kgLagerCircle', {
  chart: {
    plotBackgroundColor: null,
    plotBorderWidth: null,
    plotShadow: false,
    type: 'pie'
  },
  title: {
    text: 'TOTAL MENGDE KG pr TRESLAG'
  },
  tooltip: {
    pointFormat: '{series.name}: <b>{point.y:.0f}</b>'
  },
  plotOptions: {
    pie: {
      allowPointSelect: true,
      cursor: 'pointer',
      dataLabels: {
        enabled: true,
        format: '<b>{point.name}</b>: {point.percentage:.1f}%',
        style: {
          color: (Highcharts.theme &&
            Highcharts.theme.contrastTextColor) || 'black'
        }
      }
    }
  }
});

```

Figur 52: HighChart kode eksempel

```

//The array holds the data the pie-chart contains
var circleArray = [];

$.pmws.ajax("GET", "DB", "getrecords",
{
  tablename: "_sfv_froparti",
  fields: [
    "treslag, sum(kg_til_lager) kg_til_lager, count(*) antall"
  ],
  groups: "treslag"
}, function (data) {
  $.each(data.value.records, function (key, val) {
    var CircleObject = {name: val.treslag, y: val.kg_til_lager};
    circleArray.push(CircleObject);
  });
  circle.addSeries({
    name: "KG",
    colorByPoint: true,
    data: circleArray
  });
});
});

```

Figur 53: HighChart kode eksempel 2

5.2.3.4 Font Awesome

Gruppen trengte av og til ikoner som kunne brukes for å representere forskjellige handlinger og tjenester. Det var viktig at disse ikonene ikke lå under en lisens som hindret gruppen eller oppdragsgiver i å bruke dem. Font Awesome tilbyr over tusen forskjellige ikoner gratis, alle i vektorgrafikk som tillater vilkårlig skalering. Gruppen valgte derfor å inkludere biblioteket. Med en såpass stor samling ikoner, var det ikke nødvendig å lage egne. Ved bruk av biblioteket, kunne ikoner enkelt legges til i HTML ved å sette forhåndsdefinerte klassenavn i en i-tag.

Kode	Utseende
<code><i class="fas fa-user"></i></code>	

Figur 54: Bruk av Font Awesome i HTML

5.3 Oppdragsgivers Biblioteker

Siden vi valgte å integrere vår løsning med oppdragsgivers systemer kunne vi, og ble i flere tilfeller nødt til bruke noen av deres biblioteker eller moduler for å løse enkelte utfordringer. For å bruke modulene måtte vi kjøre løsningen vår på oppdragsgivers server, og bruke jQuery funksjoner for å kjøre oppdragsgivers egendefinerte AJAX kall opp imot backenden.

Vi måtte legge til skript fra oppdragsgivers portefølje vi ønsket å bruke før head-elementet i html filer. Skriptene har aliaser, og vi kunne også legge inn egne skript som vi fikk muligheten til å definere. Fordelen med dette var at vi enkelt kunne bruke aliasene, og slapp å bekymre oss for at stiene til filene var riktig formatert. I eksemplet under legges det til ønsket versjon av jQuery og jQueryUI.

```
<!DOCTYPE html>
<html lang="en">
<%
pmForm.AddJS('jQuery1112');
pmForm.AddJS('jQueryUi1114');
%>
<head>
  <%=pmForm.Head.text%>
</head>
```

Figur 55: Skript i fra oppdragsgivers portefølje legges til

En av de mest brukte modulene var DataForm. DataForm gjorde det mulig å automatisk sette data hentet i fra databasen inn i input-felter, og vice versa. På grunn av denne modulens potensiale ble den mye brukt i begynnelsen av prosjektet, da gruppen mente at vi modulen ville gjøre behandlingen av data lettere.

Men gruppen endte med å gå over til pmws ajax kall til databasen i stedet i noen tilfeller, da et DataForm objekt bare kunne brukes for dataen inne i en div. Det ble derfor nødvendig for gruppen å hente den ønskede dataen for å så behandle den selv i de tilfellene hvor data ble sendt/hentet i fra større HTML-dokumenter.

DataForm ble satt på som et argument i input-feltene som gruppen ønsket at DataForm skulle behandle. Om gruppen ønsket å bruke DataForm til å sette inn data i fra databasen ble denne dataen hentet i fra et JavaScript dokument med jQuery funksjonene for henting av data med DataForm. Om gruppen ønsket å bruke DataForm til å sette data inn i databasen i fra input feltene, så ble det satt på et DataForm save argument på en knapp i HTML dokumentet. Når denne knappen så trykktest på ble en jQuery DataForm funksjon kjørt som sendte dataen til databasen, gitt at input feltene og tabellen gruppen å sende data til stemte overens.

Biblioteket håndterer cookies og derunder sjekking av innlogget brukers rettigheter. Denne egenskapen ved biblioteket benyttes for login og styring av hvilken funksjonalitet og informasjon de forskjellige brukerne har tilgang på.

5.3.1 Database

Databasen vi brukte, Microsoft SQL Server Management Studio 2017, inneholdt oppdragsgivers modifiserte versjon av tabellene i fra kundens database. Selv med Gitek sin modifiserte database, endte vi opp med tabeller som inneholdt veldig mange felter, der en majoritet av dem sto tomme. Det ble derfor en oppgave å kartlegge hva som kunne klassifiseres som viktig og uviktig informasjon. Gruppen har ingen medlemmer med utpregede botaniske ferdigheter, og har følgelig lite erfaring med kundens fagfelt. Gruppen ble derfor avhengige av å følge oppdragsgivers rådgivning for felter spesifikt til frøpartier og lignende.

Data kunne hentes i fra databasen gjennom et ajax kall til Gitek's backend. Ajax kallet inneholdt parametere som spesifiserte hvilke felter i fra hvilke tabeller som skulle hentes ut. Eksempelet under hentes i fra Frøparti tabellen. Verdiene som skal hentes ut er verdien i "Land" feltet. Disse verdiene skal så sorteres etter alfabetisk rekkefølge, og hver verdi skal bare returneres en gang. Dataen som returneres av ajax kallet blir så håndtert i en anonym callback funksjon. I eksemplet blir dataen som returneres logget i konsollen slik at gruppen kan se verdiene i en nettleser.

```
$.pmws.ajax("GET", "DB", "getrecords",
  {
    "order": "Land",
    "fields": "Land",
    "tablename": "_SFV_FroParti",
    "distinct": true
  }, function (data) {
    console.log(data);
  });
```

Figur 56: Ajax kall for å hente data i fra databasen (tatt ut av sammenheng).

5.3.2 PDF

For sertifikat var det nødvendig å lage et egenutviklet alternativ for eksportering til PDF format. Originalt ønsket gruppen å bruke et tredjeparts bibliotek for generering av PDF-filer kalt jsPDF. Dette biblioteket gjorde det mulig for gruppen å lage pdf filer på klientsiden ved hjelp av jQuery [3]. Men jsPDF ligger under en MIT-lisens [3], noe oppdragsgiver frarådet gruppen å benytte under implementasjon i systemet. Oppdragsgiver foreslo heller at gruppen skulle bruke deres pdf-modul, noe gruppen sa seg enig til å gjøre.

PDF-modulen ble implementert i en egen funksjon. Funksjonen brukte, som de andre Gitek modulene, et AJAX kall til den rette modulen. I AJAX-koden ble det så spesifisert hvilke parametere pdf dokumentet skulle genereres etter. I kodesnutten under tar Funksjonskallet imot en selector på elementet og tilhørende barne-elementer som utgjør pdf'en som genereres.

```
$(document).ready(function () {  
    $('body').pmwsExport({type: 'pdf', useColorBox: false});  
});
```

Figur 57: Funksjon gjør HTML om til PDF

Ved første iterasjon av løsningen benyttet gruppen seg av bootstrap 4, dette førte til problemer med pdf-genereringen som forventet bootstrap 3 markup. Ved degradering til bootstrap 3 fløt elementene forbi A4 siden som ble generert. Ved testing og konsultering fant ikke gruppen ut av hva som forårsaket dette problemet og valgte å bruke skreddersydd css for å lage en korrekt generering av pdf'er. Dette endte opp med å bli en tidkrevende prosess. En del av feltene i sertifikatet var avhengig av data fra ordre-modulen, som ikke endte opp med å være tilgjengelige.

Som resultat ble den endelige implementasjonen for pdf'en generert med hardkodete verdier. Som et proof of concept har gruppen benyttet seg av id attributt på elementene som holder data i html-elementene som danner grunnlaget for pdf'en og endret disse ved hjelp av javascript. Dette er og hva gruppen ville implementert som løsningen etter ferdigstilling av ordre-modulen.

For rapport modulen la gruppen også til muligheten å eksportete HighChart grafer og spesifikke DataTables som pdf dokumenter. Dette lot seg gjøre gjennom de forskjellige bibliotekenes APIer, noe som sparte gruppen for en del implementeringstid.

5.3.3 Dynamisk lasting

Dynamisk lasting ble gjennomført med bruk av oppdragsgivers bibliotek for AJAX-kall etter single-page prinsipper. Følgende kodesnutt viser én av situasjonene hvor vi benytter dette. I dette eksemplet legges tabelloversikten for frøparti inn i template-siden for frøpartioversikt. (Ekstra kode som ikke demonstrerer eksempelet er fjernet for tydeliggjøre presentasjon)

```
$.pmws.ajax("get", "app", "ScriptFile", {
    "filename": "tableSeedBatchBody.html"
},
function (data) {
    $('#mainContent').append(data.value.html);
    $.pmws.addJS("prepareTable.js");
});
```

Figur 58: Dynamisk lasting av HTML side

Her hentes innholdet fra komponenten tableSeedBatchBody.html fra komponentens View. I callback-funksjonen benyttes jQuery-selector til å få tak i den dedikerte containeren, og innholdet fra komponenten fra parameteret tilføyes. Deretter hentes og kjøres skriptet fra tabellens tilhørende Controller. Innhold som alltid lastes dynamisk er den venstrestilte navbaren, header, og relevant innhold for siden.

6. Testing

For å sikre at løsningen oppfører seg som forventet har gruppen tatt i bruk forskjellige metoder for testing. Gruppen har som tidligere nevnt valgt å integrere løsning opp mot Giteks system og dermed løst all funksjonalitet som ligger i backend gjennom deres rammeverk. Dette har påvirket hvilke metoder gruppen har valgt å benytte seg av for testing av applikasjonen.

Siden gruppen har hentet mange artefakter fra eXtreme Programming og gjennom studiet benyttet seg av unit testing, et artefakt fra eXtreme Programming, ble denne metodikken vurdert tidlig i prosjektet. Da gruppen valgte å integrere prosjektet inn i Giteks rammeverk ble det klart at formatet på kall til rammeverket og data som blir returnert var noe gruppen trengte å undersøke nærmere. Tidligere i studiet hadde gruppen fått erfaring med å implementere unit testing i backend på data som mottas og transformeres, dette ville gruppen ikke ha mulighet til å gjennomføre ved integrasjon.

Det ble klart for gruppen at implementasjon av unit testing ville lagt mye ekstra arbeid på hver modul. Gruppen valgte derfor bort unit testing til fordel for andre metoder for testing av løsningen som var mindre arbeidsintensive. Videre var det også noe usikkerhet rundt hva som var fornuftig å teste, ettersom vi i lengre perioder ventet på deler av databasen, biblioteker og skiftet serverkonfigurasjon. Noe som burde hatt mer vekt i diskusjonen er muligheten for å implementere unit testing som en del av utviklingsprosessen og sett på implementering av tester i front-end.

Gruppen hadde ukentlige møter med Gitek for å vise frem progresjon i oppgaven samt diskutere med project owner (Gitek) hva som skulle implementeres i neste sprint. Siden gruppen manglet tilgang på ekspert-brukere var det høvelig å benytte seg av brukertesting i møtene med Gitek. Disse testene hadde og som formål å virke som user acceptance testing. På avlutningen av sprinter hadde gruppen en gjennomgang av ny funksjonalitet hvor Gitek kunne komme med innspill om hva som burde forandres og var uklart. Disse issueene ble satt til neste sprint og planleggingen av utviklingen av løsningen hadde en sprint etter leveranse for å rette opp i disse issueene.

Gjennom artefakter fra smidige utviklingsmetoder, derunder eXtreme Programming, hadde gruppen tilgang på personas, use cases og user stories. Dette kombinert med at vi kun hadde slutttester med Gitek, førte til at gruppen benyttet seg av for å kombinere kode inspeksjon og parprogrammering. Inspisering av koden var alltid gjort av en i gruppen som selv ikke hadde hjulpet til med utvikling av den kodeblokken og tasks i taskboardet ble flyttet til "to be verified" når koden var ferdig implementert. Tasken ble så flyttet til "done" hvis koden i tasken bestod kodeinspeksjonen, ellers ville den bli flyttet tilbake til open.

I tillegg til å inspisere koden gikk samme utvikler gjennom løsningen i rollen til personasen(e) som tasken tilhørte for å undersøke om brukeropplevelsen korresponderte med personas, use cases og user stories. To til tre av arbeidsdagene i uken satt prosjektgruppen fysisk sammen og generelt hadde gruppen som norm at man ikke skulle bruke mer enn 1 time på et problem før man fikk andre til å se på problemet. Ofte var det høvelig å sette mer enn et gruppemedlem til å jobbe på en task. I disse tilfellene hadde disse utviklerne hyppig kontakt med hverandre, dermed var det ofte at koden hadde blitt sett på av minst 3 gruppemedlemmer før tasken ble ferdigstilt.

Det ble og planlagt en større brukertest hvor gruppen ville forespurt å få testet løsningen på studenter på campus, lærere innen informatikk og ergonomi, og på andre ansatte hos Gitek. Dette var planlagt å ta sted ved fullføring av løsningen. Gruppen rakk ikke å implementere ordre-modulen og prioriterte koding og rapportskrivning foran en alpha-testing. Dette valget kom også som konsekvens av at ordremodulen ble forskjøvet og etterhvert kansellert. Den eksterne brukergruppen ble derfor irrelevant, og det var ikke aktuelt med testing for denne målgruppen. Gruppen hadde avklart med Gitek 04. april hva som måtte på plass for å gjennomføre en brukertest, men gjennomførte altså ikke dette.

7. Avslutning

I dette kapitlet vil vi diskutere hvordan prosjektet gikk sammenlignet med den opprinnelige planen. Kapitlet tar også for seg gruppens evaluering og kritikk av vårt eget arbeid og oppgaven, samt rapporten og prosjektets konklusjon.

7.1 Diskusjon og drøfting

Dette delkapitlet går inn på de sentrale valgene som ble gjort gjennom prosjektet, og hvilke betydninger dette fikk for oppgaven som helhet. Det drøftes gruppens opplevelse av dette valget, og hvilke betydninger det hadde for hvordan gruppen jobbet.

Gruppen hadde planlagt å utvikle modulene i kravspesifikasjonen sekvensielt. I tillegg til at gruppen hadde et for stort fokus på denne planen, fikk vi ikke implementert ordremodulen. Etter ferdigstilling av produktregister-modulen, som skulle etterfølges av ordremodulen fikk vi vite fra Gitek at datagrunnlaget ikke var på plass. Vi fikk tilbud om å jobbe med dataen direkte på produksjonsmiljøet, noe gruppen ikke følte seg komfortabel med da vi fortsatt var usikre på aspekter ved Giteks rammeverk.

Det var liten gevinst i å jobbe med dataen på Giteks produksjonsmiljø da gruppen ville gjort seg avhengig av et nytt miljø og ved utvikling ville funksjoner som å sette inn ordre eller sende mails skape reelle ordre og mails. Dermed ble ordre-modulen nedprioritert og analyse-modulen ble startet. Ordre-modulen ble videre forskjøvet inntil den 24. April hvor gruppen fikk tilgang på dataen i sitt lokale miljø. Gruppen tok beslutningen om å nedtrappe tid brukt på utvikling av løsningen, til fordel for rapportskrivning. Ordre-modulen ble påbegynt men er langt fra ferdigstilt.

På grunn av manglende kontakt med sluttbrukere av løsningen har gruppen, ved kartlegging og foredling av de funksjonelle kravene, valgt å benytte seg av modeller som skal holde brukeren i sentrum. Mer datasentrerte modeller som sekvensdiagram ble nedprioritert da gruppen valgte å integrere løsningen i Giteks system og dermed ikke utvikle back-end. Det å skaffe seg en oversikt over Giteks rammeverk har vist seg å være utfordrende og krevende. Gruppen ser i ettertid at å kartlegge Giteks systemer med UML-diagrammer ville både økt forståelsen ved utforming av modellene og hjulpet formidlingen av denne kunnskapen gjennom gruppen.

I planleggingsfasen ble det drøftet mellom gruppens medlemmer hvilken utviklingsmodell arbeidsformen til prosjektet skulle bygges opp rundt. Gruppen så verdien å kartlegge flere av de veletablerte modellene, dems rammer og premisser som styrer utviklingen ved valg av modellen. Verdien av å utføre denne analysen ved å se på de styrker og svakheter de forskjellige utviklingsalternativene inneholder gir prosjektstyringen mer klarhet i hvilken modell som passet utviklingsprosessen best. Denne utvelgingsprosessen ga medlemmene av gruppen gode muligheter å drøfte modellene seg i mellom, hvor hver av medlemmene fikk sjansen til å argumentere for det de mente var i fremkommende arbeid beste interesse.

Etter angitt ønske om integrasjon i deres systemer, ble valget rundt modell klargjort for gruppen. Hensikten var å kunne ved ukentlige oppdateringsmøter med oppdragsgiver presentere inkremitter som hadde blitt utformet i løpet av arbeidsuken.

Ved utvikling med smidig tilnærming, hvor valget landet på Scrum ble modellen prosjektarbeidet anvendte ikke noe rendyrket form av dette. Prosjektarbeidet bærte preg av elementer fra kanban, hvor arbeidsoversikten ble ført via et Kanban-board. Videre ble det brukt par-programmering gjennom modulene, user-stories, og delt-eierskap av utviklet kode som ble tatt fra XP sine rammer. Disse elementene er ved tilbakeblikk gjennom prosjektarbeidet sett på med gruppens blikk som et positivt tilskudd av elementer, som ga gruppen bedre samhandling.

Ved tilbakeblikk av gruppens evner til å følge rammene gruppen satt for seg selv og arbeidet, var gruppens manglende evne til å integrere den smidige tankegangen i implementasjonen av modulene. Modulene var som tidligere nevnt utviklet sekvensielt.

Det å benytte Gitflow endte opp med å bli dårlig oppfulgt i praksis. Den initiale fordelten av å ha nytt innhold isolert på egne branches, ble noe bortkastet. Dette var grunnet funksjonalitet som ble strukket over flere sprints, og uforutsette behov for refaktorering. Det virket derfor unaturlig å jevnlig dra komponenter ut i egne branches jevnlig. Det opplevdes heller som en brems på produktivitet, med unødvendige mengder overhead.

7.1.1 Integrasjon

Det aller første valget gruppen tok, var å integrere prosjektet i oppdragsgivers løsning. Denne avgjørelsen endret hele premisset for oppgaven, og var avgjørende for hvordan gruppen jobbet gjennom hele prosjektet. Vi har derfor valgt å tildele dette valget et eget punkt.

Bakgrunnen for dette valget var i utgangspunktet en bekymring om at oppgaven skulle bli noe mager for en gruppe på fire medlemmer. Samtidig virket dette alternativet som en unik mulighet til å tilegne seg kompetanse utenfor læreplanen. Å integrere kunne gi oppgaven en ekstra dimensjon. Det tillot gruppen å oppleve en mer realistisk arbeidssituasjon, hvor vi kunne jobbe direkte på en organisasjons systemer og med et reelt datagrunnlag.

Arbeidet som inngikk i å lage løsningen, ble forskjøvet til andre områder. Siden behovet om å utvikle egen backend forsvant fullstendig, ble gruppen heller nødt til å lære seg oppdragsgivers rammeverk. De benyttet helt fremmed software enn hva gruppen originalt vurderte. Det ble derfor heller et behov for å lære seg et annet interface og metode for å gjennomføre datautveksling. Dette var også tilfellet med de påkrevde systemene.

Fremfor å ta en avgjørelse på hvilke verktøy som var best egnet for oppgavene, ble vi låst til de kompatible alternativene. Dette betydde at arbeid som ellers ville gått til research, ble skjøvet over til å lære seg interne biblioteker. Et interessant og uvant aspekt for gruppen var at all

utvikling skjedde på klient-siden som følge av at vi ikke kunne forandre på backend. Dette har ført til at gruppen har implementert funksjonalitet på nye måter og blitt nødt å utforske andre metoder for strukturering av koden.

Gevinstene av å integrere var primært knyttet til selve prosessen. Her var det særlig erfaringer og kunnskap gruppen tok til seg som var det store utbyttet. Det å få være med på å ta kildekode fra repositoret til en faktisk produksjonsserver var veldig givende. Her ble gruppen gjennom veiledning fra oppdragsgiver introdusert for nødvendig konfigurasjon og oppsett. Dette belyste et steg i utviklingsprosessen vi har fått lite innsikt i gjennom studiene.

Oppsett av lokalt servermiljø med database viste seg å være en omfattende prosess. Her fikk gruppen opplæring i hvordan å konfigurere nødvendig software, slik at vi selv var i stand til å håndtere koden hele veien fra IDE til produksjon. Forbi dette var det morsomt å få benytte seg av kraftige biblioteker vi ellers ikke ville hatt tilgang til. Her utmerker HighCharts seg, som en ellers pay-to-use tjeneste.

I retrospekt er det lett å identifisere utfordringene gruppen har støtt på under prosjektarbeidet, samtidig som at gruppen erkjenner at det er valg og teknikker vi kunne tatt i bruk for å unngå noen av utfordringene vi støtte på. Hvilket utviklingsmiljø gruppen har hatt tilgang på har forandret seg gjennom prosjektet. Fra å jobbe uten kontakt med Giteks systemer, til ekstern server hvor vi endte med å få satt opp lokale systemer på egne maskiner. Allerede tidlig i utviklingsprosessen var vi for knyttet til vår opprinnelige plan og prioriterte å implementere funksjonalitet i CRM-modulen uten tilgang på hele utviklingsmiljøet. Hadde gruppen revurdert sine planer under denne perioden kunne vi konstruert layoutet til flere moduler og implementert funksjonaliteten da vi fikk tilgang til den eksterne serveren. Samtidig kunne gruppen implementert unit testing med forbehold om endringer i datagrunnlaget som testes, for å jobbe seg gjennom funksjonaliteten til modulene.

Som resultat av flere overganger, ble vi nødt til å refaktorere løsningen for å garantere kompatibilitet underveis. Dette er også som en følge av at vi ønsket å ferdigstille hver modul for seg selv. Gruppen endte med å implementere funksjonalitet på bakgrunn av antagelser og andrehånds kunnskap om utviklingsmiljøet. Da vi kjørte koden på den eksterne serveren var det enkelte funksjoner som ikke fungerte og gruppen mistet tid på bugfixing. Da vi først fikk tilgang på den eksterne serveren burde gruppen gjort en større jobb å kartlegge de forskjellige kallene vi trengte mot Giteks API. Her begynte gruppen å kjenne presset på å ligge bak skjemaet og jobbet heller på for å lære kallene samtidig som ny funksjonalitet ble implementert.

En mer konkret utfordring i forbindelse med utforskningen og læring av oppdragsgivers biblioteker var den begrensede dokumentasjonen. Gruppen var vant med mer etablerte rammeverk, som som regel har en veldig rik og velbeskrevet API. Dette kom noe uventet, og vi brukte lengre tid på å lære anvendelse enn forventet. En annen konsekvens av å jobbe med egenutviklede biblioteker, ble at gruppen ikke fikk muligheten til å finne andre kilder, eksempler

eller forum på nett for å søke hjelp. Som følge av dette ble vi nødt til å henvende oss til Gitek i tilfeller hvor vi hadde store vansker med å løse utfordringer.

Hadde gruppen vært mer reaktiv og kjent til omstendighetene som skapte utfordringer, kunne vi stilt mer forberedt til milepælene hvor vi fikk tilgang til essensielle ressurser og data. Vi holdt fast ved den originale planen om å fullføre modulene sekvensielt, noe som førte til at tidligere arbeid måtte refaktoreres for å kunne brukes.

For løsningen og resultatenes del hadde det vært fordelaktig om gruppen valgte en egenutviklet løsning, uten integrasjon. Vi kunne tatt flere empiriske valg og støttet oss mer på tidligere erfaringer. Gruppen tror det ville økt oppnåelsesgrad i forbindelse med resultatmålene. Likevel virker muligheten til å være med på hele integrasjonsprosessen som uvurderlig. Det er vanskelig å dokumentere og kvantifisere verdien av opplevelsen. Gruppen mener utvilsomt integrasjonen var verdt det for personlig vekst som programvareutvikler, i lag med de nye erfaringene tilegnet gjennom samarbeid med en utviklerbedrift.

7.2 Resultater

Som forklart i kapittel 1.6 så hadde gruppen satt seg en rekke mål for prosjektet. De overordnede målene var at gruppen skulle lage en applikasjon som oppdragsgiver kunne bruke til inspirasjon til deres eget arbeid.

7.2.1.1 Effektmål

Et av hovedpunktene som gruppen ønsket å oppnå under dette punktet var å forbedre brukergrensesnittet og brukeropplevelsen til det originale prosjektet, ved å bruke mer moderne utviklingsmetoder og standarder. Dette mener gruppen at vi klarte å oppnå. Løsningen er etter gruppens mening mer oversiktlig og enklere å bruke.

Gruppen mener også at vi har laget en løsning som kan lett videreutvikles. Dette kommer naturlig grunnet programmeringsspråkene og plattformen gruppen utviklet for. Men gruppen har også organiserte prosjektet for å tillate videreutvikling av løsningen, uten en bratt læringskurve for å forstå prosjektstrukturen.

7.2.1.2 Resultatmål

Gruppen oppnådde de aller fleste av resultatmålene som vi hadde satt oss ved starten av prosjektet. Løsningen gruppen lagde er fullt integrert inn i oppdragsgivers systemer, og kjører nå på en av deres servere og bruker ekte data i fra deres database.

Gruppen hadde også satt seg et mål hvor den nye løsningen skulle være brukervennlig og intuitiv. Her satte gruppen inn en del ressurser for å nå målet, og resultatet er gruppen ganske fornøyd med. Etter gruppens mening er den nye løsningen mer brukervennlig og lettere å navigere enn den gamle Paradox løsningen.

Overføre eksisterende funksjonalitet i fra Paradox versjonen var også et mål som gruppen mente var realistisk å oppnå. Ved unntak av ordre-modulen, så mener gruppen at det aller meste av eksisterende funksjonalitet har blitt implementert inn i gruppens løsning. Det var også viktig for både gruppen og oppdragsgiver at løsningen skalerte og var brukbar på både datamaskiner og mobile enheter. Dette målet jobbet gruppen mye med, og som et resultat så skalerer løsningen ganske bra helt ned til de aller fleste mobiltelefoner.

Målet å lage en løsning med forskjellige rettigheter for brukerne basert på grupperbruker ble dessverre ikke oppnådd, da gruppen valgte å fokusere tiden og arbeidet vårt på andre oppgaver og moduler i stedet. Som tidligere nevnt, ble dette valget tatt i forbindelse med den frafalne ordremodulen.

7.2.1.3 Læringsmål

Når det kom til læringsmålene så er det her gruppen fikk mest personlig utbytte da alle gruppemedlemmene føler at vi har lært en god del om programvareutvikling av å jobbe på prosjektet.

Gruppen tilegnet seg nye kunnskaper gjennom bruken av JavaScript i integreringen av Gitek systemet og oppsett av prinsipper for single-page applikasjon. Gruppen fikk også muligheten til å tilegne seg erfaring med teknologier som HighCharts og DataTables, som viste seg å bli viktige verktøy i prosjektet. Gruppen kommer derfor til å bringe med seg kunnskapene vi tilegnet oss om disse teknologiene videre til fremtidige prosjekter. Vi lærte også mye om Bootstrap og CSS, og deres verktøy for å lage responsive løsninger.

Men det gruppen mener vi lærte aller mest av var å jobbe med et prosjekt i en reell jobbsituasjon. Hele prosessen i fra planleggingsfasen frem til ferdigstilling av rapporten har vært en veldig lærerik prosess hvor gruppen har tilegnet seg erfaringer som vi ikke har fått av å jobbe på skoleprosjekter i løpet av våres studie. Ting som tidsestimering for større prosjekter, det å komme inn i en bedrift for å jobbe inn under deres systemer, og det å ha en ekte oppdragsgiver som forventer ukentlig fremdrifts gjennomføring har alle vært gode erfaringer som gruppen kan se tilbake på.

Totalt sett hadde oppgaven en bratt læringskurve. På grunn av dette tok det en del tid før gruppen fikk startet å jobbe ordentlig effektivt og en del tid ble brukt på å forstå de forskjellige systemene. Men etter at gruppen hadde brukt en del tid med systemet utviklet vi en god forståelse. Vi fikk oppleve kostnaden av opplæring i interne systemer. Om gruppen skulle gjennomført en ny oppgave for Gitek, ville hele prosessen gått mye raskere grunnet den kunnskapen vi har opparbeidet.

7.3 Videre utvikling og arbeid

På dette tidspunktet er det ikke avtalt noe videreutvikling av prosjektet med oppdragsgiver, men om gruppen skulle fortsette å jobbe på prosjektet så ville den første prioriteten vår være å få alle modulene helt på plass. Det vil innebære å implementere ordremodulen, samt all funksjonalitet i fra de andre modulene som er basert rundt ordremodulen.

I tillegg til dette ville gruppen testet og ferdigstilt all funksjonalitet vi ikke har fått tid til å finpusse. Et eksempel på dette er analysemodulen hvor brukergrensesnittet og mesteparten av koden er på plass. Men logikken som sender data hentet i fra mange forskjellige div'er er ikke på plass enda.

En av planene gruppen originalt hadde for å videreutvikle løsningen om vi skulle få tid til overs var å lage en app variant av løsningen. Dette ville ha vært til nytte for sluttbrukerne av applikasjonen da mange av dem bruker mobile enheter, som for eksempel Ipader, for å utføre deler av arbeidsoppgavene deres. Gruppen ville først ha laget en versjon for Android da det er plattformen gruppen har erfaring med å lage applikasjoner for, men ville også senere laget en IOS versjon for å få dekket majoriteten av mobilbrukere. App versjonene av løsningen ville ha fungert på stort sett samme måte som web løsningen, men den største forskjellen er at gruppen måtte ha laget backenden til appene selv, da appene ikke kan kobles opp i mot oppdragsgivers web systemer.

7.4 Evaluering av gruppens arbeid

I delkapitlene under kommer en diskusjon av gruppens arbeid som en helhet.

7.4.1 Innledning

Gruppen startet arbeidet på oppgaven 14.01.2018 og jobbet helt frem til innleveringsfristen 16.05.2018. I løpet av den tiden mener gruppen at vi har hatt et godt samarbeid og ingen større problemer innad i gruppen. Mindre unntak var noen små argumenter her og der angående implementasjonsbeslutninger, men det er noe som hører med i en hver utviklingsprosess og brakte ikke gruppen ned.

7.4.2 Organisering

Gruppen har drøftet de opprinnelige avgjørelsene rundt temaet organisering i kapittel 3, men vi velger å utdype mer om hvordan dette har fungert her.

7.4.2.1 Kommunikasjonskanaler

Gruppens interne kommunikasjonskanal for all utvikling av prosjektet har vært over den digitale chat applikasjonen *Discord*. Gjennom Discord har gruppen kunne opprette chatterom som har

latt oss separerer diskusjoner og samtaletemaer. Discord tillot også gruppen å ha samtaler med bruk av mikrofon. Dette ble blant annet brukt til å ha forskjellige typer møter når gruppen ikke var på universitetet.

Et konsept som ble tydeliggjort gjennom bruk av Discord som kommunikasjonskanal, var effekten av *push over pull*. Vi merket at det ofte ikke holdt å legge inn nye dokumenter i skyen hvor vi oppbevarte dokumenter. Dette ble for passivt, og oversett. Discord har applikasjoner for både desktop og mobil. Det var derfor mye mer effektivt å sende ut meldinger med notifikasjoner for å forsikre seg om at informasjon når ut til alle medlemmene.

Kommunikasjon med oppdragsgiver skjedde gjennom *Slack*, en applikasjon som var tilsvarende Discord. Gjennom Slack kommuniserte gruppen og oppdragsgiver mindre formelt, og brukte et aktivt chatroom. Dette fungerte mye bedre enn e-post, som skapte en tregere flyt i samtalene. Gruppen og arbeidsgiver hadde aktiv kommunikasjon, og det ble utvekslet meldinger de fleste arbeidsdager. Dette var den veldig positiv opplevelse, og kjempefint i situasjoner hvor vi møtte på utfordringer med deres systemer.

7.4.2.2 Møter

Gjennom hele prosjektet har gruppen jevnlig holdt møter innad i gruppen, med veileder og oppdragsgiver.

Møtene med Oppdragsgiver har skjedd hver fredag gjennom hele prosjektperioden. Disse møtene har blitt brukt til å få deres tilbakemeldinger på den ukens pågående arbeid og backloggen for den oppkommende sprinten. På denne måten viste oppdragsgiver hele tiden hva vi holdt på med, samtidig som at de kunne komme med innspill om ting som de mente ville forbedre oppgaven. Møtene ble også brukt av oppdragsgiver til å vise gruppen hvordan deres interne systemer fungerer, slik at gruppen kunne bruke den oppkommende sprinten til å implementere funksjonalitet opp mot oppdragsgivers systemer. Oppdragsgiver ønsket også at gruppen skulle sende dem agendaer minst 24 timer før møtet slik at de kunne forberede svar på de større temaene som gruppen ønsket å snakke om.

Gruppen hadde fastsatte møter med veileder hver torsdag gjennom prosjektperioden, med unntak av de ukene da gruppen mente det var unødvendig. I møtene med veileder fikk gruppen tilbakemeldinger om hva som burde gjøres fremover, samt at veileder ble oppdatert på statusen til prosjektet som en helhet. Dette gav oss en indikator på hvordan vi lå an underveis. Gruppen kunne også ta med utkast av rapporter og andre dokumenter og få tilbakemeldinger på disse. Det ble også laget agendaer for veileder i de tilfellene hvor gruppen hadde større spørsmål som krevde forberedning.

Etter de mer formelle møtene skrev gruppen møtereferater som omhandlet sakene som ble tatt opp. Disse referatene viste seg å bli veldig nyttige når gruppen trengte å se tilbake på hvilke

beslutninger som ble tatt i løpet av møtene. Noe som kom godt med mens gruppen skrev rapporten.

7.4.3 Fordeling av arbeid

Fordelingen av arbeidet ble hovedsakelig gjort i løpet av møtene gruppen hadde i starten av hver sprint/uke. I løpet av disse møtene kunne gruppemedlemmene foreslå selv hvilke arbeidsoppgaver de ville jobbe på i løpet av sprinten/uken eller bli tilegnet en oppgave i fra resten av gruppen. Etter at de forskjellige gruppemedlemmene hadde tilegnet seg erfaring innen forskjellige områder ble forskjellige arbeidsoppgaver også naturlig passende til spesifikke personer. For eksempel så var det Håvard som først satte seg inn i hvordan gruppen kunne sette opp designet til løsningen satt opp i mot oppdragsgivers systemer. Det ble derfor naturlig at det var Håvard som hadde hovedansvaret for designet i de senere sprintene også.

Som nevnt i kapittel 3.3 så brukte gruppen også YouTrack for å holde styr på hvilke arbeidsoppgaver gruppen hadde for de forskjellige sprintene. Gjennom YouTrack kunne gruppemedlemmene da se hvem som jobbet på hva, og statusen på de forskjellige oppgavene.

I gruppereglene, som ble definert i prosjektplanen (se siste vedlegg), ble gruppen også enige om at det var forventet at alle medlemmene skulle jobbe 30 timer i uken. Dette ble til tider vanskelig for gruppen da vi hadde prosjekter som måtte ferdigstilles i andre fag samtidig som gruppen jobbet på bacheloren. Dette reflekteres i timeføringen mot slutten av prosjektet. Det har også vært tilfeller hvor gruppemedlemmer har vært syke, og tider hvor vi ble nødt til å vente på å få tilgang til data fra arbeidsgiver. I slike situasjoner har vi avviket fra den opprinnelige forventningen.

Hvor og når gruppemedlemmene jobbet var opp til dem selv, med unntak av avtalte møter. Dette førte til at gruppemedlemmene jobbet en del individuelt. Når gruppen møttes på universitetet ble det ofte gjort parprogrammering. Dette hjalp gruppen med å løse problemer som enkeltindivider slet med og utnyttet den varierte kompetansen om forskjellige biblioteker og kodesnutter i mellom gruppens medlemmer.

7.4.4 Prosjekt som arbeidsform

Prosjekt som arbeidsform har bydd på fordeler og utfordringer.

Fordelen med å ha prosjekter som arbeidsform er at man kan i en satt periode jobbe med et bestemt tema, for å så avslutte prosjektet ved prosjektslutten for å starte på et nytt ett. På denne måten kan man rask tilegne seg erfaring i fra forskjellige utviklings- metoder og verktøy. Man blir også ikke like fort lei arbeidet sitt når man går i fra prosjekt til prosjekt, noe som åpner for muligheten for noe nytt hver gang.

Nedsiden med denne arbeidsformen er at det er mulig at man er ansatt i en bedrift på prosjekts basis. Det vil si at hvis du ikke får være med på det neste prosjektet til bedriften så har du ingen jobb. En annen nedside er at er at du aldri får satt det ordentlig inn i temaet du jobber med da du mest sannsynligvis bare har tid til å bruke de delene du trenger av alle de nye redskapene.

Gruppens erfaringer med prosjekter som arbeidsform i fra dette prosjektet har vært en god blanding av gode og dårlige opplevelser. Arbeidsformen passet med det at lengden på prosjektet passet bra til lengden på bachelor oppgaven som en helhet, men med det sagt så hadde ikke gruppen nokk tid til å implementere all funksjonaliteten på den gitte tiden, noe oppdragsgiver også var klar over.

7.5 Konklusjon

Gruppen har tilegnet seg mange erfaringer gjennom utførelse av denne bacheloroppgaven. Prosessen og læringsutbytte har ikke vært som forventet. Vi gikk inn i det siste semesteret og trodde vi skulle tilegne oss spisskompetanse innenfor enkelte fagfelt og at utviklingen av bachelorprosjektet ville samsvare med erfaringene vi har tilegnet oss gjennom studiet.

Den kunnskapen gruppen sitter igjen med som vi setter mest pris på, er erfaringen å jobbe tett på en oppdragsgiver og et stort system, hvor informasjon ikke kan hentes fra nettet eller gjennom dokumentasjon. Selv om vi har dratt god nytte av det vi har lært gjennom studiet, har oppgaven fått oss til å benytte denne kunnskapen på uvante måter. Vi har vært heldige og forholdt oss til en product owner som har vært investert i gruppen og prosjektet.

Kilder

[1] Gitek AS (18.12.2017). *Produktspesifikasjon Skogfrøverket*

[2] Sommerville I. (red). (2016) *Software Engineering (global edition)*. Tiende utgave. England: Pearson.

[3] Hall, J. (2010-2017). jsPDF [internett]. 2010[07.03.2018, 18.02.2018]. Tilgjengelig via: <https://github.com/MrRio/jsPDF>

[4] Sourcemaking.com. Reinvent The Wheel [internett]. [siteret 20.03.2018]. Tilgjengelig via <https://sourcemaking.com/antipatterns/reinvent-the-wheel>

[5] Microsoft. IIS [internet]. 2018 [siteret 08.05.2018]. Tilgjengelig via <https://www.iis.net/>

[6] DataSift. Introducing GitFlow [internet]. [siteret 08.05.2018]. Tilgjengelig via <https://datasift.github.io/gitflow/IntroducingGitFlow.html>

[7] The jQuery Foundation. jQuery [internet]. 2018 [siteret 10.05.2018]. Tilgjengelig via <https://jquery.com/>

[8] Jarrett C. (2008) *Forms that Work: Designing Web Forms. Designing Web Forms for Usability (1st Edition)*

[9] Norman D. A. (2013) *The Design of Everyday Things*. USA: Basic Books.

Figurer

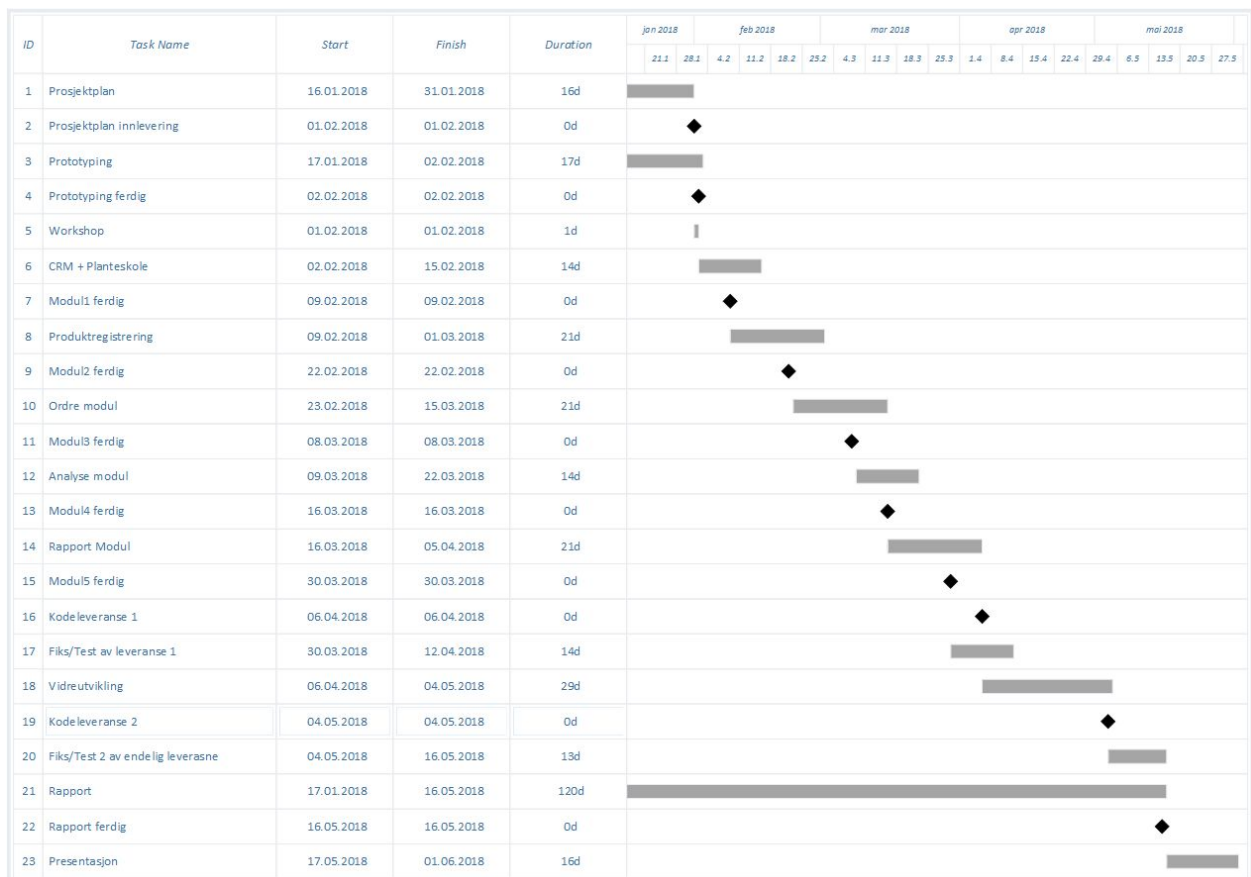
Alle bilder brukt til personas er hentet fra <https://www.pexels.com/> og er lisensiert under en CC0-lisens.

Vedlegg

Her ligger alle vedleggene gruppen mente kunne være relevante for rapporten.

Vedlegg A - Planer

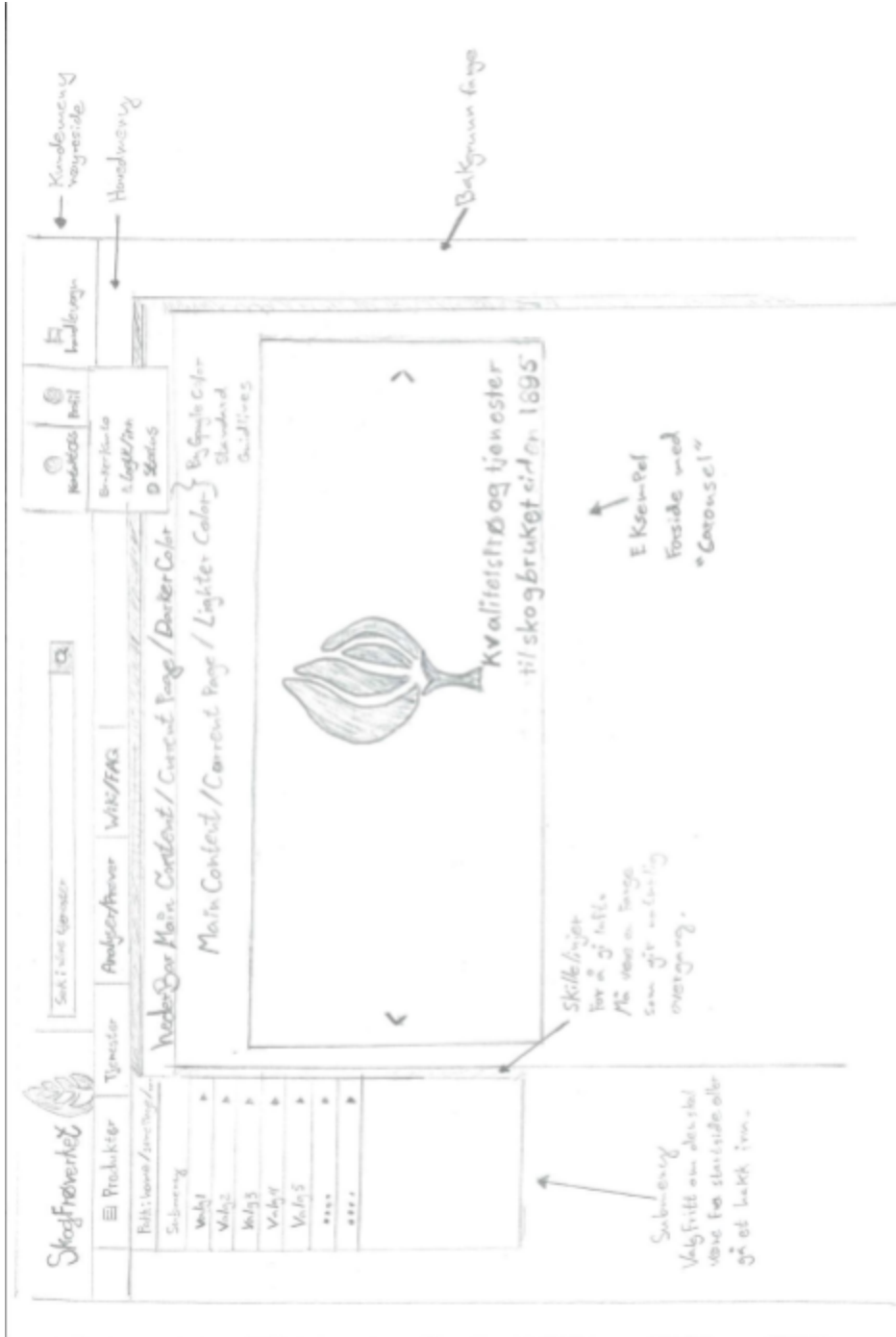
A.1 - Gantt diagram i fra prosjektplanen



A.2 - Gantt diagram i fra slutten av prosjektet

ID	Task Name	Start	Finish	Duration	jan 2018			feb 2018				mar 2018				apr 2018				mai 2018						
					14.1	21.1	28.1	4.2	11.2	18.2	25.2	4.3	11.3	18.3	25.3	1.4	8.4	15.4	22.4	29.4	6.5					
1	Prosjektplan	16.01.2018	27.01.2018	12d	█																					
2	Prosjektplan levert	01.02.2018	01.02.2018	0d				◆																		
3	Research	24.01.2018	09.02.2018	17d	█																					
4	Prototyping	01.02.2018	07.02.2018	7d				█																		
5	Temp/Demo HTML filer	05.02.2018	12.02.2018	8d				█																		
6	Fikk tilgang til Gitek Server	14.02.2018	14.02.2018	0d				◆																		
7	Modul 1 - CRM	16.02.2018	04.04.2018	48d				█																		
8	Testing og kompetanse heving av Giteks rammeverk	19.02.2018	26.02.2018	8d				█																		
9	Modul 2 - Produktregistrering	21.02.2018	23.03.2018	31d				█																		
10	Tilgang til lokalsystemer	07.03.2018	07.03.2018	0d								◆														
11	Modul 3 -Ordre (ikke gjennomført)	15.01.2018	15.01.2018	0d	◆																					
12	Modul 5 - Rapport	09.03.2018	18.03.2018	10d								█														
13	Rapport modulen stanses midlertidig	18.03.2018	18.03.2018	0d								◆														
14	Modul 4 - Analyse	18.03.2018	19.04.2018	33d																█						
15	Dynamisk lastning	21.03.2018	07.04.2018	18d																█						
16	Mappestruktur omstrukturering	21.03.2018	21.03.2018	0d								◆														
17	Påske ferie	26.03.2018	01.04.2018	7d								█														
18	Modul 5 gjennoptas	08.04.2018	17.04.2018	10d																█						
19	Fokus på programmering avsluttes	24.04.2018	24.04.2018	0d																◆						
20	Prosjekt-Rapport skrivning	26.02.2018	16.05.2018	80d				█																		
21	Ren pussing av kode	30.04.2018	16.05.2018	17d																█						
22	Prosjektet avsluttes og leveres	16.05.2018	16.05.2018	0d																					◆	

Vedlegg B - Prototyper



SKOGFRØVERKET

Log in Sign up

 Tjenester	 Produkter	 Konglefrøer
 Veiledning fra Plunder	 BlowstringRapport	 FAQ Wiki/FAQ
		 Kundesupport



Header

Content

Sidebar

PROTOTYPE: TABELLFREMVISNING

Header

Søkefelt

Filterering

TabellHeader	TabellHeader	TabellHeader	TabellHeader	TabellHeader	TabellHeader
Rad	Rad	Rad	Rad	Radiger	Slett
Rad	Rad	Rad	Rad	Radiger	Slett
Rad	Rad	Rad	Rad	Radiger	Slett
Rad	Rad	Rad	Rad	Radiger	Slett
Rad	Rad	Rad	Rad	Radiger	Slett
Rad	Rad	Rad	Rad	Radiger	Slett
Rad	Rad	Rad	Rad	Radiger	Slett
Rad	Rad	Rad	Rad	Radiger	Slett
Rad	Rad	Rad	Rad	Radiger	Slett
Rad	Rad	Rad	Rad	Radiger	Slett
Rad	Rad	Rad	Rad	Radiger	Slett
Rad	Rad	Rad	Rad	Radiger	Slett

Sidebar

Forrige 1 2 ... 5 6 Neste

Header

Form-title(Pop-up modal)

Input-title

Input-field

Input-title

Input-field

Input-title

Input-field

Input-title

Input-field

Input-title

Input-field

Input-title

Input-field


Submit

Cancel

Sidebar

Header

Profile	Edit	Orders
----------------	-------------	---------------



Picture

Userinfo

Userinfo

Userinfo

Userinfo

Organisation

Tableheaders

memberinfo

memberinfo

memberinfo

memberinfo

memberinfo

memberinfo

memberinfo

memberinfo

memberinfo

memberinfo

memberinfo

Sidebar

114

Header

Form-title

Input-title

Input-field

Input-title

Input-field

Input-title

Input-field

Input-title

Input-field

Input-title

Input-field

Input-title

Input-field

Input-title

Input-field

Previous



Next

Sidebar

Header

Form-title				
Search & Filters	Product-card	Product-card	Product-card	Product-card
	Product-card	Product-card	Product-card	Product-card
	Product-card	Product-card	Product-card	Product-card
	Product-card	Product-card	Product-card	Product-card

Sidebar

Vedlegg C - Kontrakter

Prosjektavtale

mellom NTNU Fakultet for informasjonsteknologi og elektroteknikk (IE) på Gjøvik (utdanningsinstitusjon), og

Gitek AS

(oppdragsgiver), og

Martin Trehjørningen Harestad, Vetle Nikolai Løland

Lundsvoll, Håvard Myrvold Johannessen, Martin

Stenen

(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 01.01.18 til 01.06.18.

Studentene skal i denne perioden følge en oppsatt fremdriftsplan der NTNU IE på Gjøvik 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 NTNU 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 NTNU på Gjøvik. Studentene dekker utgifter for ferdigstillelse av prosjektmateriell.
 - 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. NTNU IE på Gjøvik 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 intern og ekstern sensor. Likevel er det en forpliktelse for utøverne av prosjektet å fullføre dette til avtalte spesifikasjoner, funksjonsnivå og tider.

4. Alle bacheloroppgaver som ikke er klausulert og hvor forfatteren(e) har gitt sitt samtykke til publisering, kan gjøres tilgjengelig via NTNUs institusjonelle arkiv hvis de har skriftlig karakter A, B eller C.

Tilgjengeliggjøring i det åpne arkivet forutsetter avtale om delvis overdragelse av opphavsrett, se «avtale om publisering» (jfr Lov om opphavsrett). Oppdragsgiver og veileder godtar slik offentliggjøring når de signerer denne prosjektavtalen, og må evt. gi skriftlig melding til studenter og instituttleder/fagenhetsleder om de i løpet av prosjektet endrer syn på slik offentliggjøring.

Den totale besvarelsen med tegninger, modeller og apparatur så vel som programlisting, kildekode mv. som inngår som del av eller vedlegg til besvarelsen, kan vederlagsfritt benyttes til undervisnings- og forskningsformål. Besvarelsen, eller vedlegg til den, må ikke nyttes av NTNU 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 NTNU og/eller studenter har interesser.

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 NTNUs elektroniske eksamenssystem. I tillegg leveres ett eksemplar til oppdragsgiver.
8. Denne avtalen utferdiges med ett eksemplar til hver av partene. På vegne av NTNU, IE er det instituttleder/faggruppeleder som godkjenner avtalen.
9. I det enkelte tilfelle kan det inngås egen avtale mellom oppdragsgiver, student(er) og NTNU som regulerer nærmere 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 med oppdragsgiver, skjer dette uten NTNU som partner.
10. Når NTNU også opptrer som oppdragsgiver, trer NTNU inn i kontrakten både som utdanningsinstitusjon og som oppdragsgiver.
11. Eventuell uenighet vedrørende forståelse av denne avtale løses ved forhandlinger avtalepartene imellom. 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:

NTNUs veileder (navn): Frode Haug

Oppdragsgivers kontaktperson (navn): Khai Van Ngo

Student(er) (signatur): Martin T. Hovestad dato 15.01.18

Volle, M.C. Lundsvoll dato 15.01.18

Håvard Myrvald Johnsen dato 15.01.18

Martin Stenen dato 15.1.18

Oppdragsgiver (signatur): Khai Van Ngo dato 15.07.18

Signert avtale leveres digitalt i Blackboard, rom for bacheloroppgaven.

Godkjennes digitalt av instituttleder/faggruppeleder.

Om papirversjon med signatur er ønskelig, må papirversjon leveres til instituttet i tillegg.

Plass for evt sign:

Instituttleder/faggruppeleder (signatur): _____ dato _____

Vedlegg D - Rapporter og referater

D.1 - Møterreferater

Møtereferatene var mest for vår egen del, og er ikke nødvendigvis spesielt formelle eller tydelige. De ble primært laget for at vi skal få med oss nøkkelpunkter så vi ikke glemmer viktig informasjon.

D.1.1 Møtereferater i fra møter med Gitek

Møtereferat møte med Gitek - 15.01.2018

Tittel	Første møte med Khai fra Gitek på nyåret
Tidspunkt	14:00 til 15:30 - 15.01.2018
Oppmøte	Hele gruppen, Khai
Innhold	Drøftet oppstartsplan, og skrev under på nødvendige avtaler.
Notater	<ul style="list-style-type: none">● nytt møte fredag 19. Januar 14:30● Signerte med Gitek for skjemaene vi fikk utdelt av skolen. Denne må signeres av kursansvarlig(?), og skal sendes tilbake til Khai signert● Fikk låne mappe med screenshots av det eksisterende utdaterte systemet. Vi ble bedt om å levere denne tilbake på senere tidspunkt● Fikk utdelt kravspesifikasjon som var laget av Gitek basert på kundens forespørsler. Denne var satt opp for å gi oss en oversikt.<ul style="list-style-type: none">○ Vi står fritt til å gjøre ny sortering på hvilken rekkefølge vi ønsker å utvikle de funksjonelle kravene.○ Vi står også fritt til å velge hvordan brukergrensesnittet ser ut● Avtalt at det sendes ut en agenda til Khai over hva som skal diskuteres på møtene, én dag i forveien<ul style="list-style-type: none">○ Dette avtalte vi også å gjøre med Frode, slik at han har mulighet til å stille mer forberedt til møtet.● Vi skal sende Khai en kopi av planen vi utarbeider i Januar, med gantt diagram ● I etterkant av møtet drøftet vi vagt temaet rundt medlemmers tilgjengelighet for kommunikasjon utenfor arbeidstid, særlig i

	forbindelse med helg. Skal oppdatere reglene i forhold til dette.
--	---

Møtereferat møte med Gitek - 18.01.2018

Tittel	Møte med Khai i Gitek lokalet
Tidspunkt	15:00 til 16:00 18.01.2018
Oppmøte	Hele gruppen, Khai
Innhold	Khai hadde ikke mulighet til å ha møte på fredag som avtalt, så det ble pusket til torsdagen. Fikk drøftet punktliste fra agendaen og så på eksisterende løsning for oppgaven
Notater	<ul style="list-style-type: none"> ● Spørsmål vi sendte i agenda dagen før <ul style="list-style-type: none"> ○ Aktører i programmet og deres privilegier <ul style="list-style-type: none"> ■ gitek admin ■ skogfrøverket internbruker <ul style="list-style-type: none"> ● skogfrøverket registrerer attributter og informasjon om data ■ enkeltbrukere knyttes mot organisasjoner? <ul style="list-style-type: none"> ● andre brukere enn skogfrøverket er eksterne bruker ○ planteskole <ul style="list-style-type: none"> ■ organisasjon/privatkunde <ul style="list-style-type: none"> ● kan kjøpe/selge ● adresse ● org.nr ○ Fargevalg, eksisterende fargepalett <ul style="list-style-type: none"> ■ Står fritt til å ta avgjørelse på dette på egenhånd. Ønsker å se hva vi klarer å komme opp med ○ Noen nærmere dato for når vi kan få API <ul style="list-style-type: none"> ■ jobber fortsatt med dette ○ Lo-Fi prototyper <ul style="list-style-type: none"> ■ ønsker gjerne å se prototyper vi lager underveis ○ Operasjonelle krav <ul style="list-style-type: none"> ■ internet explorer 10 ■ 99.9% oppetid ○ Sikkerhetskrav <ul style="list-style-type: none"> ■ Gitek tar seg av sesjonshåndtering og sikker innlogging ○ Skal vi ta spesifikt hensyn til Universell utforming <ul style="list-style-type: none"> ■ Deres systemer tar hensyn til mye ■ Greit å legge til rette for fargesvake ○ Lisensiering av tredjepartsverktøy(Bootstrap) <ul style="list-style-type: none"> ■ står fritt til å bruke MIT-lisensierte verktøy ○ Verktøybruk bedriften bruker som vi burde være kjent

	<ul style="list-style-type: none"> med(linter, testing verktøy osv) <ul style="list-style-type: none"> ■ Microsoft SQL ■ UTrack ● Oppfølgingsspørsmål <ul style="list-style-type: none"> ○ Presenterte branching modell <ul style="list-style-type: none"> ■ positiv til det. nevnte gitflow
--	---

Møtereferat møte med Gitek - 26.01.2018

Tittel	Møte med Khai i Gitek lokalene
Tidspunkt	14:30 til 15:15. - 26.01.2018
Oppmøte	Hele gruppen, Khai
Innhold	Møtet med Khai som vanlig. Går igjennom agenda sendt ut dagen i forveien
Notater	<p>Her er ting fra agendaen vi sendte ut. Kommentarer fra Khai til punktene på agendaen er markert grønt</p> <ul style="list-style-type: none"> ● Plan mot påskeferie <ul style="list-style-type: none"> ○ Jobber mot å få ferdig funksjonelle krav før ○ refaktorering/tilskudd/bugfixing/finpussing etterpå <ul style="list-style-type: none"> ■ Dette er OK ● integrasjon med deres system <ul style="list-style-type: none"> ○ kan vi få se på API enda ○ Hva må til for å kunne integrere <ul style="list-style-type: none"> ■ De har søkt om reglement rundt om vi får lov til å bruke deres systemer. ■ Bruke deres AJAX moduler, og innloggingsmoduler ■ Bruker ISS.(microsoft programmeringsstack) ■ ikke jobbing mot integrasjon fra starten av ■ De kan gi oss eksempelstruktur på data, AJAX-kall osv <ul style="list-style-type: none"> ● Vi må justere strukturen på data senere for at det skal være passende ○ Designe løsning som må refaktoreres senere for å tillate integrasjon ● Pattern <ul style="list-style-type: none"> ○ Vi lurer på om det er fornuftig å bruke MVC, eller om dere har noen forslag <ul style="list-style-type: none"> ■ Positiv til MVC ● Planteskole

	<ul style="list-style-type: none"> ○ Hadde vært fint å få en skrevet definisjon vi kan forholde oss til, fortsatt ikke helt tydelig <ul style="list-style-type: none"> ■ Planteskole er som enhver annen kunde ● Skogfrøverket <ul style="list-style-type: none"> ○ stiftet 1995 eller 1996 <ul style="list-style-type: none"> ■ 1995 wikipedia ■ 1996 på hjemmesiden <ul style="list-style-type: none"> ● Vi forholder oss til 1996 ● Skal vi jobbe på lokalserver eller har dere noe eget mtp. integrasjon <ul style="list-style-type: none"> ○ XAMP? <ul style="list-style-type: none"> ■ Bruker IIS ● Vårt forslag til omstrukturering av modulene <ul style="list-style-type: none"> ○ nye moduler med tidsestimat <table border="1" style="margin-left: 20px;"> <tr> <td>CRM/brukere</td> <td>1 uke</td> </tr> <tr> <td>Intern produktregistrering</td> <td>1 uke</td> </tr> <tr> <td>Ekstern produktregistrering med sertifikater</td> <td>1 uke</td> </tr> <tr> <td>Ordre</td> <td>2 uker</td> </tr> <tr> <td>Analyse</td> <td>1 uke</td> </tr> <tr> <td>Rapport</td> <td>2 uker</td> </tr> </table> <ul style="list-style-type: none"> ○ Ekstern produktregistrering er nesten ikke eksisterende. Sertifikater er en rekke PDFer ● Oppstart 2. Februar <ul style="list-style-type: none"> ○ Setter opp første backlog ○ Begynner med første modul <p>Tilleggsspunkter utenfor agendaen</p> <ul style="list-style-type: none"> ● Bestemt oss for scrum ● sikkerhetsinstrukser fra Gitek 	CRM/brukere	1 uke	Intern produktregistrering	1 uke	Ekstern produktregistrering med sertifikater	1 uke	Ordre	2 uker	Analyse	1 uke	Rapport	2 uker
CRM/brukere	1 uke												
Intern produktregistrering	1 uke												
Ekstern produktregistrering med sertifikater	1 uke												
Ordre	2 uker												
Analyse	1 uke												
Rapport	2 uker												

Møtereferat møte med Gitek - 02.02.2018

Tittel	Møte med Khai i Gitek lokalet
Tidspunkt	14:30 til 15:40 - 02.02.2018
Oppmøte	Hele gruppen, Khai

Innhold	Drøfte agendaen utsendt dagen før, levert sikkerhetsklareringer
Notater	<p>Vi har forberedt noen nøkkelpunkter vi ønsker å ta opp i morgen.</p> <ul style="list-style-type: none"> ● Planteskole/Organisasjoner <ul style="list-style-type: none"> ○ er alle planteskoler organisasjoner <ul style="list-style-type: none"> ■ planteskoler er en type kunde ■ kan være organisasjon/bedrift/privatperson ○ er alle organisasjoner planteskoler ○ har planteskoler/organisasjoner egne brukere i systemet ○ Gjøres bestillinger alltid i planteskolens navn, eller kan enkeltpersoner legge inn ordre <ul style="list-style-type: none"> ■ kunde hos skogfrøverket gir en bruker ■ kan ikke gjøre enkeltbestillinger <ul style="list-style-type: none"> ● må være registrert kunde i skogfrøverket ● enkeltpersoner kan representere org. ● enkeltperson kan bestille som representant for egne tiltak, f.eks gård. ○ skal planteskoler/organisasjoner ha utvalgte brukere i moderator-lignende roller <ul style="list-style-type: none"> ■ Kan implementere hierarki med posisjoner innad i instansen ○ er alle planteskoler liggende i Brønnøysundsregisteret <ul style="list-style-type: none"> ■ Kan vi benytte oss av br.reg. når vi skal legge til organisasjoner ■ skal vi tillate å legge inn planteskoler som ikke er i dette registeret ■ skal vi tillate å legge inn organisasjoner utenom planteskoler ● Hvilken rolle har administrator <ul style="list-style-type: none"> ○ tilgang til alt ○ mulighet til å skifte "rolle" i systemet ○ tilgang til å redigere/slette brukere <ul style="list-style-type: none"> ■ aldri sletting av brukere utenfor ekstremtilfeller ■ bruker settes heller som inaktive/deaktiverte <ul style="list-style-type: none"> ● bruker har da ikke mulighet for innlogging ■ ● Vi har forberedt en backlog med fokus på første modul(CRM/brukere) ● Sende ut autogenerated mails <ul style="list-style-type: none"> ■ De har API på epost <ul style="list-style-type: none"> ● for passord og faktura ○ for glemt passord

	<ul style="list-style-type: none"> ○ for fakturaer ○ Aktivere bruker ved registrering ○ sette opp egen noreply eller bruke eksisterende løsning. (API eller credentials) ○ annet? ● Attributter for aktører i systemet <ul style="list-style-type: none"> ○ Opsjonale attributter ○ ID vs epost som primary keys ○ Brukere: data forbi epost, navn, passord ○ Organisasjoner: epost, navn, passord, organisasjonsnummer, adresse, telefon ● Prosjektplan ● Lo-Fi prototyper ● Signert sikkerhetsavtale ● Youtrack <ul style="list-style-type: none"> ○ vi får tilgang til Gitek lokalserver ● skal kunne se igjennom løsning uten å være registrert ● sertifikat for frøparti får man ved bestilling ● vant til å søke opp partier på nummer <ul style="list-style-type: none"> ○ kanskje ikke alltid viktig med ikoner ved produktfremvisning ● bruk breadcrumbs(?) ● modal for søk på produkter ● søk + filtrering ● prøv å ikke overbelaste velkomstsiden ● fint med handlevogn <ul style="list-style-type: none"> ○ kun være synlig som ekstern kunde
--	---

Møtereferat møte med Gitek - 09.02.2018

Tittel	Møte med Khai i Gitek lokalet
Tidspunkt	14:30 til 15:30 - 09.02.2018
Oppmøte	Hele gruppen, Khai
Innhold	Drøfte agendaen utsendt dagen før, sprint 1, ny backlog
Notater	<ul style="list-style-type: none"> ● Drøfte første sprint med modul 1 <ul style="list-style-type: none"> ○ ● Vise frem progresjon ● Backlog for modul 2 <ul style="list-style-type: none"> ○ Data som inngår i <ul style="list-style-type: none"> ■ frøparti

	<ul style="list-style-type: none"> ■ kongleparti ■ Registrere spireevne + historikk ■ Beregne pris + tilgjengelighet ■ Blandemelding ■ geografisk lokasjon på frøparti <ul style="list-style-type: none"> ● ikke lokasjon på kongleparti? ■ sertifikater <ul style="list-style-type: none"> ● Gitek server og database <ul style="list-style-type: none"> ○ Burde få tilgang i løpet av kommende uke ● Bruk av slack kontra epost <ul style="list-style-type: none"> ○ Ok å bruke slack om vi ønsker det ● Forskjell på admin og interne brukere <ul style="list-style-type: none"> ○ Ingen sentrale forskjeller ○ de bruker skjulte verktøy for utviklere for administrator ● Er det noen design guidelines fra Skogfrøverket <ul style="list-style-type: none"> ○ Fargepalett ○ Typografi ○ Ikonpakker <ul style="list-style-type: none"> ■ Foreslår Font Awesome, vi står fritt til å bruke ekstra pakker forbi dette <ul style="list-style-type: none"> ● helst uten å måtte referere tilbake til eier av ikonene
Todo	<ul style="list-style-type: none"> ● Rydde opp i release 0.1

Møtereferat møte med Gitek - 16.02.2018

Tittel	Møte med Khai i Gitek lokalet, etterfult av internmøte på skoleareal
Tidspunkt	14:30 til 17:00 - 16.02.2018
Oppmøte	Hele gruppen, Khai
Innhold	Gå igjennom hvordan å benytte seg av databasen
Notater	<ul style="list-style-type: none"> ● Spørsmål til Khai <ul style="list-style-type: none"> ○ Database <ul style="list-style-type: none"> ■ Hvordan gjøre ajax kall til databasen ■ Må vi være koblet til serveren gjennom remote for å gjøre call ■ kan vi laste ned en kopi av databasen til å bruke på egne maskiner

	<ul style="list-style-type: none"> ● for å kunne teste queries uten å måtte være tilkoblet til server ● nei det kan vi ikke <ul style="list-style-type: none"> ○ men vi kan se på databasen i f.eks datagrip ■ Hvor finner vi god dokumentasjon ■ Må ha vår kode liggende på serveren for å teste interaksjon med database ■ se på “fjernet etter Giteks ønske” url i remote <ul style="list-style-type: none"> ● gjennom inspector <ul style="list-style-type: none"> ○ gå på network og se på request payload ○ Libraries <ul style="list-style-type: none"> ■ kan vi bruke MIT lisensiert (innebærer at vi må legge på melding i kildekoden) <ul style="list-style-type: none"> ● jspdf ● De har eget bibliotek for generering av pdf <ul style="list-style-type: none"> ○ se slack
--	---

Møtereferat møte med Gitek - 23.02.2018

Tittel	Møte med Khai i Gitek lokalet
Tidspunkt	14:30 til 15:15 - 23.02.2018
Oppmøte	Hele gruppen, Khai
Innhold	Diskutert databasetrøbbel
Notater	<ul style="list-style-type: none"> ● Laste inn pmws som ekstern bibliotek i IDE <ul style="list-style-type: none"> ○ dette vil gi oss forslag til utfylling ● Dataform <ul style="list-style-type: none"> ○ Fikser put/post automatisk ● Blandemeldinger er bare flere frøpartier ● Order by descending fungerer ikke <ul style="list-style-type: none"> ○ må lage workaround ● JSONstringify ● subqueries i filter parameteret <ul style="list-style-type: none"> ○ String skal være i array ● Mulighet for å få lokal server
Interne notater	<ul style="list-style-type: none"> ● Håvard skal ta med kjeks på neste møte <ul style="list-style-type: none"> ○ ikke jaffa <ul style="list-style-type: none"> ■ eller skript

Møterefereat møte med Gitek - 02.03.2018

Tittel	Fredagsmøte med Gitek
Tidspunkt	14:30 til 15:20 - 02.03.2018
Oppmøte	Martin S, Håvard, Vetle, Khai, Daniel.
Innhold	Møtet med Khai. Gikk gjennom hva vi hadde gjort denne uken, og hva vi skal gjøre neste uke.
Notater	<ul style="list-style-type: none"> • Ikke møte med Khai neste fredag. Blir flyttet til onsdag i stedet - gruppen må avtale et klokkeslett. • Ny dataform update - vi får den snart (Khai skal oppdatere den neste uke). Har dynamisk søk og er enklere. Gruppen må følge med på det som oppdateres og endre det vi har til å matche update'en. • Vist navbars til Khai. <ul style="list-style-type: none"> ○ "breadcrumbs er bra" - Khai ○ Khai liker 3 (svart) sin oversikt, 5 sin font, 6 sin animasjon ○ liker bakgrunnsbilde ("skau") i sidebar • alle filer må være encoda som ISO filer for å få vist æ,ø,å • Gitek mangler data for Ordre modulen - lag grensesnitt i stedet • Lokalmiljø neste uke? kanskje. Khai skal avtale tid med Tor • Skogføverket har godkjent at vi kan ha lokale kopier av deres data (db) (vi skulle orginalt streame deres data). "De ønsker gode resultater på grunn av dette. ha ha ha" - Khai

Møterefereat møte med Gitek - 07.03.2018

Tittel	Møte med Khai i Gitek lokalet
Tidspunkt	14:25 til 15:00 - 07.03.2018
Oppmøte	Hele gruppen, Khai
Innhold	Drøfte agendaen utsendt dagen før, diskutere ordremodul og annet div.
Notater	<p>Vi har forberedt noen nøkkelpunkter vi ønsker å ta opp i morgen.</p> <ul style="list-style-type: none"> • Registrering av brukere <ul style="list-style-type: none"> ○ Hvilke funksjoner burde vi benytte for registrering av brukere ○ Dataform ○ epost, passord er eneste nødvendige <ul style="list-style-type: none"> ■ Evt. hvordan legge til data for flere felter enn de som er listet opp som parametere i

	<p>funksjonsdefinisjonen, f.eks. passord og etternavn.</p> <ul style="list-style-type: none"> ○ Passord <ul style="list-style-type: none"> ■ Glemt passord funksjon (del av pmws) ● Tabell <ul style="list-style-type: none"> ○ Kan vi bruke datatables bibliotek (er det kompatibelt) ● utf8 <ul style="list-style-type: none"> ○ De bruker noe eget i18x(?) <ul style="list-style-type: none"> ■ se biblioteket ■ se hjemmeside ○ Alle filer må være encodet som iso
--	--

Møterefertat møte med Gitek - 13.03.2018

Tittel	Møte med Khai i Gitek lokalet
Tidspunkt	14:30 til 15:35 - 13.03.2018
Oppmøte	Hele gruppen, Khai
Innhold	Vise fremgang, og diskutere plan fremover
Notater	<ul style="list-style-type: none"> ● Vise hva vi har så langt <ul style="list-style-type: none"> ○ PDF-generering <ul style="list-style-type: none"> ■ Tar automatisk window location ○ Organisasjonsregistrering og vedlikehold <ul style="list-style-type: none"> ■ Legg på hoverhånd der det er mulig å redigere for employees i css. ■ Litt mye gui-elementer <ul style="list-style-type: none"> ● Vanskelig å holde overisket med mange nested accordions ● bedre datapresentasjon ● Litt færre nivåer i dybden ○ Glemt passord-funksjon <ul style="list-style-type: none"> ■ Spør khai om funksjonsnavn ■ tar brukernavn så får man resetlink på epost ○ Erstatt alle steder "key" verdier er brukt i brukergrensensnittet ○ Brukertester <ul style="list-style-type: none"> ■ Hva burde vi fjerne fra løsningen før vi brukertester? <ul style="list-style-type: none"> ● Underkriftsdokument ○ Fjen admin-tannhjul nede til venstre når de har mobile viewport på hovedsiden

Møtereferat møte med Gitek - 16.03.2018

Tittel	Møte med Khai i Gitek lokalet
Tidspunkt	14:30 til 15:15 - 16.03.2018
Oppmøte	Martin S, Martin T, Vetle og Khai
Innhold	Vise fremgang og drøfte planen fremover
Notater	<ul style="list-style-type: none"> ● Vise frem det vi har gjort og ikke gjort så langt ● Sertifikater <ul style="list-style-type: none"> ○ pdf blir annerledes enn html ○ må bruke bootstrap uten print query, så skal layouten ordne seg ● Ordre <ul style="list-style-type: none"> ○ Mer informasjon? ● Endringer i dataform <ul style="list-style-type: none"> ○ Dropdown? (lik select i html) <ul style="list-style-type: none"> ■ norsk_navn på frøparti ■ kommuner(?) ■ postnr/poststed ■ skal fungere med dataform ○ hva er nytt ○ hva må byttes ut ○ kan vi bruke onchange event på input-felter? <ul style="list-style-type: none"> ■ sjekk etter listeners i chrome inspector ■ unbind eksisterende onchange listeners ● bla frem og tilbake i records automatisk ● gjenbruke forms (slipper å tømme dom) ● autocomplete (søk) ● automatisk validering ● tillater ikke spamming av tilbakkeknapp ● ajax defineres ved oppretting av form ● definer dataform så open ● insert for å oppdatere eksisterende records ● Muligens pushe analysemodul foran ordremodul <ul style="list-style-type: none"> ○ khai anbefaler å vente med implementasjon av ordremodulen. ● Hard sletting av frøparti ● analyse består av analysekort som kan redigeres etter hvert ● Analysekort

	<ul style="list-style-type: none"> ○ laste opp innscannet papirversjon ○ lage digitalb
--	--

Møtereferat møte med Gitek - 23.03.2018

Tittel	Møte med Khai i Gitek lokalet
Tidspunkt	14:30 til 15:40 - 23.03.2018
Oppmøte	Hele gruppen, Khai
Innhold	Status før påskeferie
Notater	<ul style="list-style-type: none"> ● Vise det vi har så langt ● Ting vi ikke har på plass <ul style="list-style-type: none"> ○ Organisasjoner <ul style="list-style-type: none"> ■ registrering ■ redigering ■ Legge til brukere i org & omvendt ○ Ordre <ul style="list-style-type: none"> ■ pushet til slutt ● Analyse <ul style="list-style-type: none"> ○ Hvordan det skal håndteres i database ○ kan vi laste opp bilder av analysekort som erstatning <ul style="list-style-type: none"> ■ hvor lagre filen ■ Ta bort felter som ikke kan lagres i databasen enn så lenge ● Er dere tilgjengelige på slack de neste to ukene? <ul style="list-style-type: none"> ○ Khai er ikke tilgjengelig i påske <ul style="list-style-type: none"> ■ men han er her mandag+tirsdag+onsdag, så vi kan få tak i han på slack ■ Kommer tilbake tirsdag etter påske ● Rapport <ul style="list-style-type: none"> ○ Hva skal inn ○ Frøparti <ul style="list-style-type: none"> ■ analyserapporter ■ prisrapporter ■ prisutvikling ■ Frø på forskjellige breddegrad ■ hvor er det størst/minst spireevne ■ lag div... ○ biblioteker <ul style="list-style-type: none"> ■ pmws.export på pdf

	<ul style="list-style-type: none"> <ul style="list-style-type: none"> ■ highchart <ul style="list-style-type: none"> ● Norsk graf-generering ● Ordre <ul style="list-style-type: none"> ○ “på plass” betyr når khai har begynt på det ○ Om vi ser på det jobber vi rett på produksjonsmiljøet ○ Khai kan gi oss noe informasjon om det på neste møte etter ferien ● Frøparti <ul style="list-style-type: none"> ○ Sorter på sist brukt i ajax ○ hent ut spesifikk mengde ● Ekstra bemerkninger <ul style="list-style-type: none"> ○ egen bootstrap del i registrert pmws.ini fil under common ○ testing <ul style="list-style-type: none"> ■ https://jasmine.github.io/ (ser attraktivt ut) ■ https://www.blazemeter.com/ (mer omfattende)
--	---

Møtereferat møte med Gitek - 04.04.2018

Tittel	Møte med Khai i Gitek lokalet
Tidspunkt	14:30 til 15:45 - 04.04.2018
Oppmøte	Hele gruppen, Khai
Innhold	Diskutere agenda, kansellert ordre-modul og fremgang
Notater	<ul style="list-style-type: none"> ● Vise fremgang i løsning og diskutere hvilke endringer som skal til for ferdigstilling. <ul style="list-style-type: none"> ○ Nye rapporter ○ Ordre modul <ul style="list-style-type: none"> ■ Oversikt over frø ■ Anbefalte frø ■ Handlekurv ■ Fakturahistorikk ○ Oppdaterte tabeller <ul style="list-style-type: none"> ■ Oversikt over frøparti ■ Oversikt over brukere ■ Oversikt over organisasjoner ○ Brukertest, erklæring <ul style="list-style-type: none"> ■ taushetsplikt ■ formål ■ hvilke data hentes inn <ul style="list-style-type: none"> ● hva brukes det til

	<ul style="list-style-type: none"> ■ sjekk skolens maler ● Planen videre mot 16.05 ● Informasjon om hva som ikke kan nevnes i bachelor-rapporten <ul style="list-style-type: none"> ○ nei <ul style="list-style-type: none"> ■ kildekode fra gitek-moto ■ sikkerhetsinstrukser/-implementasjoner og lignende ■ arkitektur, både logisk og fysisk ■ hardware ■ kundespesifikke(skogfrøverket) punkt om software, hardware, organisasjon og lignende ○ ja <ul style="list-style-type: none"> ■ ajax payload og request, f.eks. filter ■ kundenavn(skogfrøverket). på generell basis ■ Om Gitek og dets systemer ■ Om Skogfrøverket og dets systemer ● bestnr = bestandsnummmmer ● tabell organisasjoner <ul style="list-style-type: none"> ○ rett inn i brukerredigering når man ser på bruker ○ Få vekk animasjon med at tabell ekspanderer fra venstre når accordion åpnes på employee
--	--

D.1.2 Møtereferater i fra møter med veileder

Møtereferat møte med Veileder - 11.01.2018

Tittel	Første møte med veileder
Tidspunkt	10.00-11.00 - 11.01.2018
Oppmøtte	Frode(Veileder), Hele gruppen
Innhold	Mye info ang. bacheloroppgave og planlegging av faste møtetider med veileder
Notater	Ingen valg er faste, kan byttes Husk uansett å reflektere over valg som er gjort/kunne vært Blackboard->bachelor Prosjektavtale(Link under info) Utfyllende info for bacheloroppgaver OOP-sida Forslag til grupperegler Vær spesifikk på gruppereglene

	<p>Eksempler på sanskjeoner: Muntlig, Skriftlig, Kontakt med Frode Statusrapport til Frode rundt 20.02, 01.04 og 01.05 Møtereferat skal være med, husk dato, kl.slett og hvem som møtte Viktig med møtereferat fra møter med oppdragsgiver, send referatene til oppdragsgiver Timeføring per halvtime arbeidet: Dato + timeantall + Hva som ble gjort Prosjektplan, mail til Frode før 01.02</p> <p>Notater om punkter til prosjektplan: Prosjektmål deles i Effektmål og Resultatmål Effektmål: Hva tjener oppdragsgiver på prosjektet? PR, Økonomisk, Rekruttering etc... Resultatmål: Hva blir levert? Løsningen i seg selv. Rammer: Praktiske og Tekniske Praktiske rammer: Fysiske rammer som rom og sted. Tekniske rammer: Eks. spesifikk teknologi eller programmeringsspråk Omfangsdelen får rett inn i ferdige rapporten som del 1.1-1.3 Burde ha størst fokus og være best skrevet Fagområde: Skildre området, jobb fra generelt til spesifikt som en trakt, HVA skal jobbes med. Er en intro til resten av oppgaven og burde gi leseren et godt bilde og lyst til å lese videre. Milepæler: Innleveringer og produktsleveringer Beslutningspunkter: ting som må avgjøres Kan velge fritt mtp. formattering av oppgaven, helst standard 12pkt 1,15 linjeavstand times new roman. Mål og rammer burde være på 2 A4 sder Omfang mellom 2 og 5 A4 sider</p> <p>Kontakt med Frode skjer helst muntlig. Frode er prosessveileder, annen veiledning kan skaffes av fagpersoner på campus/oppdragsgiver.</p>
--	--

Møtereferat møte med Veileder - 18.01.2018

Tittel	etterfulgt av veiledningsmøte med Frode
Tidspunkt	14:00 til 14:40 - 18.01.2018
Oppmøte	Hele gruppen, Frode Haug
Innhold	Hadde et uformelt møte fra 12:15 til 14:00 der vi utformet agenda til Khai og hadde generell diskusjon. Møte med Frode for veiledning angående modell for systemutvikling, roller og timeplan. Møtet med Khai ble pushet til klokken 15 på torsdag, da han ikke hadde mulighet på fredag. Referat fra dette møte kommer i eget dokument.

Notater	<ul style="list-style-type: none"> ● Anbefaler å møte fysisk ofte ● Fint med planleggingsmøte for neste uke etter samtaler med Khai på fredagene ● Lukter scrums av oppgaven mtp hvordan oppgaven er modulbasert ● Frode vil ha en kopi av produktspesifikasjonen ● Greit å kjøre lokale servere
----------------	---

Møtereferat møte med Veileder - 26.01.2018

Tittel	Møte med Frode
Tidspunkt	11:00 til 11:30 - 26.01.2018
Oppmøte	Hele gruppen, Frode
Innhold	Møtet med Frode originalt Torsdag ble pushet til fredag klokken 11:00, da vi sendte agenda litt sent. Skal drøfte hva som burde endres på et midlertidig utkast av oppstartsplanen
Notater	<ul style="list-style-type: none"> ● Opprettelse orrdrbruk og norsk <ul style="list-style-type: none"> ○ 1.1 masse skrivefeil andre avsnitt ○ det heter Javaskript ikke javascript ○ 1.2: enklere utvidelse av ○ 1.2.3 tilegne, ikke tilnærme ○ 1.3 fremviser ikke fremstiller ○ 2.2 mye av dette punktet over i oppgavebeskrivelse ○ 2.3 sammenskrive mer ○ 3.1 ikke vi tar høyde for ○ 3.2 ikke "putter inn" ○ 4.1 ikke "ganske rett frem", "litt lenger inn" <ul style="list-style-type: none"> ■ gjøre om avsnittet for mer formelt språk ○ 4.1.2 hvordan sette seg inn i eksisterende systemer, og hvilken ny kunnskap må vi skaffe oss ○ 4.2.1 ikke "forklaring", heller "presentasjon" ○ 5.3.1 farger på sikkerhetsanalysene ● tidsfrister <ul style="list-style-type: none"> ○ Rapport leveres 16. Mai ○ Presentasjoner i starten av uke 23 <ul style="list-style-type: none"> ■ 2 uke retter rapporten leveres

Møtereferat møte med Veileder - 08.02.2018

Tittel	Møte med Frode på kontoret hans
Tidspunkt	14:00 til 14:40 - 08.02.2018

Oppmøte	Hele gruppen, Frode Haug
Innhold	<ul style="list-style-type: none"> ● Prosjektplanen <ul style="list-style-type: none"> ○ ingen personlige pronomen ○ Punkt 2 <ul style="list-style-type: none"> ■ 2.1 Fagområde <ul style="list-style-type: none"> ● Likte førsteutkast bedre ● Generelt, mer lignende introduksjonen(?) ■ 2.2 Avgrensing <ul style="list-style-type: none"> ● Likte førsteutkast bedre ● viktig at det ikke er begrensning ■ 2.3 Rammer <ul style="list-style-type: none"> ● Punktlisten nederst burde være først ● I funksjonelle krav burde det grupperes; ikke "registrere x, registrere y, registrere z..." <ul style="list-style-type: none"> ○ Heller "registrere <i>mange produkter</i>" ● Nederste del av 2.3 (side 7&8) muligens over til 4.1 ○ Revurdere konsekvenser for reglene ○ Dele opp gantt-diagrammet i uker fremfor dager <ul style="list-style-type: none"> ■ Den er <i>fortsatt</i> uleselig ● Hva burde vi begynner å skrive på planen <ul style="list-style-type: none"> ○ use case ○ user stories ○ Design & modularitet ○ Implementasjon og koding ● Statusrapport 20.(?) februar <ul style="list-style-type: none"> ○ Punkt 1 skal være enkelt utfyllt ○ de andre punktene skal fylles ut bedre ● Frode kan se på rapporten før Lørdag 12. Mai <ul style="list-style-type: none"> ○ Etter dette er han utilgjengelig ○ Fornuftig å forhøre seg med Frode her ○ potensielt Tom ● leselighet av kildekode blir endelig vurdert

Møtereferat møte med Veileder - 08.03.2018

Tittel	Møte med Frode på kontoret hans
Tidspunkt	14:00 til 14:30 - 08.03.2018
Oppmøte	Hele gruppen, Frode Haug
Innhold	<ul style="list-style-type: none"> ● Prosjektstatus <ul style="list-style-type: none"> ○ Arbeidsmetodikk (frem og tilbake) ● Fått satt opp lokalmiljø

	<ul style="list-style-type: none"> ○ videre arbeid ○ utsatt ordremodul ● Hvordan jobbe med rapporten <ul style="list-style-type: none"> ○ Få ned ting som ikke kommer til å endre seg etterhvert i prosjektet <ul style="list-style-type: none"> ■ F.eks biblioteker og andre ting vi må forholde oss til eller benytte oss av
--	---

Møtereferat møte med Veileder - 05.04.2018

Tittel	Møte med Frode på kontoret hans etter påskeferien
Tidspunkt	14:00 til 14:15 - 05.04.2018
Oppmøte	Hele gruppen, Frode Haug
Innhold	<ul style="list-style-type: none"> ● Vil føre timer annerledes mtp. parallelt androidprosjekt <ul style="list-style-type: none"> ○ 40 timer ukentlig men begge prosjektene teller for timesarbeid ● Kan sende iterasjon av rapporten, helst med "ferdigstilte" kapitler/seksjoner slik at frode kan gi verdifulle tilbakemeldinger ● Vektorfigurer til rapport <ul style="list-style-type: none"> ○ Ivar Farup ○ Simon McCallum

Møtereferat møte med Veileder - 12.04.2018

Tittel	Møte med Frode på kontoret hans
Tidspunkt	14:00 til 14:30 - 12.04.2018
Oppmøte	Hele gruppen, Frode Haug
Innhold	<ul style="list-style-type: none"> ● Statusrapport 2 <ul style="list-style-type: none"> ○ "høres greit ut" ○ Hvordan man kunne jobbet annerledes i siste kapittel <ul style="list-style-type: none"> ■ få frem hvordan det var å jobbe med interne systemer og begrensninger fra gitek ■ Hva som har begrenset oss fra å ta optimale / state of the art / best practise avgjørelser <ul style="list-style-type: none"> ● Gått vekk fra standard mvc

Møtereferat møte med Veileder - 26.04.2018

Tittel	Møte med Frode på kontoret hans
Tidspunkt	14:00 til 14:30 - 26.04.2018

Oppmøte	Hele gruppen, Frode Haug
Innhold	<ul style="list-style-type: none"> ● Rapport <ul style="list-style-type: none"> ○ Takk oppdragsgiver før Frode ○ Justere strukturen til å være med tilsvarende den Frode har foreslått <ul style="list-style-type: none"> ■ F.eks kommer kravspec alt for sent, burde være kap. 2 ■ Se også på denne for definisjoner på hva som burde være vedlegg ○ 1.1 Litt enda med high-level introduksjon (gamle systemer er fortsatt mye brukt etc...., teknologi styrer mer og mer..., Internett og online er viktigere...) ○ Analysekapittel(?) ○ Frode vil ha neste utkast før onsdag 2. mai 15:00

Møtereferat møte med Veileder - 03.05.2018

Tittel	Torsdagsmøte med Frode
Tidspunkt	13:10 til 13:20 - 03.05.2018
Oppmøte	Martin S, Håvard, Vetle, Frode.
Innhold	Møte hvor vi gikk gjennom status på rapport pr 03.05
Notater	<p>Neste rapport innlevering onsdag 14:00 + statusrapport use case ikke usecase personas opp innledning på kravspekk Gitek inn på kravspekk merk hvor blidene er hentet rett under bildet</p> <p>Innledning til hvert kappitel Hva skal snakkes om? YouTrack så vi kan se Design=systemarkitektur!! mvc+mønster etc... Til 5 og generelt figurer + koding GENERELT!!</p> <ul style="list-style-type: none"> ● Innledning ● Kronologisk progresjon ● Tenk at folk skal like å lese den!! ● Figurer og bilder
Struktur på kravspec++	1. innledning

	<ol style="list-style-type: none"> 2. kravspec fra gitek+ kommentarer 3. Motivasjon for å endre (Mer utvikler-rettet) <ol style="list-style-type: none"> a. Hva inngikk i prosessen å endre kravspecen b. viktig å dra frem teori 4. Personas 5. user stories 6. use cases 7. evt. ekstra 8. vår endelige kravspec
Design	<ul style="list-style-type: none"> • Notater <ul style="list-style-type: none"> ○ MVC endte opp med å ikke ha controller, og det ble tvunget endring pga server setup og single-page <ul style="list-style-type: none"> ■ Trekke inn template med "moduler" som hentes inn i singlepages ■ Prøve holde flat struktur med få nivåer <ul style="list-style-type: none"> • 1. trykk: mye info • 2. trykk: spesifikk info • 3. trykk: funksjoner relevant for innhold

Møtereferat møte med Veileder - 11.05.2018

Tittel	Møte med Frode på hans kontor
Tidspunkt	13:00 til 13:30 - 11.05.2018
Oppmøte	Hele gruppen, Frode
Innhold	Drøfte rapporten så langt
Notater	<ul style="list-style-type: none"> • Forside <ul style="list-style-type: none"> ○ Frode har henvist til annen mal <ul style="list-style-type: none"> ■ innsida søk: <i>skrive bacheloroppgave ved NTNU gjøvik</i> • Kapittel 1 • Kapittel 2 <ul style="list-style-type: none"> ○ Skriv explicit at vi ikke har lov til å dele funksjonelle krav <ul style="list-style-type: none"> ■ De ville ikke fordi ... ■ Vi gjorde derfor... ○ Rekkefølge: kravspec 2.1, personas 2.2, use case 2.3 ○ Extended use case i tabeller <ul style="list-style-type: none"> ■ use case: aktør, mål beskrivelse ■ use case: hvordan utføre • Kapittel 3

	<ul style="list-style-type: none"> ● Kapittel 4 <ul style="list-style-type: none"> ○ Kartlegg “valg”, “valgte”, “valget”, “drøfte”, “drøftet” osv.. Pass på at vi argumenterer og forteller om alternativ for avgjørelser alle steder. Evt. kom tilbake til det. ● Kapittel 5 ● Kapittel 6 Testing <ul style="list-style-type: none"> ○ Forklar hvordan vi ønsket å gjennomføre brukertesting ○ Hvorfor har vi gjennomført/ ikke gjennomført ting ○ ● Kapittel 7 <ul style="list-style-type: none"> ○ Henvis til tidligere begrunnelse av valg. ○ Lag en slags oppsummering av valgene som er tatt. (Større prinsipielle valg) <ul style="list-style-type: none"> ■ Var valgene gode/dårlige ○ Resultater <ul style="list-style-type: none"> ■ Klart vi å oppfylle målsetninger og krav ● generelt <ul style="list-style-type: none"> ○ Rotere bildene så toppen er mot venstre, i tilfellene de ligger sidelengs ○ Scan bildene, sett inn i rapporten og evt. referer til vedlegg i kapittelets intro ○ Referanser <ul style="list-style-type: none"> ■ Referer til bøker eller større artikler på slutten av dokumentet ■ Referer til nettsider på bunn av siden de brukes ○ Skal leve refleksjonsnotat innen torsdag 31. mai
--	---

D.1.3 Møtereferater i fra gruppens interne møter

Møtereferat internmøte - 10.01.2018

Tittel	Første møtet offisielle møtet
Tidspunkt	19:30 - 20:30 - 10.01.2018
Oppmøte	Hele gruppen
Roller	<ul style="list-style-type: none"> ● Leder: Vetle <ul style="list-style-type: none"> ● Kontakt med Khai/bedrift ● Har siste ord i avgjørelser når det er uenighet eller for å sikre fremgang ● Fler forslag?

	<ul style="list-style-type: none"> • Referent: Martin T <ul style="list-style-type: none"> • Ta referat ved alle møte.
Grupperegler	<ul style="list-style-type: none"> • Loggføre hva som blir gjort etter dager med jobbing • Hver person jobber minimum 30 loggførte timer i uken • Ha det gøy! • Holde god stemning i gruppen. • Holde demokratiet i gruppen levende. Hvis uenigheter får leder siste ord. • (Si fra minimum en dag før om man ikke kan komme på møte?) Møte opp til avtalte møter.
Arbeidstider/ -steder	<ul style="list-style-type: none"> • Jobbe på skolen 2-3 dager i uken (Onsdag/Torsdag/Fredag(etter timene)) • Daglige (man-fre?/lør?/søn?) mini scrum møter (over discord/ på skolen). Ikke satt noen fast tid. Tidspunkt for neste møte avtales i førstkommande møte.
Textchannels Discord	<ul style="list-style-type: none"> • <i>Mail</i>: inn/ut til bedrift • <i>Møter</i>: Møte referat / nye oppsatte møter • <i>General</i>: diskusjon
Mail til Frode	<p>Hei Frode.</p> <p>Vi har fått deg som veileder for bacheloroppgaven vår (Gitek) :)</p> <p>Vi har veldig lyst til å komme igang så fort som mulig og lurte på om du hadde mulighet til å møte oss. Da vi har en del på gruppen som har lang reisetid må vi beregne rundt 1,5-2 timers reisetid før møtet.</p> <p>Mvh Martin Trehjørningen Harestad Håvard Myrvold Johansen Vetle Nikolai Løland Lundsvoll Martin Stenen</p>
Mail til Khai	<p>Hei,</p> <p>Da har det vært kickoff for bacheloroppgaven og vi har veldig lyst til å komme igang så fort som mulig, og lurte på om du hadde mulighet til å møte oss. Da vi har en del på gruppen som har lang reisetid må vi beregne rundt 1,5-2 timers reisetid før møtet.</p> <p>Mvh</p>

	<p>Martin Trehjørningen Harestad Håvard Myrvold Johansen Vetle Nikolai Løland Lundsvoll Martin Stenen</p>
<p>Innhold</p>	<p>Sette opp (G - ett eller annet) graf over planlagte oppgaver og deres estimerte tid</p> <p>Innhold fra foilene brukt på seminar 10.01.2018 <i>“Aktuell mal (men ingen tvangstrøye) for Prosjektplan - frist 28.jan”</i></p> <p>1. MÅL OG RAMMER</p> <ul style="list-style-type: none"> 1.1. Bakgrunn 1.2. Prosjekt mål 1.3. Rammer <p>2. OMFANG</p> <ul style="list-style-type: none"> 2.1. Fagområde 2.2. Avgrensning 2.3. Oppgavebeskrivelse <p>3. PROSJEKTORGANISERING</p> <ul style="list-style-type: none"> 3.1. Ansvarsforhold og roller 3.2. Rutiner og regler i gruppa <p>4. PLANLEGGING, OPPFØLGING OG RAPPORTERING</p> <ul style="list-style-type: none"> 4.1. Hovedinndeling av prosjektet <ul style="list-style-type: none"> - Valg av SU-modell/prosessrammeverk med argumentasjon - Valg av Metode og tilnærming (avklare Teori- og Metodebruk) 4.2. Plan for statusmøter og beslutningspunkter i perioden <p>5. ORGANISERING AV KVALITETSSIKRING</p> <ul style="list-style-type: none"> 5.1. Dokumentasjon, standardbruk og kildekode 5.2. Konfigurasjonsstyring 5.3. Risikoanalyse (identifisere, analysere, tiltak, oppfølging) Teknologi, Forretningsmessig, Prosjektgruppemessig <p>6. PLAN FOR GJENNOMFØRING</p> <p>Det skal lages et Gantt-skjema Liste over aktiviteter (Work Breakdown Structure) Milepæler og beslutningspunkter Tids- og ressursplan (dere avgjør hensiktsmessig detaljgrad)</p>

Tittel	Første ordentlige møte etter oppstart med Frode
Tidspunkt	19:30 - 20:30 - 14.01.2018
Oppmøte	Hele gruppen
Innhold	<ul style="list-style-type: none"> • Fått på plass store deler av reglementet. Skal utarbeides litt mer mtp. gruppemedlemmers tilgjengelighet etter timeplanen fastgjøres • Forbereder spørsmål til Khai for første møtet etter nyttår • Laget dokument for loggføring

Møtereferat internmøte - 16.01.2018

Tittel	Discordmøte hvor vi diskuterer hvordan å lage rapporten
Tidspunkt	16:30 til 18:50 - 16.01.2018
Oppmøte	Hele gruppen
Innhold	<ul style="list-style-type: none"> • Alle fire medlemmene har sett på én gammel bacheloroppgave hver for seg, for å identifisere god og dårlig praksis. Hva er momenter vi kan ta i bruk i vår rapport. <ul style="list-style-type: none"> ○ Vi drøftet disse i fellesskap og diskuterte ting vi skal se på i møtet dagen etter på skolen
Notater	<ul style="list-style-type: none"> • Temaer som burde vurderes <ul style="list-style-type: none"> ○ Burde vi ha roller for ansvarsområder, f.eks. backend/frontend/database ○ legge til læringsmål ○ gjøre sikkerhetsanalyse? ○ vektorgrafikk på modeller ○ ta inspirasjon på GUI fra andre oppgaver ○ tidlig prototyping
Bachelor oppgaver	<ul style="list-style-type: none"> • Oppgaver med relevante GUI-valg <ul style="list-style-type: none"> ○ Gitek bestill ○ Contract Manager ○ Gitek PLU

Møtereferat internmøte - 16.01.2018

Tittel	Gruppemøte skolen
Tidspunkt	13:00 til 17:15 -17.01.2018
Oppmøte	Hele gruppen

Innhold	Prøver å utarbeide ting til startup-planen som er greit å få organisert i fellesskap, med fysisk oppmøte.
Notater	<ul style="list-style-type: none"> ● Valg av branching modell som skal drøftes i et møte med Tom. <ul style="list-style-type: none"> ○ Aktuell modell ● Starter med grunnleggende skisser for utseende <ul style="list-style-type: none"> ○ Ergonomi i utseende <ul style="list-style-type: none"> ■ Generelt <ul style="list-style-type: none"> ● Unngå bruk av scrolling i forms ● stegvis utfylling av forms med trykknbar progresjonsindikator ■ Webdesign <ul style="list-style-type: none"> ● Utnytte bredde ● Statisk banner på toppen av siden, med hovedkategorier ● Relativ meny på venstreside ■ Mobildesign <ul style="list-style-type: none"> ● Utnytte høyde ● Lager et førsteutkast på gantt-diagram som vi kan ta opp med Frode på møte dagen etter. ● Forberede agenda til Frode og Khai ● Drøfte systemutviklingsmodell <ul style="list-style-type: none"> ○ Dette er noe vi vil ta opp med Tom/Frode videre ● Laget en skisse på meny ● Begynte på gantt-diagram, men alt arbeidet ble bortkastet da Visio krasjet ● Neste møte klokken 12:00 torsdag 18. Januar på skolen <ul style="list-style-type: none"> ○ (Etter mobile timen) ○ Etterfulgt av 14:00 møte med Frode ○ Etterfulgt av 15:00 oppfølgingsmøte for å drøfte input fra Frode
Fordelt arbeid	<ul style="list-style-type: none"> ● Distribuert skrivearbeid til punkter på rapporten <ul style="list-style-type: none"> ○ Martin T <ul style="list-style-type: none"> ■ 2.1 & 2.2 ○ Martin S <ul style="list-style-type: none"> ■ 5.3 ○ Håvard <ul style="list-style-type: none"> ■ 1.1 & 1.2 & 1.3 ○ Vetle <ul style="list-style-type: none"> ■ 2.3 & 5.1 & 5.2

Møtereftrat internmøte - 18.01.2018

Tittel	Uformelt møte i klasserommet
Tidspunkt	12:15 til 14:00 - 18.01.2018

Oppmøte	Hele gruppen, Frode Haug
Innhold	Hadde et uformelt møte fra 12:15 til 14:00 der vi utformet agenda til Khai og hadde generell diskusjon. Møte med Frode for veiledning angående modell for systemutvikling, roller og timeplan. Møtet med Khai ble pushet til klokken 15 på torsdag, da han ikke hadde mulighet på fredag. Referat fra dette møte kommer i eget dokument.
Notater	<ul style="list-style-type: none"> • Sendt agenda til Khai for møte dagen etter • Avtalt at Martin S er rapportansvarlig • Nytt møte 19. Januar 15:30 på discord

Møtereferat internmøte - 24.01.2018

Tittel	Møte på skolen
Tidspunkt	13:15 til 14:15 - 24.01.2018
Oppmøte	Hele gruppen
Innhold	<ul style="list-style-type: none"> • Drøfter ting som burde tas opp i fellesskap for å få på plass siste deler av planleggingsfasen. Bla. valgt sysutviklingsmodell, laget gantt.
Notater	<ul style="list-style-type: none"> • bruk av Model View Controller (MVC) ??? <ul style="list-style-type: none"> ○ skal høre med Khai om råd • Fokus på å føre timer til skrijving på rapporten hver uke <ul style="list-style-type: none"> ○ 1 time i uken til rapportskrijving frem mot påske • Påskeferien som stort breakpoint <ul style="list-style-type: none"> ○ Ønsker å bli ferdig med første utkast som dekker det i kravspekken og en slags versjon 1.0 klar. ○ Etter påsken vil rapportskrijving trappes opp ○ Revurdering av prioriteringer • Planlegge oppsett av programmeringsmiljø tidlig februar <ul style="list-style-type: none"> ○ Bitbucket <ul style="list-style-type: none"> ■ mulighet å bruke pipeline, særlig på masterbranch ○ Bruk av testing systemer ○ Bruk av lintere ○ finne ut av IDE-valg <ul style="list-style-type: none"> ■ velge det man liker? ■ alle kjørere samme? ○ Bruke XAMP til lokal server • Systemutviklingsmodell <ul style="list-style-type: none"> ○ SCRUM med variasjoner ○ sprintstart fredager ○ Laget modell for sprint • gjennomgått mye på oppstartsplanen

	<ul style="list-style-type: none"> • refordelt oppgaver
--	--

Møtereferat internmøte - 26.01.2018

Tittel	Møtet på skolen etter møtet med Khai
Tidspunkt	15:15 til 15:45
Oppmøte	Hele gruppen
Innhold	Oppfølgingsmøte for å planlegge frem mot neste ukes innlevering av planen.
Notater	<ul style="list-style-type: none"> • Jobbe målrettet mot å få ferdig rapporten <ul style="list-style-type: none"> ○ Møter på discord mandag og tirsdag klokken 19:30 ○ Første utkast ferdig mandag kveld 19:30 ○ På onsdag <ul style="list-style-type: none"> ■ Prototyping ■ Backlog ■ Planlegge workshop ■ Enighet om verktøy

Møtereferat internmøte - 31.01.2018

Tittel	Møte på skolen før endelig innlevering av planen
Tidspunkt	13:00 til 18:15 - 31.01.2018
Oppmøtte	Hele gruppen
Innhold	Fullførte prosjektplanen, og drøftet prototyping
Notater	<ul style="list-style-type: none"> • Sendte frode prosjektplan og kansellerte møte med han torsdag • Fikk skrevet ned noen tanker rundt prototyping og hvordan privilegier fungerer.

Møtereferat internmøte - 01.02.2018

Tittel	Formøte for møte med Khai
Tidspunkt	13:10 til 19:30 - 01.02.2018
Oppmøte	Hele gruppen
Innhold	Forbereder agenda til Khai, jobbing med prototyper og backlog
Notater	<ul style="list-style-type: none"> • Utarbeidet og sendt agenda til Khai • Sette opp Youtrack

	<ul style="list-style-type: none"> ○ Vi satt opp midlertidig 30 dager trial ● Satt opp phpstorm IDE ● Satt opp bitbucket repository ● Satt opp sourcetree <ul style="list-style-type: none"> ○ konfigurert for GitFlow ● lagde utkast til funksjonalitet, aktører og privilegier ● ordnet med eposter til frode angående gjennomgang av prosjektplanen ● Nytt møte dagen etter 3. Februar 11:00 på skolen <ul style="list-style-type: none"> ○ for å fortsette med prototyping til å vise Khai
--	---

Møtereferat internmøte - 05.02.2018

Tittel	Møte på discord
Tidspunkt	13:15 flytende pga jobbing ved siden - 05.02.2018
Oppmøte	Hele gruppen
Innhold	Drøftet hvordan oppstarten har gått
Notater	<ul style="list-style-type: none"> ● Fordelt arbeidsoppgaver <ul style="list-style-type: none"> ○ Håvard: landing page / ikke enda logget inn ○ Martin S: Post-login landing page ○ Martin T: Headere ● Drøftet litt rundt fargevalg ● styret litt med youtrack ● Fått invitasjon av Khai til <ul style="list-style-type: none"> ○ Intern youtrack ○ slack ● Nytt møte dagen etter 21:30

Møtereferat internmøte - 20.02.2018

Tittel	Referat etter jobbing på skolen
Tidspunkt	12:00 til 17:00 - 20.02.2018
Oppmøte	Martin T, Martin S, Håvard
Innhold	Jobbet med diverse. Spesielt å få orden på repository med serveren.
Notater	<ul style="list-style-type: none"> ● Løst problematikk med bruk av sourcetree <ul style="list-style-type: none"> ○ Må gi hvert gruppelem "full control" i windows privilegier for mappen "D:\www\studentoppgave\GitekReg\www" ● For å få tilgang til prosjektfilene gjennom nettleser på server måtte vi ordne flere ting først

	<ul style="list-style-type: none"> ○ Måtte legge til path for vårt prosjekt i filen "pmws.ini" slik at den står konfigurert som under. Addisjonen er markert i grønt <ul style="list-style-type: none"> ■ <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"> <pre>[Path] Common=d:\www\studentoppgave\GitekReg_ Common;WWW=d:\www\studentoppgave\GitekReg\www;</pre> </div> ○ Måtte legge html filer som skal aksesserer i "D:\www\studentoppgave\GitekReg\www\template" og lage inf-fil med samme navn som har følgende innhold: <ul style="list-style-type: none"> ■ <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"> <pre>"[WebPage] Pagename=*navn_som_går_i_url"</pre> </div> <ul style="list-style-type: none"> ● Viktig: filer på serveren må bootstrap være det siste scriptet som lastes inn <ul style="list-style-type: none"> ○ har ikke funnet ut hvorfor
--	--

Møtereferat internmøte - 21.03.2018

Tittel	Gruppemøte skolen
Tidspunkt	18:30 til 19:00 - 21.03.2018
Oppmøte	Hele gruppen
Innhold	Vårrengjøring av repository
Notater	<ul style="list-style-type: none"> ● Gått igjennom hele repository for å <ul style="list-style-type: none"> ○ Fjerne unødvendige/døde filer ○ Restrukturere for å tilrettegjøre for en mer modulbasert arkitektur. Dette tillater deler av webløsningen å bli eksportert ○ Fjernet alle bilder som er lagt inn, da de ikke har blitt copyright-sjekket ○ Drøftet design av analyse-siden ○ Ordnet importering av skript

D.1.4 Møtereferater i fra møter med ansatte hos NTNU

Møtereferat konsultering prototyper m/ Eivind- 07.02.2018

Tittel	Møte med Eivind Arnstein Johansen på kontoret hans
---------------	--

Tidspunkt	09:00 til 10:00 - 07.02.2018
Oppmøte	Hele gruppen, Eivind
Innhold	Møte for å få veiledning på brukergrensesnitt, prototyping og fargepalett
Notater	<ul style="list-style-type: none"> ● brukergrensesnitt <ul style="list-style-type: none"> ○ grupper for hva brukeren ønsker å gjøre <ul style="list-style-type: none"> ■ hva ser brukeren etter ○ grupper primærhandlinger ○ grupper sekundært ○ trenger ikke bruker ikoner på sidenivåer ○ greit med ikoner på toppen <ul style="list-style-type: none"> ■ ikke fargede ikoner ■ effekt med fokus/highlighting for tab seleksjon/mouse hover ○ pass på å fokusere på det som er vesentlig, etter man er inne i selve løsningen <ul style="list-style-type: none"> ■ fjern bakgrunn etter login f.eks ○ bruk scrolling ○ greit med single page layout for innholdet da det er mindre refreshing <ul style="list-style-type: none"> ■ virker med stabilt ○ former <ul style="list-style-type: none"> ■ når man fyller inn f.eks kategori burde det komme forslag når man begynner å skrive <ul style="list-style-type: none"> ● likt som når man velger fakultet på feide ○ Vurdere om sertifikater ikke trenger sin egen side <ul style="list-style-type: none"> ■ heller være på siden til relatert produkt som en lenke til pdf <ul style="list-style-type: none"> ● f.eks "sertifikat 1, sertifikat 2..." ● prototype ● farger <ul style="list-style-type: none"> ○ kun brukes for å fremheve viktig informasjon ○ liten fargepalett ○ passe på å ikke bruke farge som eneste informasjonsbærer <ul style="list-style-type: none"> ■ mtp. fargeblindhet ● typografi <ul style="list-style-type: none"> ○ basere på brand/merkevare ○ brødtekst ikke noe særlig mindre enn 16px ○ linjehøyde 1.4-1.6 for mer luft ○ ole lund som kilde for typografi ● informasjonsarkitektur <ul style="list-style-type: none"> ○ Kontakt Ole Vatne <ul style="list-style-type: none"> ■ kollega av Eivind ■ for søkefunksjonen

	<ul style="list-style-type: none">○ se bok Information architecture for the world wide web<ul style="list-style-type: none">■ organisere informasjon■ gjøre informasjon tilgjengelig○ Luke wroblewski<ul style="list-style-type: none">■ Kilde for skjemadesign○ Caroline Jarett<ul style="list-style-type: none">■ Kilde for skjemadesign
--	---

Vedlegg E - Time logg

Håvard Lømer	Håvard Innhold	Vedle timer	Vedle Innhold	Martin S Lømer	Martin S Innhold	Martin T Lømer	Martin T Innhold
16.01	4.5 Møte Glikk og samtaler i referat		4.5 Møte Glikk + intern. Leving av tidligere bachelor	2.25 Møte + leing av tidligere bachelor oppgave		2 Møte med glikk og indring av referat og notat	
17.01	4.3 lever bachelor oppgave lesing 1 time møte aktivitet		1 Innomtale	2 møte + start på prosjektpånen		5. Jobb med tidligere bacheloroppgave og møte/referatkriving	
18.01	4.5 Jobbing på skolen		4. Jobbing på skole	4. Jobbing på skolen, startet på gantt		4. Jobbing på skolen	
19.01	3.5 Jobbing på skolen		3.5 Møte Frode + Glikk + Forbundstale + Diskutering etter møt	3.5 møter		3.5 div møter	
20.01	1 møte		2.5 møte + research giffow			2. Jobbing på rapporten og dicordmøte	
21.01	2 research og jobbing		2 Prosjektplan	1 jobbet på prosjektpånen			
Total	17		17.5	13.25		16.5	
22.01	3 arbeid og research		3 arbeid og research				
23.01	3 div arbeid		6.20 Div møter + research loggi	2 Jobbet på prosjektpånen		1 prosjektpånen	
24.01	5 Div møter		4.25 Møling + research + prosjektpånen + kommunikasjonsetup	0.25 Møling		5 Div møter	
25.01	6 Møling		2 Møte Frode + research	2 Møter på skolen		1.75 Møling, og jobbing hjemmefra	
26.01	2 Møter på skolen		3 Research	1.5 Prosjektplan skrivning		3.5 Møte med Frode, Glikk, referat og oppretting på rapport	
27.01	4 prosjektpånen / gitt gjennom prosjektpånen / redigert ++		8 Prosjektplan + research				
Total	23		26.5	11.75		12.25	
28.01	9 Strive om kap 1 + gå over hele dokument og editererrette		1.5 Prosjektplanmøte	3.5 Prosjektplan + vector + bootstrap + møte		3 møte og skrivning	
30.01	8 skrev kap 4 Samskrivning av prosjektpånen		3 Samskriving av prosjektpånen	4 prototyping + samskriving av prosjektpånen		4 Skrivning på rapporten og møte	
31.01	5 Jobbing på skolen		5 Jobbing på skolen	5 Jobbing på skolen		5 Jobbing skolen	
01.02	6 Møte og jobbing		6.5 Møte og jobbing	6.5 Møte + oppsett av div IDE		6 Møte og jobbing	
02.02	1.5 møter på skolen		2.5 møter på skolen	2.5 prototyping + glikk møte		3 Møter på skolen og research rundt designspør	
03.02	4 Møter på skolen		4 MVC + Bootstrap research	3 MVC lesing + rodet i youtube + fikset rotet mitt			
04.02	3		9 Sette opp Bootstrap + Prøving av landing page	3.5 jern og juary research + kamp plan filler laget		2.5 Eksperimenterte med bootstrap	
Total	32.5		30.5	28		23.5	
05.02	7 Logn / Register-view forslag, sett på nav meny + møte		1 møte	4.5 editUser view + editUserController + møte		7 Jobbing med navigasjon på testside, prøving og "nye" felling	
06.02	5 lagde 4 stk forskjellige kontakt oss til preLoginView		1 møte	6 landing page + møte + prototyping mod 2		2.5 Møte + jobbing	
07.02	6 møte Eivind + jobbing skolen + skrive test database boken		6 Møte + jobbing skole + jobbing hjemme	5 møte Eivind + jobbing på skolen + div former		9 Møte eivind + skolejobbing + prøving av generell header	
08.02	10.5 landing page item, JQuery/ajax SPA flux på prelogin / add		9 Jobb på skole + møte + jobb hjemme	5.5 møte Frode + bruker oversiktutvikling		7 Jobbing med forms på skolen/hjemme og møte frode	
09.02	7.5 landing page item, prøve flask trp bakerend + trenger forsa		3 Møte Glikk + v 7.01	2 prosjektpånen v 7.01		6 Skolejobb med 0.1 og møte glikk, ordne med referat	
10.02	6		4 Research i BRN, Planlegger mappestruktur	3 prosjektpånen v 7.01 + like v, bare gruppen			
11.02	2.5 delagte egen kode		6 Leving + iBRN + Fylling av filler	3.5 Startet statusrapport 1, rapport, usecase-diagi		1.5 GUI & form research	
12.02	5 delagte ferdig, logn test og autologn		30	1.5 lesing av prosjektpånen v 7.01 og statusrapport 1		32	
13.02	6 kommersielle kode / ryddet i bruk kode + div		1.5 lesing av prosjektpånen v 7.01 og statusrapport 1	2 Startet implementasjon server		3.5 Les gjennom anbefalte boker og skrevet på rapporten	
14.02	10 research skolebr / lagde 2 forslag til skolebr index2 og innk		3 Møte + Se gjennom status	4.5 møte + servertest + glikk server		7.5 Møte, form, sett på serverfil, sett etter ja lenng for pdf (loggf)	
15.02	1 research skolebr		5 Buffering / Statusrapport	4.5 statusrapport 1 + rapport		2 Rapport, statusrapport	
16.02	5.5 glikk møte + div jobbing skole		8 glikk møte + div jobbing skole + Start sjakk på Ajax + net r	5 glikk møte + div jobbing skole		5 glikk møte + div jobbing skole	
17.02	1 Jobbet med å fikse source tree på server		8 Prøve mer på repo + research ajax + db	3.5 UserOverview + rapport + statusrapport 1			
18.02			6 Research i glikks server system + api + sette opp into fra d	3 startet implementasjon av moduler + rapport		2.5 statusrapport og so på server	
Total	26.5		30	24		20.5	
19.02	5 ryddet i filene på server		5.5 statusrapport 1 + møte med db + rapport skole	5.5 statusrapport 1 + møte med db + rapport skole		8 "Nye" kledning database, finne ut av feiler, undersøke glikk	
20.02	5 Sidebar + CSS		4.5 litt vider filler til å lenke på server + editUser	4.5 litt vider filler til å lenke på server + editUser		5.5 litt vider filler til å lenke på server + sett på database	
21.02	8 Fikset logn til Glikk / gruppearbeid på skolen / møte med k		3 Les opp på tidligere arbeid	5 Glikk møte + ajax/on trossering + nå funks		7 Glikk møte, database, fungerende tabeller for rapport og bruk	
22.02	9.5 Gruppe arbeid på skolen, Cookie extract + GUI		9 server + fiksing av div scripts + glijenknis rese	7.5 server + fiksing av div scripts + glijenknis rese		6 Heret ut data og presenterer, jobber med pullpost	
23.02	5.5 gruppe arbeid, møte og research		8.5 Møte, eksperimentering med postput	5.5 det vi gjorde på skolen		5.5 Møte, pullpost silling, tabell for frontendbruker m. mediger-sak	
24.02	5 navigasjons utvikling / GUI design		3 Jobbe med Dataform klassen	3 research for rapporten + skrivning på rapporten		1.5 prøver å få glikk system for forms si & lungere	
25.02			2.5 Jobbe med Dataform klassen	5 statusrapport 1 + møte med db + rapport skole			
Total	38		26	31		33.5	
26.02	7.5 GUI / sidebar / db		4 Dataform	5 dataform		4.5 Føtt i gang modale til å lungere, sett på pooling	
27.02	8 Templates og GUI research		5 Dataform	1.5 dataform + skrev på rapporten		2.5 Forsøk på dataform + rapportkriving + research	
28.02	9 Ferdigsett Sidebar m/ navbar		10 Skole(Dataform) / Hjemme(pdf research)	5 dataform		9.5 Skole, redigering av rapport, dynamisk listing av forms	
01.03	8 arbeid skole gitt over GUI med gruppen / møte med Frode		7.5 Møte Frode + tabeller og dataform	7.5 Møte Frode + tabeller og dataform		7.5 Møte Frode + Tabeller og dataform	
02.03	10.5 view/User profile m/ ja-file / Simer GUI / CSS		6.5 Møte Glikk + mer dataform report	6.5 Møte Glikk + redigering av brukere + det som i			
03.03	3 GUI		3 Bruker registrering	3 Bruker registrering			
04.03			5.5 Prøver å fikse det at flere dataforms genereres i tabelform	2 regUser + notit i BRN mappefiler		6.5 Delele på innput ferdig, delele brukere delebr i accouser	
Total	46		34	30.5		30.5	

06.03	7 Bootstrap CSS og nav	1 Gruppemøte + design frasesertifikat	2 gruppe møte + reg bruker	7.5 Tabell fropani
06.03	9 6.5 mer på skolen med Martin / sett opp lokal server + jobbing hjemme etterpå	7 Oppsett lokalsystemer + gliemøte + frasesertifikat	7.5 Oppføring oppsettning av lokale systemer	8 Tabell user + databasles
07.03	9 arbeid skole og hjemme. Userprofile / mate Gliek / Sette of	8 Møte Frode + Form jobbing, fjerne av null JSON	5 møte møte + Startet multistep form for fropani	6.5 Skolejobbing og oppsett av server
08.03	6 Møte Frode / jobb grupe skole / jobb hjemme begynne se	6 Sertifikat	4 multistep form	4 Møte med Frode og databasles
09.03	6 Jobbe hjemme med userprofile editing / CSS errors	5 Sertifikat	2.5 multistep form bug fix	
10.03	1 CSS	5 Sertifikat	laptop krasjet For ikke jobbet	
11.03	2 CSS /html på CRM modulen	5 Sertifikat		
Total	40	32	27.5	26
12.03	7 Fikset template klar for CRM - pills / profile-card / profile pk		1.5 rapport skrivning, la til punktene fra Frodes ført	
13.03	2 Forsatte på det JS blen.		4 multiform + skrev også litt på rapporten	8 databasles users på skolen, fropani på kvelden
14.03	7 skole, bugfixing / scripting	5 Skole, Dataform + ferdiggjølling av sertifikat	5 multiform + startet på basic ordre frontend	6 multiform + trøbbel med datatable cas
15.03	9 Oss Gliek jobbet med dynamical page loading / jobbet samr	5.5 Forberedt GUI for Gliek, jobb med Dynamisk lashing Mat	5.5 forberedt ting for møte + møte Gliek	5.5 Møte gliek, multiform, møteprep
16.03	2 la til dynamical upload fropani oversakt / bugfixet litt andre s	5.5 Sertifikat + mer Dynamisk lashing	2 Startet på analyse modulen	
17.03				
18.03				
Total	27	20	18	19.5
19.03	10 Jobbet med upload av profil info uten brukte dataform, lagde	3 analyse	3 analyse	
20.03	8 fikset innsett av blob files til database, css edit	2 analyse + skrev litt på rapporten	2 analyse + skrev litt på rapporten	
21.03	9 skole, cleanup av repo, fikse broken links, laste ned nye bli	7.5 analyse + omstrukturering av mappestruktur +	7.5 analyse + omstrukturering av mappestruktur +	8.5 Våringjølling av repo, møtere, skrive prosjektstruktur
22.03	6 fortsatte med samme som forrige dagen.	5 Mer refaktoring, som onsdagen	5.5 analyse	5.5 refaktorete hele skriptet til multiform
23.03	5 Møte med Kral og, sette inn resterende felt som skal ut og	7 Index refaktoring, møte gliek, Sette opp backlog for 1. uk	6.5 analyse + Gliek møte + plåske backlog	6.5 multiform + gliek
24.03	4 Strippet 11 000 linjer fra Skogverket cas.	6 Dynamisk lashing, testing på hvorfor jscas ikke laster ordi	1.5 analyse	
25.03		6 Mer Dynamisk lashing, Vanskelig bug, databasles legger til t		
Total	42	32	26	20.5
26.03	8 Bugg fixes + jobbet med og fikse cas, nete opp ting som fo			
27.03	6 databasles, begynt å opprette admin kontroller for planter	5 Dynamic loading		
28.03	7 fortsatte med planeskole kontroller, view + edit	8 Nothing works		
29.03	1 planeskole kontroller			
30.03				
31.03	1 databasles research	5 Mer research på dynamic loading		
01.04	2 databasles	3 Prøving på dynamic loading (Fjernet timer som gikk til waste pga dynamic loading...)		
Total	25	21	0	0
02.04	4.5 planeskole / org view + edit	3 analyse - backlog	3 analyse - backlog	
03.04	9 planeskole / org view + edit	mobile	mobile	
04.04	6 jobbet med orgtable	3 Knoding dynamic loading, gr møte, fokus på applikke ført	4.5 Analyse - backlog + Følsakingsforsøk av å fikse	
05.04	9 jobbet med orgtable / møte frode ++	6 Dynamic loading linker O	4 Analyse skaling	
06.04	4 jobbet med orgtable	10 Fix ting som braker pga dynamic loading. -	3 Analyse skaling og merging av versjoner + k	
07.04	3 jobbet med orgtable	10 Mer fix av ting som braker, sette opp lokal på stasjonær,	3.5 lokalsystem testrapport + merge av analyse 2	
08.04		6 Sette opp ferdig lokalsystem, Sertifikat		
Total	35.5	35	18	0
09.04	7 Orgtable	3.5 Statusrapport 2 + analyse index		
10.04	6 Orgtable edit organisation Details / Save details	1.5 analyse		
11.04	6 Orgtable Cms / view user / edit user / Save changes	6 analyse + Startet på highchart		
12.04	6 Org table Cms / view user / edit user / Save changes	5.5 analyse + Startet	6 la lokalt servermiljø opp og på igjen	
13.04	5.5 Org table add function done	10 Sertifikat fra helvete, dynamic loading, skolejobbing	12.5 skole + lage grafier hvor brukeren definerer verdier	
14.04	4 Research bootstrap 4 modal / debugging modal test	10 Dynamic loading, møte Gliek, sertifikat	3 databasles egne verdier	
15.04		8 Enda mer sertifikat		
Total	36.5	40	21.5	21.5
16.04	8 jobbet med forandre tables, popup modal on user click,	5 rapport + analyse fix	5.5 fullført tabell med filter	
17.04	8 fikset edit modal / debugging av sidebar / + mer debugging	4 rapport + analyse	6 Charts for salg med filter	
18.04	8 sett på databasles med card, orgtable fikse og sidebar (jobb	4 analyse	6.5 Charts for salg med filter	
19.04	7 skiver om userstable	4 analyse + rapport (+ 1.5 time android)	5 Kalender for chart og refaktoring	
20.04	4 userstable js	8 Sertifikat, Gliemøte	6 databasles export og responsivitet	
21.04	2 userstable cas og js	9.5 Rapport	3 Skrev på rapporten	
22.04	3 userstable / index fix scrolling bugg / sidebar html+css for user done / and pageload fix	0.5 Møte	0.5 møte	
Total	40	30.5	27	29.5

Vedlegg F - Oppgavebeskrivelse

Oppdragsgiver: Gitek AS
Kontaktperson: Khai Van Ngo
Adresse: Teknologiveien 22. 2815 Gjøvik, C-bygg 2. etasje (C237-C240)
Tlf: 98 85 19 44
Epost: kvn@gitek.no



Gitek REGISTER

Nøkkelord: ordresystem, webapplikasjon, reponsivt design

Gitek AS leverer skreddersydde applikasjoner til bedrifter som ønsker bedre innsikt i driften gjennom dataanalyser, som igjen er med på å effektivisere den daglige driften.

Oppgavebeskrivelse

Det har blitt forespurt et system fra en av våre kunder, som kan håndtere og automatisere flere aspekter ved registrering og bestilling av frø på ulike treslag fra både innland og utland. Systemet skal samle og effektivisere hele prosessen med å sanke, teste, analysere, registrere og selge frø til kjøpere fra hele verden. Systemet skal være tilgjengelig på WEB og være tilpasset mobile enheter. Systemet skal utvikles i moduler og leveres modulsvis.

Funksjonalitet:

- Ordresystemfunksjonalitet som kjøp, ordrehistorikk, handlekurv, ordrekansellering
- Register og vedlikeholde frøparti
 - Kvalitet
 - Opphav
- Rapportere på ulike aspekter ved et frøparti
- Lagersystem
- Fakturering

Funksjonalitet som er listet opp er ikke absolutt og viser kun hovedtrekkene. Det må forventes at det tilkommer funksjonalitet i utviklingsfasen og vi ønsker innspill fra studentene.

Gitek vil bistå med veiledning gjennom hele prosessen og vi ønsker at terskelen for å kontakte oss er lav.

Om Gitek AS

Gitek AS har lokaler i C-bygget ved Høgskolen i Gjøvik/NTNU avd. Gjøvik. Dette gir studentene gode muligheter til å besøke oss i våre lokaler. Tidligere gjennomførte bacheloroppgaver har resultert i sommerarbeid og fast ansettelse. En av våre medeiere startet sin karriere i Gitek som følge av en vel gjennomført bacheloroppgave.

Du finner mer om oss her:

Nettside: www.gitek.no

Facebook: gitekas

Vedlegg G - Prosjektplanen

Kommer på siden under

Prosjektplan for Gitek-gruppen

Dato: 13.02.2018

Martin Trehjørningen Harestad - 470217

Håvard Myrvold Johannessen - 769809

Vetle Nikolai Løland Lundsvoll - 471220

Martin Stenen - 471191

Ordliste	2
1. MÅL OG RAMMER	4
1.1. Bakgrunn	4
1.2. Overordnet prosjektmål	4
Få eksisterende digital funksjonalitet over på en moderne plattform.	4
1.2.1 Effektmål	5
1.2.2 Resultatmål	5
1.2.3 Læringsmål	5
1.3. Rammer	6
2. OMFANG	6
2.1. Fagområde	6
2.2. Avgrensning	7
2.3. Oppgavebeskrivelse	7
3. PROSJEKTORGANISERING	9
3.1. Ansvarsforhold og roller	9
3.2. Rutiner og regler i gruppa	9
Rutiner og normer	9
Regler	10
4. PLANLEGGING, OPPFØLGING OG RAPPORTERING	12
4.1. Hovedinndeling av prosjektet	12
4.1.1 SU-modell	12
4.1.2 Moduler	15
4.1.3 Metode og tilnærming	17
4.2. Plan for statusmøter og beslutningspunkter i perioden	18
4.2.1 Statusmøter	18
5. ORGANISERING AV KVALITETSSIKRING	20
5.1. Dokumentasjon, standardbruk og kildekode	20
5.2. Konfigurasjonsstyring	20
5.3. Risikoanalyse (identifisere, analysere, tiltak, oppfølging) Teknologi, Forretningsmessig, Prosjektgruppemessig	21
5.3.1 Teknologi	22
5.3.2 Forretningsmessig	24
5.3.3 Prosjektgruppemessig	25
6. PLAN FOR GJENNOMFØRING	27
7. KILDER	28

Ordliste

For å unngå flertydighet og forvirring rundt betydningen av ord brukt i rapporten, har vi valgt å legge ved en ordliste med definisjoner.

Ord	Definisjon
Oppdragsgiver	Gitek AS.
Kunden	Det norske Skogfrøverk.
Planteskole	Tilhørende organisasjon for brukermasse av systemet.
CRM	Customer relationship management

1. MÅL OG RAMMER

1.1. Bakgrunn

Stiftelsen 'Det norske Skogfrøverk', heretter kalt kunden, ble opprettet i 1996. Kunden har ansvar for å administrere landets frøforsyning, samling av frø, og distribuering av alle sorter frø til planteskoler og skogeiere. For å holde oversikt over daglig drift som salg og varebeholdning har kunden benyttet seg av en tidligere utviklet programvare for å kunne administrere alle arbeidsoppgavene [1].

Denne løsning ønsker nå kunden å modernisere ved å overføre den eksisterende funksjonalitet i programvaren over i en ny og mer modernisert webløsning. I denne anledning har kunden vært i dialog med oppdragsgiver som har fått i oppdrag å levere ny programvare etter kundens ønske. Vår studentgruppe har vært i dialog med oppdragsgiver som har gitt oss mulighet til å utvikle en direkte integrert løsning for dem som videre vil bli levert til kunde når produktet er ferdigstilt. Funksjonalitet fra den eksisterende løsningen må bli erstattet og implementert i den nye web-løsningen[1].

1.2. Overordnet prosjektmål

(basert på [1])

Få eksisterende digital funksjonalitet over på en moderne plattform.

- Digitalisere analoge løsninger som håndholdte loggbøker.
- Bedre tilgjengeligheten til verktøy og dens funksjonalitet.
- Skape merverdi for kunden ved implementasjon av en mer brukervennlig og intuitiv webløsning
- Redusere driftskostnader knyttet til verktøyet
 - Opplæring
 - Installasjon
- Tilrettelegge for enklere utvidelse av funksjonalitet i fremtiden.
- Enklere integrering med eksisterende interne og eksterne systemer.

1.2.1 Effektmål

- Forenkle brukergrensesnittet og brukeropplevelsen ved å benytte seg av kjente patterns og prinsipper.
- Design i henhold til Donald Normans design prinsipper [3] når det gjelder utforming og layout.
- Lett oversiktlig kostnadsoversikt, tid fra bestilling er gjort til varen er mottatt, transportkostnader, anbefalinger og tips vedrørende varen dvs/f.eks: oppbevaring av varen, holdbarhet, bruksområder
- Lettere tiltrekke seg brukere ved webbasert programvare. Kan øke kundemassen fra eksempelvis utenlandske aktører.
- Styrke konkurranseevnen til firmaet.
- Utvikle et program med høy brukervennlighet slik at det blir enklere for brukerne av programmet å utføre daglig arbeidsoppgaver.

1.2.2 Resultatmål

- Utvikle programvaren med direkte integrasjon inn i oppdragsgivers database/server struktur
- Utviklet en web plattform, sikret for kommersielt bruk.
- Lovere en brukervennlig og intuitiv løsning.
- Overføre eksisterende funksjonalitet inn i web-løsningen.
- Lovere programvare med forskjellige rettigheter for ulike brukergrupper
- Utvikle en webløsning med støtte for mobile enheter
- Videreutvikle løsningen forbi de funksjonelle kravene definert av oppdragsgiver

1.2.3 Læringsmål

- Tilegne seg nye kunnskaper på programmeringsspråk, programvare og teknologi som skal anvendes i oppgaven.
- Lovere en bacheloroppgave som møter de oppsatte kriterier.
- Effektiv kommunikasjon og koordinasjon
- Få erfaring innen flere rammeverk og API for webløsninger
- Benytte seg av kjente prinsipper og patterns for å bedre brukeropplevelsen av løsninger
- Jobbe med et mellomledd mellom utviklere og sluttbrukere

1.3. Rammer

Webløsningen tar sikte på å bygges opp rundt HTML5 teknologien, og skal gi støtte til Internet Explorer 10, Chrome og Firefox. Webløsningen skal utvikles opp mot direkte integrasjon av oppdragsgivers backend løsning, og hente data derfra. Dette impliserer at egen utviklet backend til testing ikke blir levert sammen med endelig løsning. Denne skal kommunisere med databasene til kunden.

Systemet skal ha støtte for internasjonalisering, slik at andre språk kan bli lagt til i etterkant. Front-end skal utvikles primært ved hjelp av HTML, CSS med bootstrap og Javascript. Kommunikasjon mellom front-end og back-end skjer ved hjelp av Ajax JQuery, og data skal returneres på JSON format. All dataauthenting kommer fra oppdragsgivers database i Microsoft SQL, ved å benytte spørringer på JSON format med mulighet for direkte spørringer via SQL statements.

Varighet på prosjektet skal gå over en periode på 18 uker, hvor gruppen avslutningsvis skal levere en sammenfattet rapport over arbeidsformen og den helhetlige løsningen. Dette etterfølges av en presentasjon for sensorer hvor gruppen fremviser webløsningen.

2. OMFANG

2.1. Fagområde

For å kunne lage et moderne og responsivt design har gruppen valgt å lage layout for web-løsningen ved hjelp av HTML, CSS, bootstrap og Javascript. For prosjektet vil det være essensielt å lage et mer navigerbart og brukervennlig design. Dette er noe som er en av de største svakhetene i eksisterende løsning. Det å omorganisere den ekstensive funksjonaliteten på en mer intuitiv måte vil være en stor del av oppgaven. Det vil være behov for å følge mer moderne retningslinjer for menneske-maskin-interaksjon. Viktigheten av dette bygges videre opp av at løsningen også skal kunne brukes på nettbrett og mobil, noe som skaper et krav om effektiv plassutnyttelse på flere oppløsninger og størrelser. Det vil være behov for å følge mer moderne retningslinjer for menneske-maskin-interaksjon.

Den andre tungtveiende oppgaven vil være å få implementert god dataauthenting fra Microsoft SQL database. Dette innebærer å få i stand AJAX jQueries som returnerer JSON data. Det er også en mulighet for at gruppen blir nødt til å benytte oss av SQL-statements dersom uvanlig spesifikk data skal hentes ut. Det må følgende også lages en fornuftig datapresentasjon.

2.2. Avgrensning

Gruppen har valgt å integrere vår applikasjon inn i oppdragsgivers systemer. Dette medfører noen avgrensninger i seg selv. Gruppen kan ikke bruke PHP, da dette programmeringsspråket ikke er støttet i det eksisterende systemet. Gruppen skal også ikke lage deler av backenden til systemet, da disse eksisterer i oppdragsgivers system. Grunnet dette har gruppen valgt å avgrense programmeringsspråkene til HTML, AJAX, og JavaScript (og dermed jQuery) for å sørge for full funksjonalitet i oppdragsgivers systemer.

Siden oppdraget går delvis ut på å modernisere kundens eksisterende programvare, har gruppen valgt å avgrense funksjonaliteten til vår applikasjon til å matche, forenkle, og forbedre den eksisterende funksjonaliteten. Om ny funksjonalitet skulle ønskes av oppdragsgiver, vil dette bli tatt i betraktning av gruppen og mulig bli implementert etter at all grunn funksjonalitet har blitt ferdigstilt.

2.3. Oppgavebeskrivelse

Gruppen har fått i oppgave av oppdragsgiver å lage en webapplikasjon for kunden. De har tidligere hatt en løsning lagd i Paradox som er særdeles utdatert og det er vanskelig å navigere seg gjennom systemet uten mye erfaring med systemet. Oppdragsgiver har valgt å sette opp den nye løsningen som en webapplikasjon og det er denne applikasjonen gruppen skal utvikle.

Løsningen vil ha tre brukergrupper:

- Eksterne brukere: Enkeltpersoner og organisasjoner som kjøper og selger frø gjennom kunden. Planteskoler vil i løsningen være en variant av organisasjoner.
- Interne brukere: Kundens ansatte som organiserer, lagrer, registrerer og analyserer frøene samt fakturerer ved salg av frø.
- Administrator: Oppdragsgiver vil administrere løsningen og brukerbasen.

Funksjonalitet som vil inngå i hver av modulene er som følger:

- CRM-modul
 - Registrering, pålogging og endring av personlig informasjon for brukere
 - Registrere og vedlikeholde rapportering på frøsetting
- Produktregister-modul
 - Registrere og endre informasjon som tilhører produkter.
 - Beregne pris og tilgjengelighet på produkter for kunde
- Registerings-modul
 - Opprette, importere og støtte nedlastning av sertifikater for frøpartier
- Ordre-modul
 - Web-shop som støtter
 - Automatisk genererte bestillinger
 - Manuelle bestillinger
 - Legge inn ordre på vegne for kunder
 - Lage rapporter for besilte varer med tilstand
 - Reservasjon av produkter
 - Digitalt fakturasystem med epost
 - Legge inn og behandle kreditnota
- Analyse-modul
 - Registrere og vedlikeholde analyser på produkter
 - Lage analysekort
 - Lage vente
- Planteskole-modul
 - Registrere og vedlikeholde planteskoler, og deres kunder

Gjennom utviklingen av webløsningen vil gruppen ha et fokus på brukergrensesnitt og user experience og vil forsøke å oppnå en god brukeropplevelse ved hjelp av design patterns og prinsipper fra Googles Material Design og Apples Human Interface Guidelines. Gruppen håper å få muligheten til å teste løsningen mot en brukergruppe fra kunden.

Oppdragsgiver har utarbeidet en liste over funksjonelle krav det nye systemet skal møte. Disse kravene utgjør kjernefunksjonaliteten som er fundamental i oppgaven. For oversiktens skyld har de gjort oss tjenesten å kategorisere disse i moduler, men de har gjort oss oppmerksom på at disse modulene er lagd for å vise kunden hva som vil inngå i løsningen og det er ikke tatt hensyn til arbeidsmengden som vil inngå i modulene.

3. PROSJEKTORGANISERING

3.1. Ansvarsforhold og roller

Gruppeleder(Vetle):

- Gruppelederen har ansvar for å styre møter og ha oversikt over agendaer for møtene, både interne og eksterne.
- Gruppeleder vil videre brukes som kommunikasjonskanal mellom gruppen og eksterne aktører for eksempel oppdragsgiver og veileder.
- Det er derfor naturlig og ha rollene gruppeleder og scrum-master som en og samme rolle.

Referent(Martin T.)

- Skal skrive referater fra gjennomførte møter

Rapportansvarlig(Martin S.)

- Skal ha oversikt over den endelige rapporten og kommunisere dette ut mot gruppa.

Det forventes at flere ansvarsområder vil oppstå organisk etterhvert som utviklingen går fremover. Med prosjektets progresjon vil være behov for å delegerer ansvar og fordele roller for å sikre at oppgaver blir utført.

3.2. Rutiner og regler i gruppa

Rutiner og normer

Dette er punkter som oppfordres og bør følges, men som ikke har noen konkrete sanksjoner dersom de ikke følges. Dette er for å oppfordre til det gruppen mener er god praksis for organisering av gruppearbeid.

Ved uenigheter

- Ved uenigheter får leder siste ord. "Leder har siste ord i avgjørelser når det er uenighet eller for å sikre fremgang" - "møte 1" 10.01.18

Informasjonsflyt

- Alle gruppe medlemmer forplikter seg til å informere dersom man føler at man ikke forstår aspekter ved oppgaven/løsningen og man plikter seg samtidig til å sørge for at alle gruppens medlemmer har en god forståelse av oppgaven/løsningen.

Regler

Arbeidsinnsats

- Det forventes at alle gruppedeltakerne jobber med bacheloroppgaven minst 30 arbeidstimer i uken.
- Jobbe på skolen 2-3 dager i uken
 - Onsdager etter behov
 - Torsdag alltid
 - Fredag alltid
- Det må tas beslutning nær ferie om antall arbeidstider skal senkes.

Loggføring

- Hver gruppedeltager skal selv loggføre eget arbeid og antall (halv)timer de jobber.

Disse loggene danner grunnlaget for møtereferat for gruppens interne daglige møter.

Møter og møteplikt

- Hvert møte planlegges og tiden fastsettes på forrige møte.
- Dersom man ikke har mulighet til å møte til avtalt tid skal man si ifra minimum 12 timer før møtestart, unntatt ved uforutsette/ekstreme omstendigheter. Det gis skjønn for problemer ved reisevei, men forventes at det blir gitt beskjed.
 - Må man endre tiden på et møte skal man selv foreslå ny tid.
- Referenten tar notater av alle møter, hvis han ikke er tilstede vil gruppeleder ha dette ansvaret.
- Møter skal kjøres på tross av at enkeltmedlemmer ikke kan stille
 - Daglige møter skal ha minimum 2 medlemmer.

Tilgjengelighet

- Gruppemedlemmer forplikter seg til å være tilgjengelig til å svare på henvendelser fra andre gruppemedlemmer innen 24 timer. Dersom medlemmer er utilgjengelige for lengre perioder enn dette skal det meldes skriftlig til minst 1 i gruppen.
- Ved lengre bortfall (min. 5 dager) må man spørre gruppen om dette vil gå mtp. prosjektets fremgang.

Sanksjoner ved brudd på følgende regler

Dette er reglene gruppen har avtalt å innføre spesifikke sanksjoner for.

- **Arbeidsinnsats og loggføring**
 - Manglende arbeidstimer på ukesbasis overføres til uken etter.
- **Møter og møteplikt**
 - Ta med kjeks på neste møte

Som en mer allsidig sanksjonsmetode, har gruppen valgt å benytte en slags 3-strike regel. Det vil si at det tillegg til opplistede konsekvenser, utdeles en advarsel ved alvorlige og/eller gjentatte brudd. Ved to advarsler skal det settes opp et møte med veileder om hvordan situasjonen skal løses. Etter dette, vil neste advarsel føre til ekskludering fra bachelorgruppen og prosjektet.

Jeg er inneforstått og godtar reglene og rutinene for Bacheloroppgaven Vårsemesteret 2018.

Dato: 31.01.2018

Sted: Gjøvik

Håvard Myrvold Johannessen

Signatur: Håvard Myrvold Johannessen

Martin Trehjørningen Harestad

Signatur: Martin T. Harestad

Vetle Nikolai Løland Lundsvoll

Signatur: Vetle N.-L. Lundsvoll

Martin Stenen

Signatur: Martin Stenen

4. PLANLEGGING, OPPFØLGING OG RAPPORTERING

4.1. Hovedinndeling av prosjektet

4.1.1 SU-modell

Programvareutvikling er en prosess og en aktivitet som dreier seg om å utvikle en programvare. Når man utvikler programvare innebærer det å evaluere, og å utarbeide en programvare modell som er en abstrakt representasjon av hvordan best foreta denne prosessen. I denne anledningen har gruppen evaluert og drøftet flere forskjellige modeller for å finne ut hva slags teoretisk tilnærming og utviklingsmodell som gir prosjektet best grunnlag.

Første utviklingsmodell som ble evaluert var fossefallsmodellen. Dette er en plandrevet utviklingsprosess, hvor man må utvikle plan og struktur for alle prosessaktiviteter før en kan starte å jobbe med dem. Denne strukturen kan inndeles i fem stadier: 1.Krav, analyse og definisjon, 2.System og program design, 3.Implementasjon og enhetstesting, 4.Integrasjon og system testing, og 5.Operativ oppe-gang og vedlikehold. Hver fase skal være klar og avsluttet før neste fase i prosessen kan starte. I praksis vil disse utviklingsstadiene overlappe hverandre og gi tilbakemeldinger med informasjon til hverandre.

Dette handler bl.a. om hvordan arbeid med designet kan avdekke feil ved krav, og hvordan utforming av koden kan avdekke feil ved design delen m.m . Dette viser da til at modellen ikke er en enkel lineær modell, men at hvert stadie gir tilbakemelding til forrige stadiet. Dersom det da oppdages feil kan man gå tilbake for å endre det forrige utarbeidede stadium. Dette er et av nøkkelpunktene for argumentasjonen å velge en plandrevet modell og metodikk. Kostnaden ved å produsere og godkjenne dokumenter ved bruk av smidige utviklingsmodeller kan på sin side vise seg å øke tidsbruken betraktelig, og prosjektarbeidet kan resultere til at tidsplanen rakner. Det er derfor vanlig, ved jobbing i smidige utviklingsmodeller, at feil oppdaget i tidligere iterasjoner som f. eks. krav og spesifikasjoner blir fryst og satt på vent til et senere tidspunkt og man jobber heller videre med neste iterasjon. Dette kan da føre til et dårlig strukturert system. En plandrevet utviklingsmodell vil lønne seg dersom man utvikler en løsning hvor man kan utarbeide en solid plan i oppstarten av prosjektet, og hvor kravspesifikasjonen vil holde seg stabil.

Gjennom flere konsultasjoner og produktspesifikasjon, uttrykte oppdragsgiver sin oppfordring om å bruke smidig metodikk. Den sterkeste argumentasjonen for valg av en

smidig utviklingsmodell er at oppdragsgiver ønsker iterativ modulbasert leveranse av det web-løsningen. Gruppen så det som hensiktsmessig at oppdragsgiver ønske skulle veie tyngst.

Ved modulbasert utvikling vil sannsynligheten for oppkommende problematikk fra tidligere iterasjoner minskes ettersom hver modul er godt definert, og er tydelig klargjort fra oppdragsgiver om hva den innebærer. Noe som støtter opp om at gruppen skal klare å overholde tidsplanen. Inkrementell programvareutvikling er en fundamental del av smidig tilnærming. For mange bedrifter, kommersielle systemer og personalsystemer viser det seg at smidig tilnærming er mer passende enn plandrevet [4].

Å utvikle inkrementelt er ofte mindre kostnadskrevende i det oppdragsgiver mener det er nødvendig med forandringer i system- arkitekturen. Dette medfører at oppdragsgiver kan ta imot moduler på et tidligere tidspunkt for evaluering og tilbakemelding. Slik kan også eventuelle uoverensstemmelser på designvalg innenfor heuristikken endres etter oppdragsgivers ønske.

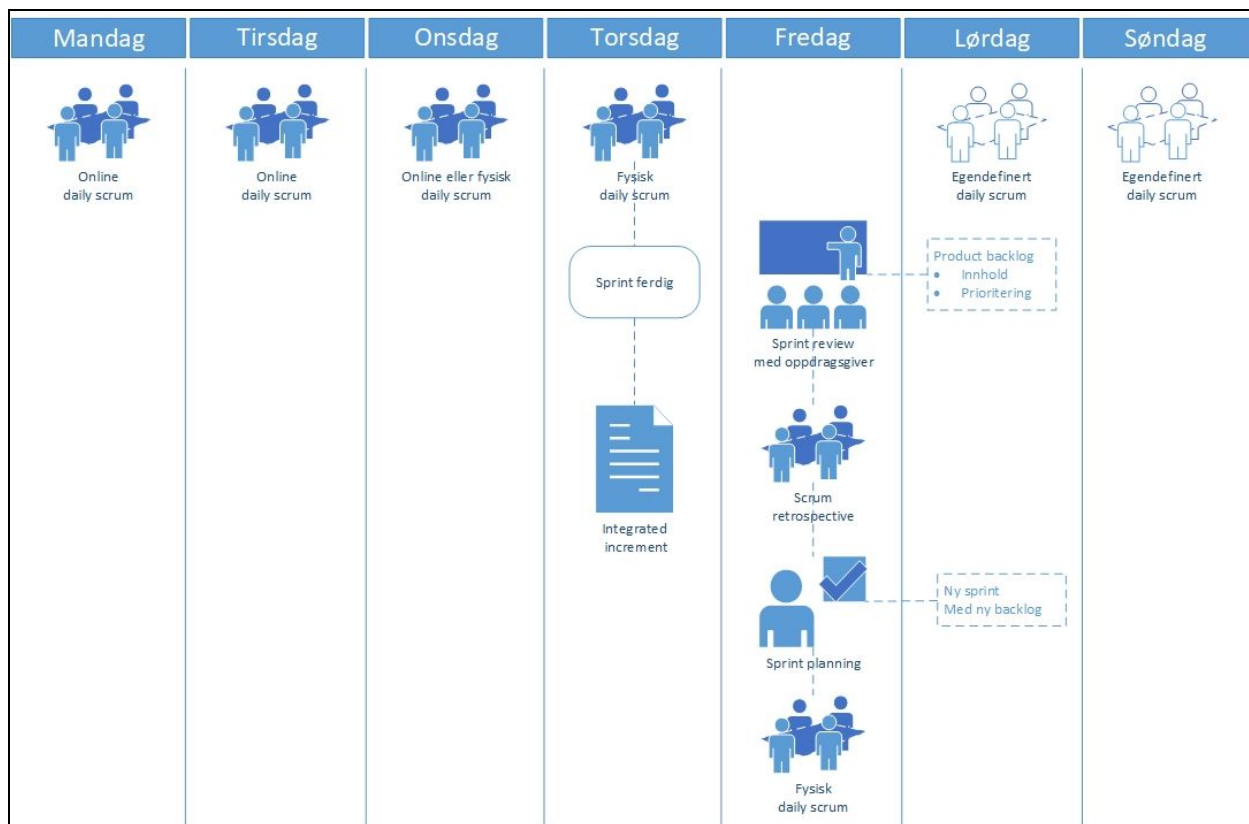
Ettersom valget nå stod på utviklingsmodeller innenfor smidig tilnærming var det tre modeller som ble videre drøftet, Scrum, kanban og Xtreme programming (XP). Etter at gruppen hadde utarbeidet en samarbeidsplan med oppdragsgiver og veileder om ukentlig møteaktivitet, og at systemet skulle utvikles i moduler som strekker seg fra en til to ukers tidsrom, så gruppen det mest hensiktsmessig å jobbe innenfor rammene til Scrum. Her har gruppen lagt alle modulene i produkt backlogen. Noen av modulene blir slått sammen til en sprint, mens de tyngre modulene får stå alene. Tidsrammene per sprint er fastsatt med varighet fra en til to uker. For mer detaljert oversikt se utarbeidet Gantt-diagram.

Ved drøfting av alle smidige utviklingsmetoder fant gruppen styrker og karakteristikk i alle modellene. Flere av disse karakteristikkene har gruppen valgt å inkorporere i vår utviklingsmodell. Gruppen har blant annet vedtatt at i de tyngste modulene, med høyest kompleksitet da skal gruppen anvende par-programmering fra XP. Gruppen har også valgt å ta med noen elementer fra Kanban sin verktøykasse ved arbeid av modulene etter behov, som 'Kanban board', 'user-stories', og 'Epics'. Hver ukers sprint har gruppen konkludert med er mest hensiktsmessig å starte etter ukentlig møte med oppdragsgiver. Derfor anser gruppen at sprinter og ukestart begynner på fredag, etter at gruppen har koordinert informasjon og fylt opp ny sprint backlog.

Sprintene som har to ukers varighet, innebærer at det fortsatt skal holdes møte med oppdragsgiver, for å gi status oppdateringer og foreta videre koordinasjon. andre

februar vil gruppen presentere fullstendig prosjektplan og lo-fi prototyper til oppdragsgiver. Her vil gruppen undertegne konfidensiell avtale med organisasjonen og begynne arbeidet med første sprint.

Tabelloversikten nedenfor viser hvordan sprintene skal foregå på ukesbasis



Kommentarer til modell

- **Online daily scrum** er møter som holdes over internett med talekommunikasjon
- **Online eller fysisk daily scrum** er møter hvor fysisk oppmøte tas etter behov. Dersom møtet kan tas over internett, gjøres det i stedet.
- **Fysisk daily scrum** er møter hvor gruppen stiller opp på campus.
- **Egendefinert daily scrum** er uformelle møter som vanligvis ikke arrangeres, men som avtales etter behov.

4.1.2 Moduler

Basert på gruppens valg av en smidig utviklingsmodell, samt at oppdragsgiver har gitt oss en grundig kravspesifikasjon basert på modulvis levering. Har gruppen derfor valgt å levere vår løsning på samme format, men med noen endringer på modulene.

Våre moduler med tidsestimat:

Modul	Tidsestimat
CRM/brukere	1 uke
Produktregistrering med sertifikater	2 uker
Ordre	2 uker
Analyse	1 uke
Rapport	2 uker

Gruppen flytter alt som har med brukere og CRM å gjøre inn i en modul og vil implementere denne og produktregistreringen først for å kunne takle de større modulene senere. Dette sørger for at gruppen får tid til å bli bedre kjent med løsningen og business-logikken til kunden før gruppen implementerer ordre-, analyse- og rapport-modulene. I tillegg vil de tre siste modulene være avhengig av funksjonalitet som implementeres i CRM- og produktregistrering-modulene.

Selv om funksjonaliteten i kravspesifikasjonen er essensiell i oppgaven og får høyest prioritering, har gruppen gjennom dialog med oppdragsgiver blitt opplyst om at det er muligheter for ekspansjon forbi de funksjonelle kravene. I den sammenheng oppfordret oppdragsgiver oss å komme med forslag og innspill om hvordan oppgavens nivå kan heves.

Mulighetene for utvidelse av oppgaven er noe gruppen tidlig kjørte idemyldring på, for å forsikre at prosjektet er egnet for en gruppe med fire medlemmer. De mest sentrale konseptene gruppen drøftet var potensialet for å utvikle en mobilapplikasjon i tillegg til webløsningen, ekstra verktøy for å lage verdifull statistikk eller gjennomføring av grundigere sikkerhetsanalyser. Alle utvidelser av oppgaven er noe gruppen vurderer som bonus dersom gruppen har tid til overs etter kjernefunksjonalitet er utviklet. Målet er å øke vanskelighetsgraden og muligheten til å vise kompetanse, fremfor å øke størrelsen og mengden arbeid på lavere ferdighetsnivå.

Siden oppdragsgiver har gitt oss klare mål for hva som skal inngå i løsningen har gruppen valgt å prioritere å skape en versjon 1.0 som vil implementere de fem modulene. Deretter vil gruppen ekspandere oppgaven og skape nye moduler. Dette vil også skape et insentiv for å lage en løsning som er lett å videreutvikle. Gruppen vil benytte oss av god dokumentering og patterns som MVC for å nå dette målet.

Gruppen har fått muligheten til å integrere webløsningen inn i oppdragsgivers systemer, noe gruppen ønsker å gjennomføre. Dette betyr at vår webløsning vil kommunisere med oppdragsgivers API og databasesystemer. Som et overordnet mål for denne bacheloroppgaven ønsker gruppen å lage en løsning som oppdragsgiver kan benytte seg av.

4.1.3 Metode og tilnærming

Når det kommer til metode og tilnærming kommer gruppen til å bruke en narrativ tilnærming for innhenting av kunnskap fra kunden gjennom oppdragsgiver, det hadde vært gunstig å kunne snakke med kunden direkte. Det vil si at gruppen vil gjennom arbeidsprosessen ha en tett dialog med oppdragsgiver via ukentlig møteaktivitet og mail. Dette vil da sammen med eksisterende produktspesifikasjon være med å belyse oppdragsgivers erfaringer, synspunkter og tilbakemeldinger på forskjellig funksjonalitet som blir utviklet gjennom modulene. Dette gjøres for å forsikre seg om at arbeidet som blir gjort i løpet av prosjektet er i tråd med oppdragsgivers ønsker og forventninger til produktet som leveres. Denne informasjonen vil da bli benyttet gjennom prosessen til å utforme user-stories og danne grunnlag for ulike system diagrammer som gruppen kommer til å bruke for å bygge opp design og systemarkitektur.

Videre kommer gruppen til å bruke en deduktiv arbeidsmetode gjennom utviklingsprosessen. Med dette menes det å omforme teori lært gjennom utdanningsløpet om til praksis ved utvikling av systemet. Dette vil naturligvis ikke være tilstrekkelig, og det forventes at gruppen må tilegne oss kunnskap om retningslinjer for best practises og state of the art innenfor berørte emner.

De mest sentrale emnene som kommer til å spille en viktig rolle:

- web teknologi (HTML, CSS, Javascript, JQuery og Bootstrap)
- Programvaresikkerhet
- Ergonomi i digitale medier
- Systemutvikling
- Objektorientert systemutvikling

I tillegg til anvendelse av lært teori, må gruppen tilegne seg ny kunnskap om:

- Microsoft-SQL
- GitFlow versjonskontroll

- JetBrains-IDE
- Prinsipper om user experience og design i web-løsninger
- Oppdragsgivers interne systemer
 - Database
 - API

4.2. Plan for statusmøter og beslutningspunkter i perioden

4.2.1 Statusmøter

Statusmøter har blitt inndelt i tre deler. Mandag og tirsdag er noen av gruppemedlemmene opptatt med en deltid utviklerjobb i tillegg til arbeidet med bacheloroppgaven. Dette vil kompenseres med helgejobbing på bachelorprosjektet. Gruppen fant det derfor mest hensiktsmessig å sitte hver for oss og jobbe i hjemmekontorløsning disse to dagene, og unngår følgelig å benytte oss av scrum sine daily standup meetings. For å sørge for at gruppen holdes oppdatert på fremdriften i prosjektet vil gruppen legge til informasjonen som typisk kommer i daily standup meeting på tekstkanaler på internett. Videre vil gruppen jobbe på campus onsdag, torsdag og fredag og har da gode muligheter for fellesarbeid og møter. Gruppen har blitt enige om følgende møtestruktur:

1. Mandag-Tirsdag Blir daglig scrum møter avholdt over nettet.
2. Onsdag-Fredag vil gruppen møttes på campus hvor gruppen vil jobbe med prosjektet og holde statusmøter og videre planlegge neste ukes arbeid. Torsdager vil gruppen ha møter med prosjektveileder. Etter statusmøte på torsdager vil daværende sprint avsluttes (så lenge ikke arbeidet befinner seg i uken mellom en to ukers sprint). Da vil gruppen ferdigstille et fungerende inkrement som kan presenteres til kunden. Fredager avholdes 'sprint-review' med kunden, hvor gruppen går gjennom Innhold og prioriteringer. Etter dette avholdes 'scrum retrospective' møte intern i gruppen for gjennomgang av avsluttet sprint. Det vil også bli avholdt et 'sprint planning', hvor gruppen setter opp ny sprint backlog og markerer starten for ny ukestart.
3. Lørdag-Søndager blir det avholdt statusmøter over nettet. Disse vil bli planlagt på forhånd, og gjennomføres kun etter behov.

4.2.2 Beslutningspunkter

Vedtatte beslutningspunkter for prosjektperioden:

Fra	Til	Innhold
-----	-----	---------

05.0 1	01.02	Utarbeide prosjektplan, oppsett og konfigurasjonsstyring av programvare
01.0 2	08.02	CRM-modul skal produseres.
09.0 2	22.02	Produktregister- og Registreringsmodul skal produseres.
23.0 2	08.03	Ordre-modul skal produseres.
09.0 3	15.03	Analyse-modul skal produseres.
16.0 3	29.03	Rapport-modul skal produseres.
01.0 2	06.04	Alle modulene fra kravspesifikasjonen skal være ferdigstilt, og brukertestet, og videre funksjonalitet ønsket av kunden legges i backloggen for produksjon.
06.0 4	04.05	Hovedtyngden av rapportskrivningen blir produsert, og tilleggsfunksjonalitet blir utviklet for systemet.
15.0 5	16.05	Bacheloroppgaven leveres inn for evaluering.
16.0 5	02.06	Gruppen har muntlig presentasjon av bacheloroppgaven.

5. ORGANISERING AV KVALITETSSIKRING

5.1. Dokumentasjon, standardbruk og kildekode

Gruppen skal som tidligere nevnt utvikle en webløsning som kommuniserer med oppdragsgivers API. Gruppen har derfor valgt å benytte oss av JavaDoc som dokumentasjonsverktøy. Dette betyr at alle klasser, interfaces, constructorer, funksjoner og andre kodeblokker skal være kommentert. Kommentarene skal inneholde tags som @param, @return, @author og andre tags brukt i JavaDoc der det passer. Gruppen vil benytte oss av WebStorm, IDE fra JetBrains, sin innebygde JavaDoc generator. Kildekoden skal være oversiktlig og identifiere skal skrives med camelCase og ha et beskrivende, fornuftig navn.

Gruppen skal benytte oss av model-view-controller mønsteret for å strukturere kildekode til vår løsning. Det betyr at strukturen til websidene som blir vist frem ligger i view, controller vil ha all kildekode som forflytter data og model vil inneholde klasser for å holde på dataen. Controller vil og være delen av løsningen ansvarlig for å koble seg opp mot API'et til oppdragsgiver.

Siden gruppen skal benytte oss av rammeverket bootstrap vil gruppen følge best practises fra w3school. Javascript har en strict mode man kan bruke for å gi feilmeldinger på dårlig kodenstandard som ellers ville tolket uten feil som f.eks. å bruke udeklarte variabler eller skrive til en read-only variabel. IDEen gruppen skal benytte oss av har native støtte for JavaScript Standard Style, en linter som har fokus på kodeformatering. En standardisert og automatisk testing av koden gruppen skriver vil, sammen med kommentering, øke lesbarhet og bedre samarbeidet i gruppen.

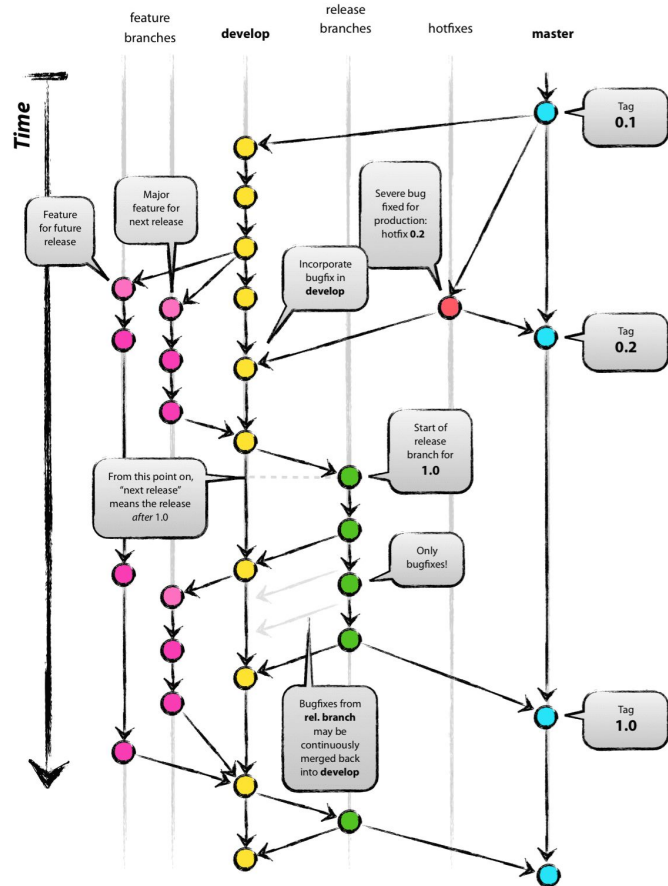
5.2. Konfigurasjonsstyring

Martin T. har tidligere benyttet seg av GitFlow som metode for konfigurasjonsstyring. I tillegg er dette en metode integrert i Atlassian sine produkter og metoden oppdragsgiver benytter seg av. Derfor var det veldig naturlig å benytte seg av denne modellen for vårt prosjekt.

I GitFlow er det to branches som aldri skal slettes; develop og master. For hver nye funksjonalitet som skal utvikles lages det en feature branch. Da forker gruppen ut fra develop branch, utvikler i vår egen feature branch og merger inn med develop igjen når vi er ferdig og koden fungerer lokalt. I develop branchen vil gruppen kun gjøre endring i koden for å rette opp i feil som kan oppstå etter flere merges med feature branches. Før

gruppen skal levere en modul vil gruppen forke ut en release branch som vi vil bruke for å skape en stabil modul og hver modul vil gå inn i master når den er stabil og blir ansett som ferdig. Skulle det være nødvendig med hotfixes etter levert modul vil gruppen forke ut en hotfix branch som skal merges med både master og develop branch. Ved levert modul inkrementeres versjonsnummeret etter første punktum (som eksempel fra 0.1 til 0.2). Mindre endringer som hotfixes vil inkrementere versjonsnummeret på andre punktum (som eksempel fra 0.1.1 til 0.1.2).

Når gruppen har dekket oppdragsgivers funksjonelle krav vil gruppen øke versjonsnummeret til 1.0 og jobbe modulbasert videre med mål om å nå en versjon 2.0. Gruppen ser ingen markant forskjell mellom GitHub og BitBucket men har allerede studentkontoer på BitBucket og vil derfor benytte oss videre av denne løsningen som repository.



5.3. Risikoanalyse (identifisere, analysere, tiltak, oppfølging) Teknologi, Forretningsmessig, Prosjektgruppemessig

Gruppen har valgt å bruke DREAD modellen for å gjøre vår risikoanalyse. Hver risiko deles inn i kategoriene skade, reproduserbarhet, utnyttelse, påvirkede brukere, og oppdagbarhet. Hver av kategoriene fylles så inn med en poengskår fra null til ti. 0 - 3 vil si at det ikke er noen trussel. 4 - 7 betyr at det er en trussel, men at den er mindre farlig/ikke kritisk. 8 - 10 betyr at det er en kritisk trussel [2]. De rutene som ikke er fylt inn anses som å ikke være relevante for den risikoen.

I følge OWASP så medfølger de forskjellige kategoriene dette [2]:

- Skade - hvor mye skade som vil bli påført oppgaven om risikoen skulle inntreffe. 0 = ingen. 5 = Individens data skades. 10 = alle deltagernes data skades.

- Reproduserbarhet - hvor lett er det å reprodusere scenarioet hvor risikoen inntreffer. 0 = veldig vanskelig eller umulig. 5 = ett eller to trinn må gjøres. 10 = veldig lett.
- Utnyttelse - hva trengs for å kunne utnytte risikoen. 0 = En rekke verktøy, samt kunnskaper om prosjektet. 5 = Det finnes tilgjengelige verktøy som gjør jobben. 10 = bare basic verktøy, som f.eks. En nettleser.
- Påvirkede brukere - Hvor mange av deltagerne vil bli påvirket om risikoen skulle inntreffe. 0 = ingen. 5 = individer. 10 = alle.
- Oppdagbarhet - hvor lett det er å oppdage risikoen for angriperen. 0 = veldig vanskelig eller umulig. 5 = Litt jobb (research) kreves. 10 = all informasjonen som trengs er helt tilgjengelig.

5.3.1 Teknologi

De teknologiske risikoene gruppen risikerer å støte på i løpet av hele prosjektet er:

	Risiko	Skade	Reproduserbarhet	Utnyttelse	Påvirkede brukere	Oppdagbarhet
1	Scenarioet hvor gruppen ikke har kontrollen over databasen.	3			10	
2	Rapporter/tekster kan slettes/mistes.	5	5	10	10	10
3	Utviklernes (våres) datamaskiner/data kan slutte å fungere eller bli stjålet.	10	5	10	5	10
4	Kommunikasjonskanalenes servere kan gå ned.	3	1	0	10	10

1. **Kontroll over databasen.**

Siden gruppen velger å integrere vår applikasjon med oppdragsgivers systemer så mister gruppen kontrollen over en rekke data, dette innebærer at gruppen må bruke oppdragsgivers database for vår applikasjon. Oppdragsgiver vil da ha full tilgang til og kontroll over databasen. Realistisk vil dette ikke være et problem da

det er oppdragsgiver som er vår arbeidsgiver og der er liten risiko for at de kommer til å sabotere oppgaven.

2. **Rapporter/tekster mistes.**

Gjennom hele prosjektperioden kommer gruppen til å skrive en rekke dokumenter, blant annet møtereferater, prosjektplanen, og sluttrapporten. Om gruppen skulle miste disse dokumentene så vil det skade prosjektet på flere måter avhengig av hvilke dokumenter gruppen mister. Tap av møtereferatene vil gjøre mindre skade, da disse bare brukes for å dokumentere hva som skjedde i løpet av møtene.

Tap av prosjektplanen er alvorlig siden deler av den skal være med i sluttrapporten. Prosjektplanen skal også inkluderes når den helhetlige vurderingen av prosjektet gjøres.

Å miste sluttrapporten vil være ganske kritisk for gruppen, da det er hovedsakelig denne rapporten karakteren vår vil være basert på. Rapporten kommer også til å være et stort dokument skrevet over en lengre periode, så om gruppen skulle miste den så er det ikke sikkert at gruppen kommer til å rekke å skrive rapporten på nytt.

Gruppen har laget noen forholdsregler for å sikre oss i tilfellet noen av dokumentene våre skulle bli borte. Gruppen har backups på internett, i tillegg til at alle har noe/alt av dokumentene lastet ned på våre maskiner.

3. **Maskiner mistes/stjeles.**

Om en eller flere av våre arbeidsmaskiner skulle mistes eller bli stjålet i løpet i bacheloroppgave-perioden vil dette gjøre skade på to forskjellige måter.

Den første er at mangelen på arbeidsmaskiner kommer til å redusere, om ikke fullstendig stoppe, arbeidsmengden eieren av maskinen kan gjøre. Dette vil så føre til at gruppen som en helhet havner bak skjemaet, og at gruppen må bruke en del tid på å hente oss inn igjen.

Den andre er at om en arbeidsmaskin skulle komme på avveie så vil dette være en stor sikkerhetstrussel for vår applikasjon og oppdragsgivers systemer. Om noen skulle få fri tilgang til en av våre maskiner vil dette gi dem muligheten til å sabotere, kopiere, og modifisere vår bacheloroppgave. Personen vil også få tilgang til deler av oppdragsgivers systemer, som inneholder bedriftshemmeligheter som ikke skal deles med noen.

4. **Kommunikasjonskanal-serverne går ned.**

Om kommunikasjonskanalene skulle bli utilgjengelige vil dette føre til komplikasjoner for gruppen, spesielt siden gruppen jobber hver for oss fire dager

i uken. Scenarioet vil altså ikke føre til store skader, men det er et problem gruppen må håndtere raskt likevel. Måter å løse problemet på er å midlertidig bytte til en annen kommunikasjonskanal, eller å jobbe videre med våre tildelte oppgaver til det neste fysiske møtet.

5.3.2 Forretningsmessig

De forretningsmessige risikoene rundt prosjekt er:

	Risiko	Skad e	Reprodus er-barhet	Utnyttels e	Påvirkede brukere	Oppdagbarh et
1	Universitetet (NTNU) eller oppdragsgiver velger å kansellere oppgaven.	10	0	0	10	10
2	Lekkasje av hemmelig/kritisk data	10	5	0	10	
3	Båndet mellom kunde og oppdragsgiver brytes	1	0	0	1	10

1. Oppgaven kanselleres.

Om bacheloroppgaven kanselleres av enten NTNU eller oppdragsgiver, må gruppen starte på en ny oppgave. Dette vil bety at gruppen ikke kommer til å ha nok tid til å fullstendig fullføre den nye oppgaven. Noe som kommer til å påføre stor skade på gruppens sluttprodukt og karakter.

2. Lekkasje av hemmelig/kritisk data.

En del av avtalen gruppen har med oppdragsgiver er at gruppen ikke skal dele konfidensiell data i fra deres systemer og kontorer. Det vil derfor skade forholdet i mellom arbeidsgiver og gruppen om kritisk data skulle komme på avveie. For å unngå at dette scenarioet skal inntreffe har gruppen inngått en avtale med oppdragsgiver som spesifiserer handlinger som gruppen ikke skal gjøre med arbeidsrelaterte data/maskiner.

3. Båndet mellom kunde og oppdragsgiver brytes

Selv om dette scenarioet er ille for oppdragsgiver, så kommer det ikke til å påføre gruppen noen stor skade. Gruppen kommer til å måtte fjerne kundens navn og logo i fra alle våre dokumenter. Det vil likevel være mulig å fortsette med vår oppgave siden oppdragsgiver er vår arbeidsgiver, ikke kunden.

5.3.3 Prosjektgruppemessig

	Risiko	Skad e	Reproduse r-barhet	Utnyttels e	Påvirked e brukere	Oppdagbarh et
1	Det oppstår uenigheter i gruppen.	2	8		7	5
2	Det oppstår "alvorlige" uenigheter i gruppen.	7	5		10	8
3	En deltager av gruppen bryter gruppe reglene.	4	8		5	
4	En person blir syk, eller havner i en situasjon som gjør så han ikke lenger kan jobbe med prosjektet.	5	5		10	
5	Flere medlemmer stiller ikke til møtene.	2	5		5	
6	Medlemmer av gruppen avslutter studiet tidlig, eller må ekskluderes.	1			7	10

1. Uenighet i gruppen.

Om det skulle oppstå uenighet i gruppen vil det være en midlertidig hindring for gruppen. Disse uenighetene kan raskt løses med god kommunikasjon innad i gruppen. Om de involverte individene ikke finner en løsning på problemet innen en rimelig tid, kommer gruppelederen til å ta en avgjørelse og løse opp i uenigheten.

2. Alvorlige uenigheter i gruppen.

Alvorlige uenigheter er større problemer som truer med å bryte opp gruppen eller å sette gruppen langt bak skjemaet. Siden alvorlige uenigheter er av den mer

sjeldne typen, så kommer de ikke til å inntreffe ofte. Men i de situasjonene hvor de skjer så kommer gruppen til å måtte håndtere disse seriøst. Som med andre problemer hvor gruppen ikke raskt finner en løsning, så skal gruppelederen ta en beslutning for å løse problemet. Når en beslutning har blitt tatt, skal gruppen innrette seg etter dette. Som sagt i punkt 3.2, så skal veileder informeres og inkluderes for å hjelpe til med å løse situasjonen, i de tilfellene hvor en person har gjort alvorlige brudd gjentatte ganger.

3. En person bryter gruppereglene.

Om en person gjentatte ganger bryter gruppereglene kommer dette til å påføre mindre skade på gruppen og prosjektet. Innad i gruppen kommer det til å skade moralen og føre til uenigheter. For prosjektet kommer det til å føre til at gruppen må sette av tid til å ta igjen den tapte tiden, som enten ikke ble jobbet eller ble brukt på å diskutere innad i gruppen.

4. Sykdom.

Om en medlemmer i gruppen skulle bli borte i en lengre periode, grunnet f.eks. Sykdom, kommer dette til å være et problem som setter tidsskjemaet. Resten av deltagerne i gruppen må derfor jobbe ekstra mye for å ta igjen den tapte tiden, i tillegg kommer individet som er borte til å måtte gjøre en del jobb for å sette seg inn i det han har gått glipp av mens han var borte.

Problemet kan delvis løses ved at det gjøres jevnlig samtaler i mellom den manglende personen og resten av gruppen. Disse samtaler kan oppdatere gruppen på statusen til individet, samt at individet kan bli informert om hva gruppen har gjort i hans fravær.

Ved lengre fravær informeres gruppens veileder for råd om hvordan gruppen skal håndtere personens fravær.

5. En eller flere møter ikke til møtene.

Siden gruppen har valgt å bruke scrum, og dermed scrums patterns, så er det viktig for gruppen å ha de daglige scrum møtene for gi statusrapporter for den pågående sprinten. Det vil derfor påføre gruppen mindre skade dersom en person ikke møter opp til disse møtene. Personen som ikke møter opp mister mulig viktig informasjon, samt at gruppen ikke får en oppdatering på den manglende personens status.

Om fler enn en person ikke møter til de daglige scrum møtene har gruppen valgt å avlyse det møtet og flytte det til et annet tidspunkt. Dette gjøres siden gruppen mener at det er poenngløst å ha oppdateringsmøter hvor bare halvparten av

gruppen er tilstedet. I tillegg så trengs det en majoritets stemme for at beslutninger skal gå gjennom. Så om det mangler flere personer må beslutninger flyttes til et annet møte, og da kan gruppen like så godt flytte hele møtet.

6. Medlemmer av gruppen avslutter studiet tidlig, eller må ekskluderes.

I dette tilfellet er skaden avhengig av hvilket tidspunkt i prosjektet dette inntreffer. Dette kommer til å påvirke hvordan arbeidsfordelingen i gruppen organiseres. Det vil bli et behov for å revurdere prioriteringer og oppgaver etter hva som er realistisk å få gjennomført.

6. PLAN FOR GJENNOMFØRING

Gruppen har med oppfordring fra oppdragsgiver, og diskusjon innad i gruppen valgt å levere inkrementelt gjennom moduler nevnt tidligere ved hjelp av scrum som systemutviklingsmodell.

I Gantt-diagrammet vedlagt illustreres planen for gjennomføring av prosjektet. Ved prosjektstart lages lo-fi prototyper som skal presenteres til oppdragsgiver. Før første sprint arrangeres en workshop hvor gruppen organiserer det tekniske, med formål å forberede alle gruppemedlemmer så gruppen får et godt utgangspunkt. Dette innebærer oppsett av konfigurasjonsstyring, utviklertmiljø og lokale testservere på egne maskiner.

Etter dette starter selve utviklingsperioden hvor gruppen jobber i sprinter. Her arbeides det i perioder på én eller to uker for å bli ferdig med aktuell modul. Dette mønsteret gjentas frem til 30. Mars, da siste planlagte modul skal leveres. Siden siste modul leveres vil gruppen på dette punktet være ferdige med det som inngår i de funksjonelle kravene fra oppdragsgiver. Dette er en stor milepæl i utviklingen, og det vil i følge planen leveres en 1.0 versjon av programvaren. Grunnen til at dette er satt som en hard tidsfrist er for å bli kjent med oppdragsgivers preferanser, før gruppen legger til ekstra funksjonalitet. Dermed er gruppen bedre forberedt til å ta velbegrunnede valg av ny funksjonalitet som skal legges til løsningen. I den følgende perioden mot innlevering av endelig rapport er det planlagt bugfixing, videreutvikling og en endelig 2.0 versjon.

7. KILDER

[1] Gitek AS (18.12.2017). *Produktspesifikasjon Skogfrøverket*

[2] OWAP (13.07.2017). *Threat Risk Modeling*. Tilgjengelig i fra:
https://www.owasp.org/index.php/Threat_Risk_Modeling#DREAD (Hentet: 21.01.18)

[3] Norman D. A. (2013) *The Design of Everyday Things*. USA: Basic Books.

[4] Sommerville, I. (2014) *Software Engineering*. Ninth Edition

[5] Driessen V. (2010) *A successful git branching model*.
<http://nvie.com/posts/a-successful-git-branching-model/> (Hentet: 24.01.2018)