Anders Graval Steen
Stian Ådnanes

Visualisering av registrerte sauer på utmarksbeite

Masteroppgave i Datateknologi
Veileder: Svein-Olaf Hvasshovd
Juni 2022

Masteroppgave

NTNU
Norges teknisk-naturvitenskapelige universitet
Fakultet for informasjonsteknologi og elektroteknikk
Institutt for datateknologi og informatikk

NTNU
Norwegian University of
Science and Technology

Anders Graval Steen
Stian Ådnanes

Visualisering av registrerte sauer på utmarksbeite

Masteroppgave i Datateknologi
Veileder: Svein-Olaf Hvasshovd
Juni 2022

Norges teknisk-naturvitenskapelige universitet
Fakultet for informasjonsteknologi og elektroteknikk
Institutt for datateknologi og informatikk

NTNU
Kunnskap for en bedre verden

Sammendrag

I 2021 ble det sluppet ut 1,4 millioner sauer og lam på utmarksbeite. I perioden dyrene er på utmarksbeite, må bonden gå minst en tilsynstur i uken for å sjekke velferd, status og eventuelt tap av sauer. Ved en tilsynstur kan bonden se hvordan sauene beveger seg, og registrere hvor mange sauer som ble sett. På tilsynsturen blir sauene registrert med penn og papir, eller memorert av bonden. Det var derfor et ønske om å registrere sauer på en mobil enhet som ikke var avhengig av internett eller dekning. Et av de viktigste kravene var at registrering av sauer kunne utføres uten å se på mobilskjermen. Etter fullført tilsynstur, var det ønskelig å kunne se og analysere dataene som hadde blitt registrert, på en webapplikasjon.

Prosjektet er delt inn i to deler, forprosjekt og masteroppgave. Forprosjektet startet med et litteratursøk, for å finne ut hvilke digitale løsninger som eksisterer i forbindelse med å registrere sauer på utmarksbeite. Resultatet var en mobilapplikasjon der en kunne registrere fargene til bjelleslipsene til sauene uten å se på skjermen til mobilapplikasjonen. Videre kunne applikasjonen registrere hvor sauer ble observert, hvor mange som ble observert, hvilken farge sauene hadde og hvilke type sau det var (søye eller lam). Registreringene var ikke avhengige av internett, og kunne lastes opp i en database når brukeren fikk internettilgang.

Målet med masteroppgaven var å hjelpe bønder med å visualisere dataene som ble registrert på tilsynsturer. Med utgangspunkt i tidligere arbeid på feltet, eksisterende løsninger, krav fra myndighetene og erfaringer fra veileder Hvasshovd, ble det utviklet en webapplikasjon, *BeiteSpor*. Webapplikasjonen hentet data som ble lastet opp i en skydatabase, fra tidligere tilsynsturer som ble registrert i mobilapplikasjonen. I webapplikasjonen ble Norgeskart brukt som kartgrunnlag, med mulighet for å vise alle turene og posisjon til saueflokkene som ble registrert. Kartet kan også vise vekstpotensialet ved bruk av et bonitetslag. For å hjelpe bøndene med å se hvor rovdyr ble observert, ble posisjonene til alle observerte rovdyr i Norge hentet fra Rovbase. For å teste løsningen innenfor målgruppen, ble det gjennomført brukertester på fem bønder.

Fra brukertestene kommer det frem at en webapplikasjon med innsamlet data fra tilsynsturene vil hjelpe bøndene med å få bedre oversikt over sauene. Videre kan webapplikasjonen hjelpe med å oppdage mønster, og se sammenhenger mellom ulike turer. Ved bruk av innlogging sikrer en at hver bruker kun kan se sine egne turer. Ved videre utvikling kunne en lagt inn beitelag, slik at personer i samme beitelag kan se hverandres turer. Sammenlignet med bruk av tilsynsbok eller notering på papir, vil *BeiteSpor* forenkle og tydeliggjøre

innsamlet data fra tilsynsturene og øke forståelsen av bevegelsesmønsteret.

Abstract

In 2021, 1,4 million sheep and lambs were released on open pasture. When the sheep are on open pasture, the farmer needs to supervise at least once every week to check on the welfare, status, and possible loss of sheep. During supervision, the farmer will check on the movement of the sheep and register the number of sheep. During supervision, the sheep are usually registered either with pen and paper, or memorized by the farmer. Therefore, it was desirable to create a mobile application that worked without internet access. One of the most important requirements was that the registration of sheep could be performed without looking at the mobile screen. After completing the supervision, it was desirable to be able to view and analyze the data that had been registered on a web application.

The project is divided into two parts, a research project and a master's thesis. The research project started with a literature review to find existing digital solutions related to registering sheep on free range pastures. Based on the literature review, a mobile application was developed where the user can register the bell ties' colors without looking at the screen of the mobile application. The other registration options were the location of the flock of sheep, number of sheep, color of the sheep, and what type of sheep (sheep or lamb). The registrations were not dependent on an internet connection, and could be uploaded to a database when the user gained internet access.

The goal of the thesis was to help farmers visualize data from supervision trips. The web application, *BeiteSpor*, was based on previous work, existing solutions, requirements from the authorities, and knowledge from supervisor Hvasshovd. Data registered in the mobile application from previous supervision trips, was uploaded to a cloud database, and then fetched by the web application. Norgeskart was used as map basis in the web application with the possibility of showing all the registered trips and the position of the flock of sheep. Additionally, a map that shows grass growth potential can be placed over the map. To help the farmers keep track of predators, all the positions of observed predators in Norway were fetched from Rovbase. Five user tests where conducted on farmers to see how the web application fulfills the needs of the users.

These user tests show that a web application with collected data from the pasture inspections will help the farmers get a better overview of their own sheep. Furthermore, the web application can help to detect patterns, and see connections between different inspections. Since every user needs to be authorized, the application ensures that each user can only see their own inspections. Possibility for cooperation between farmers in the application can be considered in further development, so that farmers inside the cooperation group

can see each others pasture inspections. Compared to the use of an inspection book or note on paper, *BeiteSpor* will simplify and clarify the collected data from inspections and increase the understanding of the sheep's movement patterns.

Forord

Denne masteroppgaven har blitt gjennomført av Anders Gravdal Steen og Stian Ådnanes ved studiet Datateknologi, med spesialisering innen programvaresystem ved IDI, NTNU.

Vi ønsker å takke vår veilder Svein-Olaf Hvasshovd for ukentlige møter, og tilbakemeldinger og innspill underveis i prosjektet. Videre ønsker vi å takke bønder som stilte opp til brukertesting av webapplikasjonen. Videre i teksten vil også vi bli referert til som utviklerne.

Innhold

Figurer **x**

Tabeller **xii**

Akronymer **xiii**

I Introduksjon **1**

1 Motivasjon **2**

2 Problemstilling **3**
2.1 Mål for oppgaven . . . 3
2.2 Problemstilling . . . 3

3 Rapportoversikt **4**

II Bakgrunn **5**

4 Dagens situasjon **6**
4.1 Utmarksbeite . . . 6
4.2 Skade fra rovdyr . . . 8
4.3 Sauekontrollen . . . 8
4.4 Rapportering av tilsynsturer . . . 8
4.5 Oppsummering . . . 9

5 Forprosjekt **10**
5.1 Bruk av mobilapplikasjon . . . 10
5.2 Registrering av sauer . . . 10
5.3 Datainnsamling fra turer . . . 11
5.4 Endring av løsning . . . 11
5.5 Oppsummering . . . 11

6 Tidligere arbeid **12**
6.1 Telespor . . . 12
6.2 FindMy . . . 13
6.3 Smartbjella . . . 13

6.4 BeiteSnap . . . 14
6.5 ZooMonitor . . . 15
6.5.1 Oppsummering . . . 16

III Eget bidrag 17

7 Valg av prosess 18
7.1 Oppsummering . . . 20

8 Programflyt og kravspesifikasjon 21
8.1 Use Case . . . 21
8.2 Kravspesifikasjon . . . 22
8.2.1 Funksjonelle krav . . . 23
8.2.2 Ikke-funksjonelle krav . . . 25
8.3 Oppsummering . . . 26

9 Valg av teknologi 27
9.1 Valg av teknologi til brukergrensesnittet . . . 27
9.1.1 Rammeverk til brukergrensesnittet . . . 27
9.1.2 Kartløsning . . . 28
9.2 Valg av teknologi til serveren . . . 28
9.2.1 Programvarearkitektur . . . 28
9.2.2 Rammeverk til serveren . . . 29
9.2.3 Database- og publiseringsløsning . . . 30
9.3 Oppsummering . . . 30

10 Design 31
10.1 Brukeropplevelse . . . 31
10.2 Fargevalg . . . 32
10.3 Prototype . . . 34
10.3.1 Resultater . . . 35
10.4 Oppsummering . . . 39

11 Implementasjon 40
11.1 Database . . . 40
11.1.1 Databasestruktur . . . 40
11.1.2 Azure database . . . 42
11.1.3 Entity Framework Core . . . 42
11.2 ASP.NET Core Identity . . . 43
11.3 Struktur . . . 44
11.3.1 Brukergrensesnitt . . . 44
11.3.2 Server . . . 45
11.4 Norgeskart . . . 46
11.5 Bonitet . . . 47
11.6 Identifisering av saueflokker . . . 49

11.7 Årsrapport 50
11.8 Rovdyrskade 50
11.8.1 Rovbase 50
11.8.2 Registrert hendelse fra mobilapplikasjon 53
11.9 Saueflokk 53
11.10Oppsummering 56

IV Utførelse og resultat 57

12 Brukertest 58
12.1 Kvalitativ analyse 58
12.2 Gjennomføring av brukertester 59
12.2.1 Før gjennomføring av brukertester 59
12.2.2 Under brukertesten 60
12.2.3 Etter brukertesten 61
12.2.4 Digital gjennomføring 62
12.2.5 Pilottest 62
12.2.6 Oppsummering 62

13 Resultat brukertest 63
13.1 Bakgrunnsinformasjon 63
13.2 Oppgave 1 (Velg periode og rovdyr) 63
13.2.1 Brukertest observasjoner 63
13.2.2 Spørsmålsbesvarelse 64
13.3 Oppgave 2 (Oversikt over tilsynsturer) 64
13.3.1 Brukertest observasjon 64
13.3.2 Spørsmålsbesvarelse 64
13.4 Oppgave 3 (Måneds- og ukesvisning med heatmap og saueflokk) 65
13.4.1 Brukertest observasjon 65
13.4.2 Spørsmålsbesvarelse 65
13.5 Oppgave 4 (Bonitet og PDF) 66
13.5.1 Brukertest observasjon 66
13.5.2 Spørsmålsbesvarelse 66
13.6 Spørsmål etter gjennomført brukertest 67
13.6.1 Hva bruker du nå for å holde kontroll på sauene dine? 67
13.6.2 Har du registrert sauer manuelt tidligere? 67
13.6.3 Tror du denne løsningen som du har prøvd ut nå kan hjelpe deg til å bedre forstå hvordan saueflokkene beveger seg? 67
13.6.4 Var det noe som manglet for å kunne se bevegelsene til sauene? . . . 67
13.6.5 Kan løsningen gjøre jobben din som sauebonde enklere? 68
13.7 Oppsummering 68

V Diskusjon **69**

14 Resultat brukertest **70**
14.1 Turoversikt i applikasjonen . . . 70
14.2 Navigasjon mellom oppsynsturer . . . 70
14.3 Informasjon om en saueflokk . . . 71
14.4 Visning av saueflokker . . . 71
14.5 Viktighet av rovbase . . . 71
14.6 Genering av PDF-rapport . . . 72
14.7 Oppsummering . . . 72

15 Evaluering av prosess **73**
15.1 Design . . . 73
15.2 Utvikling . . . 73
15.3 Gjennomføring av brukertest . . . 73
15.4 Oppsummering . . . 74

16 Evaluering av løsningen **75**
16.1 Server . . . 75
16.2 Brukergrensesnitt . . . 75

17 Kravspesifikasjon **76**
17.1 Funksjonelle krav . . . 76
17.1.1 FK7 - Glemt passord . . . 76
17.1.2 FK13 - Vise bondens tilsynsrute på kart . . . 76
17.1.3 FK9 - Generere en PDF-årsrapport . . . 76
17.2 Ikke-funksjonelle krav . . . 77
17.2.1 Sikkerhet . . . 77
17.2.2 Vedlikehold . . . 77
17.2.3 Brukbarhet . . . 77
17.2.4 Ytelse . . . 77
17.3 Oppsummering . . . 77

18 Sammenligning med eksisterende løsninger **78**

19 Prosjektevaluering **79**
19.1 FS1.1 - Kan et digitalt system fremstille innsamlet data fra tilsynsturer på en mer oversiktlig og effektiv måte sammenlignet med penn og papir? . . . 79
19.2 FS1.2 - Hvilke faktorer bidrar til en bedre forståelse av bevegelsesmønsteret, og øker effektivitet i videre tilsynsturer? . . . 79
19.3 FS1.3 - Kan et system integrert med rovdata hindre eller redusere tap av søye og lam til rovdyr? . . . 80

VI Konklusjon og videre arbeid **81**

20 Konklusjon **82**

21 Videre arbeid **83**
21.1 Turvisning . . . 83
21.2 Glemt passord . . . 83
21.3 Integrering av sporingsenhet . . . 84
21.4 Sauer på kart . . . 84
21.5 Integrasjon mot Sauekontrollen . . . 84
21.6 Implementere beitelag . . . 84
21.7 Legge inn beiteområde . . . 85
21.8 Testing . . . 85

Referanser **85**

A Brukertester **92**
A.1 Brukertest A . . . 92
A.2 Brukertest B . . . 97
A.3 Brukertest C . . . 102
A.4 Brukertest D . . . 107
A.5 Brukertest E . . . 111

B Bruksanvisning for å kjøre koden **116**
B.1 Laste ned kildekoden . . . 116
B.2 Kjøre koden . . . 116
B.2.1 Backend . . . 116
B.2.2 Frontend . . . 116

Figurer

4.1 Sau på utmarksbeite [5]. 7
4.2 Sanking av sauer [5]. 7
4.3 Antall sauer og lam erstattet som tapt til rovvilt i 2021 fordelt på rovdyr til venstre og fordelt på fylke til høyre [6]. 8
4.4 Rapporteringsflyt av tilsynsturer. 9

6.1 Sporingsenheten til Telespor [13] 13
6.2 Skjermbilder fra funksjoner i FindMy kartapplikasjon [18] 13
6.3 Sporingsenheten til Smartbjella [19] 14
6.4 Skjermbilde av registrering av skadet sau på BeiteSnap [24] 14
6.5 Oversikt over struktur til ZooMonitor applikasjon [25] 15
6.6 Grafer generert av ZooMonitor programvare [25] 16
6.7 Grafer generert av ZooMonitor programvare etter installasjon av tre matstasjoner [25] 16

7.1 Vannfallsmetoden [26] og smidig metodikk [37] 19
7.2 Trello ble brukt til å lage Kanban Board 20

8.1 Use Case diagram for BeiteSpor 22

10.1 Oversikt over hvordan farges sees med rødgrønn fargeblindhet [69] 33
10.2 Fargepalett for webapplikasjon 33
10.3 Prototype av registrering og innlogging av bruker 35
10.4 Prototype for meny 36
10.5 Årsrapport over sauene 37
10.6 Prototype av en valgt tur 37
10.7 Prototype av en tur på applikasjonen 38
10.8 Prototype av heatmap på kartet 39

11.1 ER-diagram for databasestruktur 41
11.2 ER-diagram for lagring av brukerinformasjon i database 43
11.3 Autentisering på API-endepunkt 44
11.4 Overordnet struktur til brukergrensesnittet 44
11.5 Kart over endepunktet på serveren 45
11.6 Fremstilling av rutekoordinat i Norgeskart 47
11.7 Kart med Google Maps og Norgeskart 47

11.8 Oversikt over bonitetsklasser [81] 47
11.9 Kart med Noregskart og bonitetskart 49
11.10 PDF-årsrapport over sau og rovdyr 50
11.11 Skjermbilde med sammenligning av Rovbase sin nettløsning og integrasjon med Rovbase i våres løsning 51
11.12 Skjermbilde av filtrering av rovdyrhendelser i webapplikasjonen 52
11.13 Ikoner brukt for markører i webapplikasjonen, som er hentet fra Rovbase . 52
11.14 Markør for død sau på kartet 53
11.15 Oversikt over saueflokker på beite med ukes- og månedsvisning 54
11.16 Informasjon om en bestemt saueflokk 54
11.17 Heatmap med uke- og månedsvisning for saueflokker i *BeiteSpor* 55
11.18 Oversikt over alle turene tilgjengelig innenfor en valgt tidsperiode 55

12.1 Myers og Avison [88] epistemologier for kvalitativ analyse 58

Tabeller

8.1 Funksjonelle krav for server 23
8.2 Funksjonelle krav for brukergrensesnittet 24

Akronymer

AAS Azure App Service. 30, 74, 75, 77
ANC ASP .NET Core. 29, 30, 75, 77
API Application Programming Interface. 11, 22, 28, 47, 51, 75, 77

CORS Cross-Origin Resource Sharing. 51
CSS Cascading Style Sheets. 50
CSV Comma-Separated Values. 15

EFC Entity Framework Core. 29, 30, 42
EFM Entity Framework Migrations. 29

GNSS Global Navigation Satellite Systems. 12
GPS Global Positioning System. 14

HTML HyperText Markup Language. 50
HTTP Hypertext Transfer Protocol. 45, 48

IDE Integrated Development Environment. 27, 30
IoT Internet og Things. 12, 14

JWT JSON Web Tokens. 45

LTE-M Long Term Evolution Machine Type Communication. 12

NIBIO Norsk institutt for bioøkonomi. 21, 47

OBB Tilskudd til drift av beitelag. 8
ORM Object-Relational Mapper. 29

REST Representational State Transfer. 28, 75

URL Uniform Resource Identifier. 46

WIP Work In Progress. 19, 20
WMS Web Map Service. 48

Del I

Introduksjon

Introduksjonen til masteroppgaven inneholder motivasjonen og målet for oppgaven, som vider blir definert i en problemstilling som legger grunnlaget for utviklingen av *BeiteSpor*. Oversikten over strukturen til rapporten vil også bli gjennomgått her.

Kapittel 1

Motivasjon

Utmarksbeite er en stor del av sauedriften i Norge. I 2021 ble det sluppet ut om lag 1,4 millioner sauer og lam på utmarksbeite [1]. Dyrene slippes ut på vårparten og oppholder seg på beite over sommeren, før de sankes inn om høsten. På utmarksbeite er det flere farer som lurer. Dersom sauene møter på rovdyr, blir avmagret eller blir syke, kan dette føre til at ikke alle overlever beitesesongen. I 2021 var den samla tapsprosenten på sau og lam i utmark på 5,2 % [1]. Det er bondens ansvar å redusere risikoen for at sauer og lam går tapt på utmarksbeite. Deteksjon av tegn på unormale omstendigheter gjøres gjennom jevnlige tilsynsturer i beiteområde. Det er et krav om at tilsynsturer skal gjennomføres minst en gang i uken. Myndighetene har også satt inn tilskudd for bønder som slipper sauer og lam på utmarksbeite. Som krav for tilskuddet må det fremstilles dokumentasjon fra tilsynsturene. I tillegg til kravene som stilles av myndighetene, går bonden tilsynsturene for å holde oversikt over sine egne dyr.

Frem til i dag har dokumentasjon fra tilsynsturene blitt gjennomført med penn og papir. Dokumentasjonen må ikke skrives i et bestemt format, og vil derfor variere fra bonde til bonde. Informasjon som kan bli registert fra tilsynsturene er lokasjon, antall dyr, og om noen dyr er syke eller døde. Egne notater over observasjoner fra tilsynsturer på papir er det eneste bonden sitter igjen med etter fullført tur. Det er vanskelig å få et helhetsbilde med notater på papir, derfor er det et potensiale for å kunne gjøre jobben til bonden enklere. Digitalisering av registreringene og visualisering av dataene i en applikasjon, kan gi bonden en bedre oversikt over egne sauer.

Veileder hvasshovd presenterte et oppgaveforslag om å lage et system for samling og visualisering av innsamlet data fra tilsynsturer. Oppgaven er basert på dagens situasjon, der registrering foregår med penn og papir, og bonden har lite system på tidligere turer som har blitt skrevet ned. I løpet av høsten 2021 ble det utviklet en mobilapplikasjon for registrering av observasjoner fra tilsynsturer, som gir datagrunnlaget til webapplikasjonen som har blitt utviklet våren 2022. Masteroppgaven presenterer planleggingen, utviklingen og testingen av webapplikasjonen. Hensikten med oppgaven er å hjelpe bonden til å bedre forstå hvordan sauer beveger seg med bruk av innsamlet data fra tilsynsturer.

Kapittel 2

Problemstilling

2.1 Mål for oppgaven

Høsten 2021 ble det utviklet en mobilapplikasjon som forprosjekt til masteroppgaven [2]. Bonden kunne bruke mobilapplikasjonen på tilsynsturer, og registrere blant annet posisjon og hvor mange sauer som ble observert. Dataene ble lastet opp i en database når bonden kom hjem, ettersom det er mangel på dekning og internett ute på beiteområdet. Per dags dato er det ingen måte å se dataene som ble registrert på tilsynsturene. Målet med masteroppgaven er å utvikle en webapplikasjon som kan vise dataene som ble registrert ved hjelp av mobilapplikasjonen.

2.2 Problemstilling

Basert på mobilapplikasjonen som ble laget i forprosjektet, og hvordan bonden gjennomfører en tilsynstur per dags dato, er det laget en problemstilling med tre forskningsspørsmål som skal besvares i løpet av masteroppgaven.

- **PB1:** Hvordan fremstille innsamlet data fra tilsynsturer for å bedre forståelsen av bevegelsesmønsteret til sauer?
- **FS1:** Kan et digitalt system fremstille innsamlet data fra tilsynsturer på en mer oversiktlig og effektiv måte sammenlignet med penn og papir?
- **FS2:** Hvilke faktorer bidrar til en bedre forståelse av bevegelsesmønsteret, og øker effektivteten i videre tilsynsturer?
- **FS3:** Kan et system integrert med rovdata hindre eller redusere tap av søye og lam til rovdyr?

Kapittel 3

Rapportoversikt

Resten av masteroppgaven er delt inn i fem deler:

- **Del II: Bakgrunn** tar for seg dagens situasjon, forprosjektet som ble gjennomført høsten 2021 og eksisterende løsninger innenfor feltet.
- **Del III: Eget Bidrag** beskriver utviklingsprosessen og begrunnelse for teknologivalg. Videre kan en se kravspesifikasjon, designvalg og hvordan løsning ble implementert.
- **Del IV: Utførelse og resultat** går gjennom hvordan brukertestene ble utført, og hvilke resultater som kom ut av testene.
- **Del V: Diskusjon** tar for seg drøfting av resultatene fra brukertestene og svarer på forskningsspørsmålene, evaluering av prosess og løsningen, og sammenligner eksisterende løsninger.
- **Del VI: Konklusjon og videre arbeid** oppsumerer arbeidet som har blitt gjort, svarer på problemstillingen, samt kommer med forslag til videre arbeid til webapplikasjonen.

Del II

Bakgrunn

I denne delen blir det presentert bakgrunn for masteroppgaven. Først vil dagens situasjon bli gjennomgått med hvordan sauer på utmarksbeite håndteres i dag, hvilke rapporteringer som må gjøres til myndighetene og hvilke trusler sauer står overfor på utmarksbeite. Dette etterfølges av gjennomgang om hva som har blitt gjort i forprosjektet høsten 2021, og til slutt tidligere arbeid på feltet.

Kapittel 4

Dagens situasjon

På vårparten slippes sauer ut på beite. I løpet av beiteperioden har bonden ansvar for tilsyn av sine egne beitedyr, som utføres gjennom jevnlige tilsynsturer. På en tilsynstur oppsøker bonden sauene i teigen og sjekker generell tilstand hos sauene. Ved observasjon av skadet eller død sau, må bonden iverksette tiltak for å ivareta sauenes velferd. Frem til i dag har registrering og loggføring av observasjoner fra tilsynsturer blitt gjennomført med penn og papir.

4.1 Utmarksbeite

Ved drift av sauer i Norge er det vanlig å ha sauer på utmarksbeite om sommeren. Tall fra produksjons-tilskuddsregister hos Landbruksdirektoratet viser at om lag 74 % av alle sauer og lam ble sluppet ut på utmarksbeite i 2021 [1]. Bøndene kan få tilskudd for å la sauene gå på beite i utmark. *Forskrift om tilskott til organisert beitebruk* fra Landbruks- og matdepartementet inneholder betingelser for å få tilskudd på utmarksbeite [3]. I §5 står det at et vilkår for å få tilskudd til organisert beitebruk er at husdyreierne er organisert i beitelag. Et godkjent beitelag må bestå av minst 5 medlemmer og 300 småfe hvis det er nyregistrert eller minst 3 medlemmer og 150 småfe for eksisterende lag.

I henhold til informasjon fra domeneekspert og veileder, Hvasshovd, slippes sauene ut i mai og sankes inn i løpet av september. Perioden sauene er ute på utmarksbeite kan variere fra sted til sted med tanke på klima, trussel for rovdyr og kvalitet på beiteområde. Utsatt slipp av dyrene kan forekomme dersom det er en unormal kald vår. Etter dyrene har blitt sluppet ut på beitet starter prosessen for å sørge for at dyrene i utmark har det bra. *Forskrift om velferd for småfe* som er laget av Landbruks- og matdepartementet skal legge til rette for god helse og trivsel hos dyrene [4]. §19 i forskriften omhandler tilsyn og stell av småfe. For dyr som holdes på utmarksbeite, skal det gjennomføres et tilsyn en gang per uke i områder uten særskilt risiko [4]. Hvis det er mistanke om økt fare, skal tilsynet intensiveres. Dette kan være avmagring som følge av dårlig beiteforhold eller skadde dyr fra rovdyr.

Figur 4.1: Sau på utmarksbeite [5].

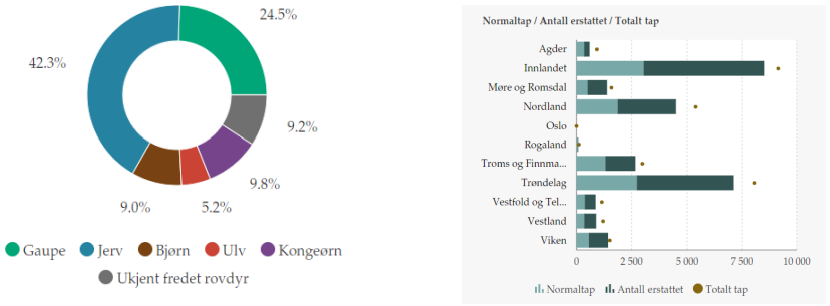
En tilsynstur gjennomføres enten av en enkelt bonde eller gjennom et beitelag minst en gang per uke. En tilsynstur går ut på at den ansvarlige for tilsynsturene oppsøker dyrene i beiteområdet. Posisjonen til dyrene kan enten finnes gjennom digitale sporingsverktøy eller at den ansvarlige tilsynspersonen har kjennskap til områdene dyrene beveger seg i. Etter dyrene er lokalisert skal tilsynspersonen se etter det som er utenfor det normale, slik som skadde, døde eller avmagret dyr. Siden beiteområdene ofte er store og sauene er posisjonert spredt, er det ikke noe krav at alle sauene blir telt og registrert på tilsynsturen. Fra §19 i *Forskrift om velferd for småfe* skal da tilsynsturer intensiveres til flere ganger i uken. Den vanligste måten å loggføre hendelser fra en tilsynstur er ved bruk av penn og papir. Derfor er det potensiale for å kunne gjøre det enklere ved å loggføre hendelser med en digital løsning der dataene fra tilsynsturene kan samles i en felles database, slik at de ansvarlige for tilsynsturene enklere kan sette inn tiltak når det trengs.

Figur 4.2: Sanking av sauer [5].

Mot slutten av utmarksbeitesesongen starter arbeidet med å få dyrene tilbake fra beite. Denne prosessen starter på forskjellige tidspunkt, basert på hvor en er i landet og forholdet til det aktuelle beiteområde. *Forskrift om velferd for småfe* inneholder paragrafer om hvilke hensyn som må tas når sanking av sauer skal gjøres [4]. §27. *Hjemsanking av sau* sier at dyreeierne må sørge for å få dyrene tilbake fra utmarksbeite i god tid før frost og snø kommer på høsten. Klima varierer både i landet og fra år til år, og bonden må ta hensyn til hvordan klima utvikler seg i beiteområde. I tillegg til klima må dyreierne ta hensyn til rovdyr i beiteområdene. I §26 om forebyggende tiltak, står det at dyr som holdes på utmarksbeite skal beskyttes mot rovdyr. Høy aktivitet av rovdyr i beiteområdene kan gjøre at dyreeierne må ta en tidligere innsanking av dyrene enn planlagt.

4.2 Skade fra rovdyr

I 2021 var det 16864 sauer som ble erstattet etter tap til rovdyr [6]. Det har vært en nedgang i tilfeller av sauetap til rovdyr, men det er forsatt et problem sauebønder i landet trenger å forholde seg til. Som det nevnes i kapittel 4.1 har også bøndene en plikt for å skåne sauene mot rovdyrangrep gjennom §26 i *Forskrift om velferd for småfe* [4].



Figur 4.3: Antall sauer og lam erstattet som tapt til rovvilt i 2021 fordelt på rovdyr til venstre og fordelt på fylke til høyre [6].

Hendelser av skadde og døde sauer fra rovdyr varierer i landet. Mye av dette skyldes forekomster av de ulike rovdyrene rundt i landet. Figur 4.3 viser fordeling av sauetap fordelt på rovdyr og fylker i 2021 [6]. Forskjellen i tap av sauer til rovdyr varierer stort i Norge, som har en sammenheng med forekomsten av rovdyr i de forskjellige landsdelene. Jervbestanden er størst i fylkene Innlandet og Trøndelag. I 2021 var det registrert 18 valpekull i Hedmark, samt 17 valpekull i Trøndelag, og Møre og Romsdal [7]. Disse områdene hadde opp mot dobbelt så mange valpekull sammenlignet med andre områder i landet. 42,3% av alle erstattede sauer og lam var forårsaket av jerv, slik en kan se i figur 4.3. Dette har en sammenheng med at klart flest sauer går tapt i fylkene Innlandet og Trøndelag.

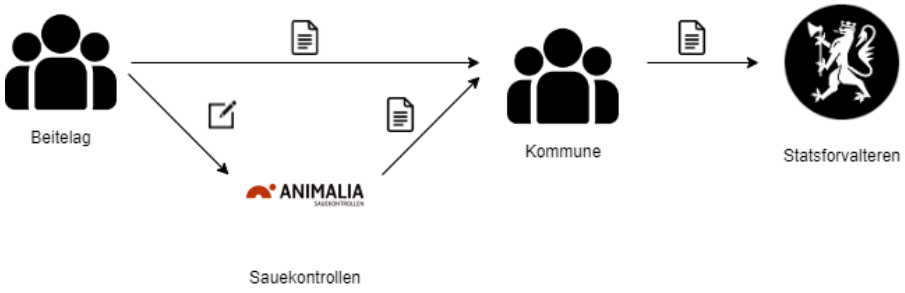
4.3 Sauekontrollen

Sauekontrollen er en landsomfattende husdyrkontroll for sauer, som tilbys av Animalia AS [8]. Det er over 5 400 medlemmer av sauekontrollen i dag, som er en stor andel av de som driver med sauedrift i dag [8]. Tall fra statistisk sentralbyrå i 2021 viser at det er totalt 13 568 sauebønder, som vil si at rundt 40 % av alle sauebønder bruker sauekontrollen [9]. Med sauekontrollen registrerer medlemene lamming, høstvekter på lam og utmelding av dyr som dør utenom slakt [8]. Fra registreringene som blir gjort kan det blant annet bli generert rapporter for slakteresultater eller årsresultater.

4.4 Rapportering av tilsynsturer

Alle beitelag har mulighet for å søke om tilskudd til organisert beitebruk gjennom *Tilskudd til drift av beitelag (OBB)* fra Landbruksdirektoratet [10]. Ordningen er til for å få en bedre utnyttelse av utmarksbeite og redusere tap av dyr, der beitelaget får en pris per sau som blir sluppet ut på utmarksbeite. For å få tilskuddet må alle medlemmer ha et foretak i Enhetsregisteret, og sende inn søknadsskjemaet til Landbruksdirektoratet en gang i året.

I søknadsskjemaet skal det oppgis en rapport for tilsynsturene som har blitt gjennomført i løpet av året.



Figur 4.4: Rapporteringsflyt av tilsynsturer.

Figur 4.4 viser hovedtrekkene i flyten for rapportering av tilsynsturer. §6 i *Forskrift om tilskott til organisert beitebruk* sier at beitelagene skal dokumentere alle tilsynsturene som blir gjennomført [3]. Det er ikke krav til hvilket format dokumentasjon skal bli ført i, men den skal inneholde dato, tilsynsrunde, unormale forhold i beiteområdet og tiltak som er iverksatt. Den vanligste måten å føre dokumentasjonen på er å fylle ut et skjema med penn og papir. Kommune og statsforvalter kan kreve dokumentasjon, som fremlegges i §6. Som nevnt i kapittel 4.3 er det en stor andel av sauebøndene som er medlem av Sauekontrollen. Applikasjonen til Sauekontrollen tilbyr registrering av tilsynsturer, som gjør at bøndene kan bruke tilsynsdokumentasjonen fra Sauekontrollen overfor kommunen og statsforvalteren, representert i figur 4.4 [11].

4.5 Oppsummering

Bruk av utmarksbeite er en stor del av driften for sauebønder i Norge. Som en konsekvens av å ha sauer på utmarksbeite, plikter bonden seg til å følge opp sauene i beiteperioden. Fra forskrifter skal bonden gjennomføre minst en tilsynstur en gang i uken, der dokumentasjon kan kreves til fremlegging overfor myndighetene. Motivasjonen for å gjennomføre tilsynsturene er å sørge for at egne sauer har det bra på beite. I tillegg til krav fra myndighetene, er det en fordel å holde oversikt gjennom beiteperioden, som kan hindre tap av sauer og gjøre en innsanking av sauene enklere. Siden tilsynsturene kan være ressurskrevende å gjennomføre, og mange bønder deler beiteområder, er beitelag en vanlig måte å organisere seg på. Her fordeles arbeidet blant medlemmene i beitelaget. Tilsynsturer gjennomføres ved at tilsynsansvarlig lokaliserer sauene, hvor sporingsverktøy kan benyttes, og registrerer relevant informasjon om saueflokkene.

Kapittel 5

Forprosjekt

Høsten 2021 ble det jobbet med mobilapplikasjonen som la grunnlaget for data som blir brukt i masteroppgaven. Mobilapplikasjonen blir brukt som et verktøy på tilsynssturer av sauer på beitet. For hver inspeksjon starter bonden en tur i applikasjonen, som logger posisjonen til bonden under turen.

5.1 Bruk av mobilapplikasjon

Mobilapplikasjonen er et tilleggsverktøy for brukeren når en skal gjennomføre en tilsynstur av sauer. Når tilsynsturen starter oppretter brukeren en ny tur i applikasjonen. Et kart med kartlag fra Norgeskart viser område brukeren navigerer seg i, sammen med en indikator av mobilens nåværende posisjon. En bakgrunnsprosess logger periodisk mobilens posisjon. Loggingen foregår fra brukeren oppretter og starter en tur, til den blir avsluttet. Alle posisjonene blir lagret i en lokal database på mobilen som kan hentes opp etter at turen er gjennomført.

5.2 Registrering av sauer

Den primære hensikten til mobilapplikasjonen er å registrere sauer som blir observert på tilsynsturen. Når en saueflokk blir observert, kan brukeren holde inne posisjonen til saueflokken på kartet, og det vil starte en ny registrering av sauer i applikasjonen. Registreringsmodulen til applikasjonen har mulighet for å legge inn antall sauer med fargene hvit, brun og svart, samt antall lam og søyer blant de tre fargene. I tillegg til antall typer sauer, kan det registreres farger på øremerker og slipsfarger. Fargen på øremerker indikerer eieren av dyret, mens fargen på slipset forteller hvor mange lam søya har [12]. De forskjellige fargene er rød (0 lam), blå (1 lam), gul (2 lam) og grønn (3 lam). Sammen med de nevnte karakteristikkene til saueflokkene, registreres både brukerens observeringsposisjon og posisjonene til saueflokkene. Ved endt registrering lagres flokken i en lokal database på mobilen.

5.3 Datainnsamling fra turer

Etter en gjennomført tilsynstur kan brukeren stoppe turen i mobilapplikasjonen og alle de innsamlede dataene fra den gjennomførte turen vil være lagret i den lokale databasen til mobilen. Applikasjonen har en lokal lagringsmulighet, ettersom internettdekning i områdene til turene kan være manglende. Løsningen baserer seg derfor på at brukeren selv laster opp dataene fra en tur til *BeiteSpor* etter en tur er blitt gjennomført. *BeiteSpor* har API med endepunkt for å ta imot dataene fra mobilapplikasjonen, og lagrer dette til en skydatabase.

5.4 Endring av løsning

I masteroppgaven har veileder og domeneekspert, Hvasshovd, lagt inn ønsker om å implementere tilleggsfunksjoner. I skytjenesten som ble laget i masteroppgaven er det lagt opp støtte for funksjoner som ikke er støttet i mobilapplikasjonen. Mobilapplikasjonen har ingen mulighet for å registrere døde sauer eller rovdyr som er observert, men dette er mulig å laste opp i skytjenesten. I tillegg har databasemodellen mulighet for å knytte en tur til en spesifikk bruker. I mobilapplikasjonen er ikke turen knyttet til en bruker.

5.5 Oppsummering

I forprosjektet ble det laget en mobilapplikasjon, der bonden kunne registrere sauer på utmarksbeite. Informasjonen ble lagret lokalt på mobilen, og sendt til en skytjeneste når mobilen fikk internettilgang. I en registrering var det mulig å legge til antall sauer med tilhørende farge, samt hvor mange søyer og lam det var i flokken. I tillegg kunne det bli registrert farge på øremerker og slipsfarger. Forbedringer som veileder Hvasshovd ønsket å implementere i masteroppgaven, var mulighet for å registrere sauer og døde dyr.

Kapittel 6

Tidligere arbeid

For å få en bedre innsikt i hva som har blitt gjort tidligere på området, og finne inspirasjon til masteroppgaven, har vi sett på tidligere arbeid som har blitt gjort. Innenfor visualisering av data fra manuell registrering av saueflokker på beite er det lite eksisterende arbeid. Det er tre bedrifter, Telespor, Smartbjella og FindMy som leverer elektroniske sporingsbjeller og digitale løsninger knyttet til sporingsenheten. Sammenlignet med forprosjektet og masteroppgaven har løsningene fra de tre bedriftene data med høyere oppløsning ved at posisjonsdata registreres oftere. Dette gjør at løsningene ikke er helt sammenlignbare, men de har flere funksjoner som kan være aktuelle inn mot masteroppgaven.

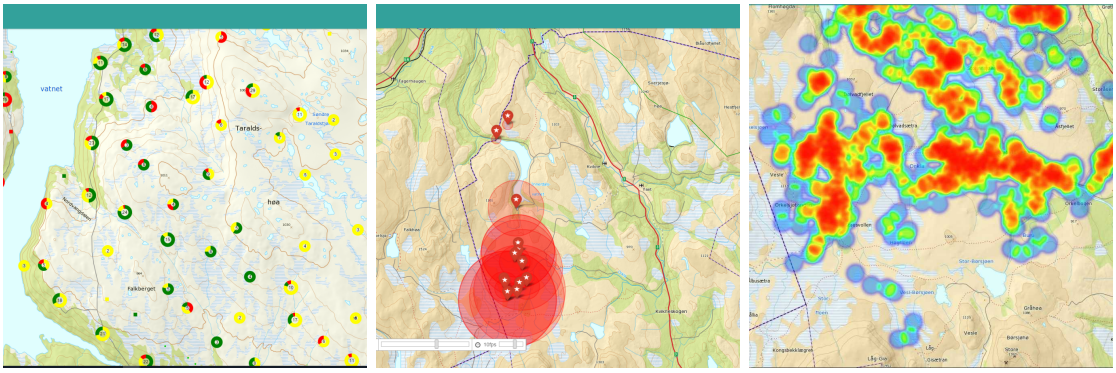
6.1 Telespor

Telespor er et norsk selskap som leverer sporingsløsninger av beitedyr [13]. Løsningene baserer seg på en radiobjelle som kan festes på sauer til en pris av 989 kr per bjelle [14]. Bjellene bruker GNSS satellitteknologi for å dele posisjonen til sauene [13]. Hyppigheten av posisjonsdelingen kan gå fra 5 minutter og oppover, der de har utviklet løsninger som baserer seg på data fra sporingsenhetene. I normale forhold kan sporingsløsningen rapportere omtrent 1000 ganger [15]. Dersom det blir sendt posisjon en gang i døgnet, vil batteritiden vare i nesten 3 år. I applikasjonen kan en se beitedyrposisjonene visualisert i et kart med kartlag fra Norgeskart, samt en liste over alle aktive radiobjeller. Telespor bruker et varslingssystem for hendelser utenfor det normale, som lavt batteri, manglene dekning fra radiobjella eller ingen bevegelse. Varslene kan enten komme gjennom SMS eller i mobilapplikasjonen. Radiobjella tilbyr toveis kommunikasjon ved bruk av LTE-M og IoT teknologi. Det betyr at brukeren kan endre alarmsensitiviteten, og hvor ofte bjella skal sende posisjon [13]. Fra figur 6.1 ser en et eksemplar av sporingsenheten til Telespor.

Figur 6.1: Sporingsenheten til Telespor [13]

6.2 FindMy

FindMy er også et norsk selskap som leverer sporingsløsninger av beitedyr [16]. Prisen per bjelle er 1980 kr, og batteritiden på halvårsbeitende dyr er 2 til 3 sesonger [17]. I likhet med Telespor har FindMy en elektronisk bjelle, E-bjelle, som løsningene deres bygges rundt. Det som skiller FindMy fra Telespor er et større spekter av funksjoner med beregninger og visualiseringer i deres kartapplikasjon, som også baserer seg på Norgeskart.



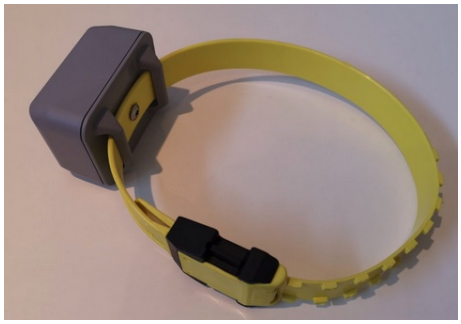
Figur 6.2: Skjermbilder fra funksjoner i FindMy kartapplikasjon [18]

Figur 6.2 viser eksempler på funksjoner FindMy har implementert i kartapplikasjon. Bildet til venstre i figuren viser tilvekstkart, som sier hvor stor tilvekst beiteområdene har og baserer seg på vår- og høstvekt fra Sauekontrollen [18]. Det midtre bilde er en visualisering av urovarsel som er implementert i applikasjonen, som oppdager bevegelser utenfor det normale mønsteret. Det siste bildet viser beitetrykket i områdene visualisert gjennom et heatmap, som er en visualiseringsmetode hvor områder vises med en skala fra rødt, med høy aktivitet, til blått med lav aktivitet.

6.3 Smartbjella

Smartbjella er i likhet med Telespor og FindMy produsert i Norge [19]. Bjella er laget for å spore sau og andre beitedyr. I forskjell fra de andre selskapene, startet Smartbjella originalt med å spore containere [20]. Ettersom gründeren hadde over 1000 sauer, ble sporingsenheten testet på sau, og dermed var de i dyrebransjen. Sammenlignet med både

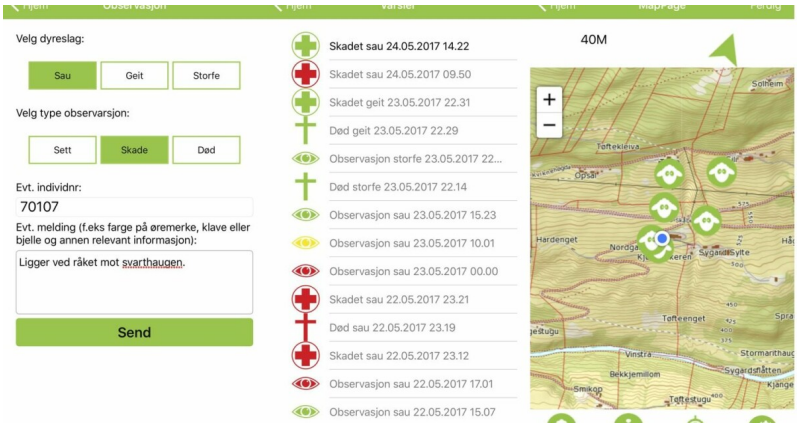
FindMy og Telespor, koster smartbjella 999 kr per bjelle, men den største forskjellen er batteritiden [21]. Dersom bjella skal rapportere en gang i døgnet, har bjella en levetid på 17 år. Smartbjella bruker mobilteknologien *Narrow band*-IoT, som bidrar til den lange batteritiden. Smartbjella tilbyr to ulike leverandører for mobildekning, enten Telenor eller Telia. Dekningen kan derfor være begrenset noen steder i Norge. I likhet med FindMy, har smartbjella en mobilapplikasjon med funksjonalitet som heatmap og GPS-visning på kart [22]. Fra figur 6.3 kan en se en sporingsenhet fra Smartbjella.



Figur 6.3: Sporingsenheten til Smartbjella [19]

6.4 BeiteSnap

BeiteSnap er laget for at bønder kan få varslinger fra turgåere i beiteområder [23]. Hver bonde må sette opp et forhåndsdefinert beiteområde på kartet, slik at det er mulig å få varsel dersom det blir registrerert et dyr innenfor beiteområdet. Turgåerne kan registrere observasjoner, skadde dyr, eller drepte dyr. De tar et bilde med telefonen, og skriver hvilken type dyr og observasjon det er. Posisjonen blir automatisk registrert basert på hvor de er. Dersom turgåerne ikke er innenfor et gyldig området, kan de velge hvilke beitemark de ønsker å registrere dyret på. Om høsten vil bøndene få en rapport med dokumentasjon av alt tilsyn som er blitt utført i løpet av sesongen. Bøndene som oppretter beiteområder, må betale en årspris på 1200 kr pluss moms for å kunne få et produsentnummer. Hver bonde kan opprette syv brukere på et produsentnummer. For turgåerne er det gratis å registrere observasjoner. Fra figur 6.4 kan en se skjermbilde fra mobilapplikasjonen til BeiteSnap, der det ble registrert en skadet sau på kartet.

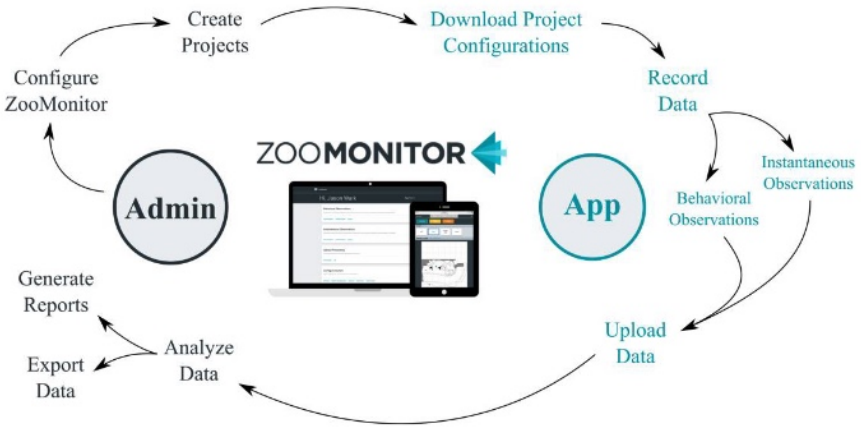


Figur 6.4: Skjermbilde av registrering av skadet sau på BeiteSnap [24]

6.5 ZooMonitor

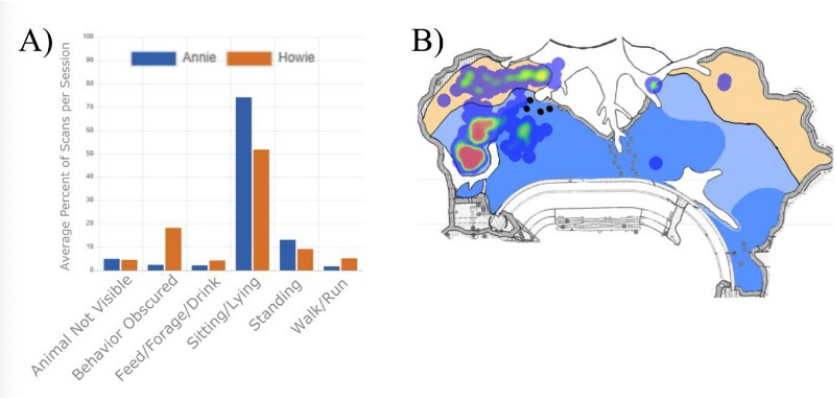
J.D. mfl. [25] har sett på hvordan innføring av en applikasjon kan hjelpe med å overvåke oppførsel og habitat til dyr i en dyrepark, samt hvordan dette påvirker velferden til dyrene. Forskningen har blitt gjennomført sammen med Zier Niemann Consulting og Lincoln Park Zoo, som ønsket en applikasjon til å overvåke dyrene i dyreparken. Tidligere har overvåkingen blitt gjennomført med bruk av penn og papir. Utvikling og forbedring av applikasjonen ble gjort ved hjelp av 15 dyrehager og akvarium.

ZooMonitor er delt inn i to områder, et for administratorbrukere og en for selve applikasjonen. Funksjonaliteten til ZooMonitor kan en se i figur 6.5 [25]. Ved internettilgang, kan administratorene opprette, kopiere og endre på prosjekter. På applikasjonen kan hvert prosjekt bli lastet ned på mobilen, slik at en har mulighet til å bruke applikasjonen uten internett. Videre kan data registreres uten internett, og er lagret på mobilen helt til brukeren kobler seg til internett og dataene blir lastet opp til en skytjeneste. Etter data har blitt lastet opp, kan administratorer se på dataene gjennom rapporter eller laste ned som CSV-filer.



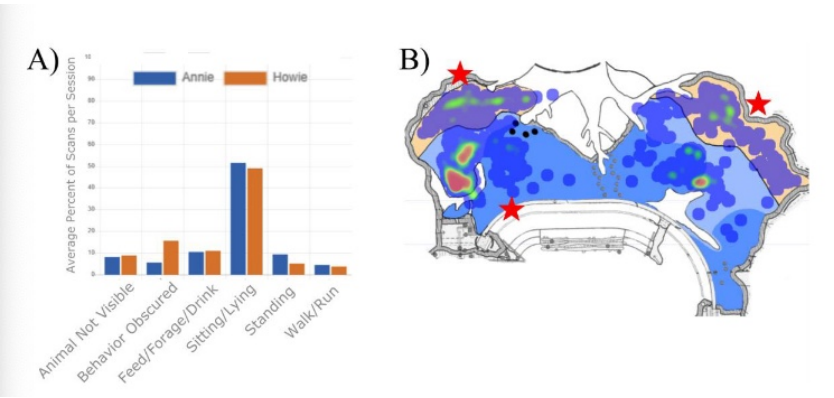
Figur 6.5: Oversikt over struktur til ZooMonitor applikasjon [25]

Applikasjonen har mulighet for å laste opp et kartbilde av området til dyrene, og en kan plassere de nåværende posisjonene til dyrene [25]. I tillegg kan brukerne skrive inn hendelser de observerer. De noterer inn oppførselen til dyrene, som kan være hvor mye de beveger seg, hvor synlige de er på område, og om de er mye i solen eller i skyggen. Fra figur 6.6 kan en se to grafer som ble generert av ZooMonitor. Grafene inneholder data fra april til september, med 5240 minutter med observasjoner av flodhester.



Figur 6.6: Grafer generert av ZooMonitor programvare [25]

Figur 6.6 B viser aktiviteten til flodhestene, der en kan se at de tilbringer mesteparten av tiden sin på venstre side i habitatet sitt. Etter dette resultatet, ønsket dyrepasserne og forskere fra Animal Welfare Science Program at dyrene skulle utnytte området bedre og øke aktiviteten til dyrene [25]. Løsningen ble å installere automatiske matstasjoner på tre lokasjoner. Resultatet av dette kan en se i figur 6.7 B, med matstasjonene markert som røde stjerner på figuren. En ser at flodhestene nå beveger seg mer rundt, og det er mye mer aktivitet på høyre side av grafen.



Figur 6.7: Grafer generert av ZooMonitor programvare etter installasjon av tre matstasjoner [25]

6.5.1 Oppsummering

Fra tidligere arbeid kom det frem at det eksisterte tre løsninger som tilbydde sporingsenheter for småfe, som sender posisjonen minst en gang i døgnet. Telespor brukte satellitt til å kommunisere, mens Smartbjella og FindMy brukte mobildekning. BeiteSnap var en løsning som gikk ut på at turgåere kunne ta bilder av et dødt eller skadet beitedyr, samt legge inn posisjonen til dyret innenfor et beiteområde. Bonden fikk varsel på mobiltelefonen, og kunne se på kartet hvor dyret var registrert. Den siste løsningen var forskningsprosjektet ZooMonitor, som registrerte dyrenes posisjon i en dyrehage. Dyrene ble visualisert ved bruk av heatmap, og målet var å øke aktiviteten til dyrene innenfor et område. I tillegg genererte ZooMonitor en rapport som viste oversikten over alle registreringene. Fra de eksisterende løsningene vil heatmap, generering av rapport og muligheten til å se registrerte dyr på kart være relevant for *BeiteSpor*.

Del III

Eget bidrag

Denne delen av rapporten tar for seg arbeidet som ble gjort for å få et komplett produkt. Det innebærer valg av utviklingsmetode, utarbeiding av kravspesifikasjoner, samt design og implementasjonen av løsningen.

Kapittel 7

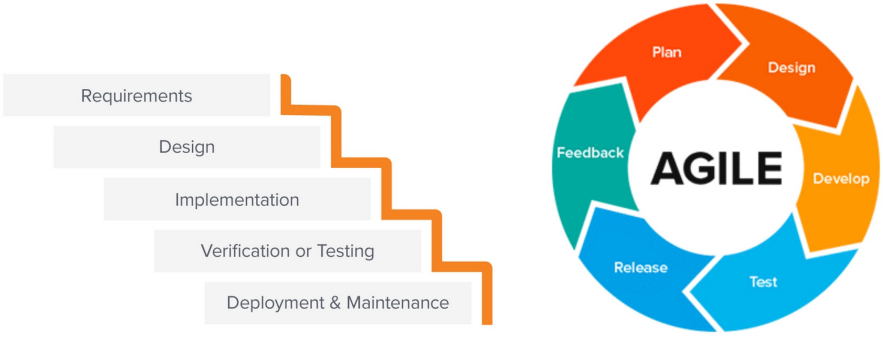
Valg av prosess

Vannfallsmetoden er en sekvensiell utviklingsmetodikk som handler om nøye planlegging, detaljert dokumentasjon og utførelse [26]. Vannfallsmetoden er sekvensiell, der det ikke er rom for å gjøre feil og gå tilbake for å endre. Metoden består av fem steg som er oppdelt i krav, design, implementasjon, validering eller testing, og til slutt produksjon og vedlikehold. Et alternativ er å bruke ulike team for å gjennomføre hvert steg i denne prosessen. Det er ikke et krav om at teamene skal kommunisere kontinuerlig, og en trenger ikke å gi statusrapport ved et bestemt tidspunkt eller et gitt tidsintervall. Vanligvis gjør en ferdig hvert steg, før en går videre på det neste (se venstre bilde på figur 7.1).

I 2001 kom Agile Manifesto som var en revolusjonerende og ny måte å gjennomføre prosjekter på [27]. Smidig metodikk er en iterativ tilnærming som blir brukt innenfor programvareutvikling, der team kan levere resultater til kundene sine hyppigere [28]. Fremfor å ha et stort team, deles teamene opp, samtidig som arbeidsoppgavene blir flere og mer konkrete. Dette fører til at deler av løsningen kan leveres og evalueres hyppigere, istedenfor å satse alt på en stor utrulling til slutt. Krav, planlegging og resultater blir evaluert kontinuerlig slik at teamene kan gjøre endringer fortløpende dersom det skulle oppstå problemer, eller at det i løpet av utviklingsprosessen kommer inn nye momenter som gjør at kravene endrer seg. [29]. Fra høyre bilde på figur 7.1 ser en smidig arbeidsmetodikk som en sirkel, der en runde tilsvarer en iterasjon.

Innenfor smidig utvikling er det Scrum og Kanban som er de mest kjente rammeverkene [27]. Scrum er et rammeverk som hjelper team med å jobbe sammen, der en skal lage verdi gjennom tilpassede løsninger på komplekse problemer [30]. Scrum-rammeverket består av å sette opp møter, verktøy og roller som hjelper teamene med å strukturere arbeidet sitt [31]. I smidig metodikk skal oppgavene være små, og ferdigstilles før en går videre til neste oppgave. Dette blir gjort i løpet av en sprint, som kan ha en lengde på en til fire uker. En sprint består av å planlegge hva som skal gjøres ved bruk av et Scrum Board (Sprint Planning), daglige møter der en gjennomgår hva en har gjort, hva en jobber med og hva en skal jobbe med fremover (Standup) [32]. I slutten av sprinten, går teamet gjennom og demonstrerer hva de har gjort (Sprint Review). Helt til slutt, for å avslutte sprinten, går teamet gjennom og sier hva som var bra og hva som kan forbedres (Sprint Retrospective). Deretter startes prosessen på nytt.

Ordet "Kanban" kommer fra Japan, og betyr visuelt bord. Kanban ble først brukt av Toyota til å planlegge produksjon [33]. I ettertid har det blitt utviklet til et rammeverk for programvareutvikling. Rammeverket krever direkte kommunikasjon mellom teammedlemmene for å oppnå best mulig resultat, og at de som jobber sammen er helt åpne om hvilke arbeidsoppgaver de jobber med [34]. I Kanban ønsker en å redusere unødvendig arbeid, uten å miste produktivitet. Målet er å lage et produkt som ikke skal koste mer, men øke verdien for kunden [33]. For å oppnå målet, blir det brukt et Kanban Board for å visualisere oppgavene som må gjennomføres. Kanban Board består av tre kolonner, som er inndelt i "To do", "In progess" og "Complete" [35]. På dette bordet lager en kort der en beskriver et gjøremål per kort. For å sikre at en ikke jobber med for mange oppgaver samtidig, har kanban en Work In Progess (WIP) grense, som sier maks antall gjøremål en kan jobbe med samtidig. Dette blir gjort for å hindre at en jobber på flere halvferdige oppgaver [36].

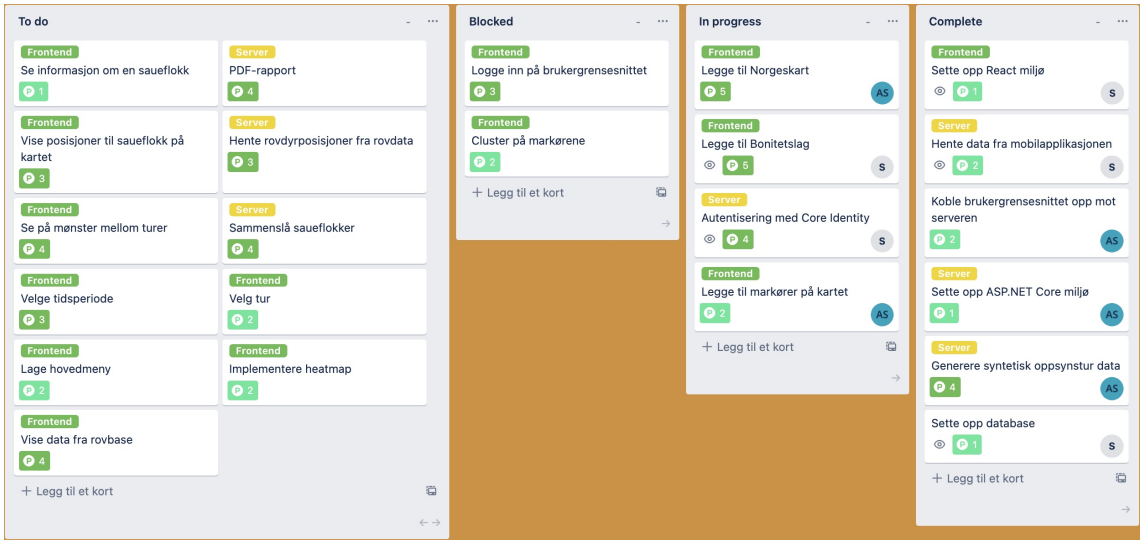


Figur 7.1: Vannfallsmetoden [26] og smidig metodikk [37]

Ved bruk av vannfallsmetoden må en planlegge alt på forhånd, før en går videre til neste steg. Ettersom utviklerne ikke visste alle detaljer ved sluttproduktet, fikk ny informasjon underveis og tilbakemelding fra veilder som måtte håndteres, var ikke vannfallsmetoden et alternativ. Dermed måtte det bli en smidig arbeidsmetodikk. Videre var spørsmålet om en skulle gå for Scrum eller Kanban. Både Scrum og Kanban bruker et bord der en henger opp kort med arbeidsoppgaver. Scrum jobber i sprinter der alle arbeidsoppgavene for en sprint blir forhåndsdefinert. Scrum bruker også 15 minutter Standup hver dag, Sprint Review av hva som har blitt gjort, og en Retrospective der en snakker om hva som kan forbedres. Ved bruk av Kanban jobber teamet med å få framgang i arbeidskortene sine, eksempelvis at teamet maks kan jobbe med fem oppgaver samtidig [38].

Med et utviklingsteam bestående av to personer, var det flere ting som gjorde at Kanban ble foretrukket fremfor Scrum. Sprint Planning og Retrospective etter hver sprint var lite effektivt. Det var ikke behov for å sette opp sprinter på et bestemt antall uker, ettersom utviklerne hadde et bestemt antall oppgaver som skulle utføres. Det var ikke viktig å prioritere oppgavene i ulike sprinter, så lenge alle oppgavene ble utført innen en gitt frist. I tillegg jobbet utviklerne ved siden av hverandre, som førte til at en visste hva den andre på teamet jobbet med til enhver tid. Dermed trengte en ikke Standup hver dag, ettersom utviklerne kommuniserte kontinuerlig. Videre passet Kanban bedre på grunn av Kanban Board, der en kun kan jobbe med et par oppgaver samtidig, og gjøre seg ferdig med de før en går videre.

For å opprette et Kanban Board brukte utviklerne Trello, et digitalt samarbeidsverktøy som skal hjelpe team med å organisere arbeidsoppgaver. I Trello kan en lage nye tavler, lister og kort, samtidig som en kan sette vanskelighetsgrad og tilegne medlemmer til hvert kort [39]. Fra figur 7.2 kan en se et Kanban Board fra Trello. Her ble de tre kolonnene "To do", "In progess" og "Complete" brukt, samtidig som det ble lagt til en ekstra kolonne, "Blocked" [40]. I denne kolonnen ble kort som ikke var mulig å gjennomføre plassert på grunn av at de var avhengig av andre kort. I Kanban skal det bli satt en WIP grense, slik at teamet ikke jobber med for mange oppgaver samtidig, og denne grensen ble satt til to oppgaver per person [40]. For å kartlegge hvor vanskelig en arbeidsoppgave var, fikk alle kortene en verdi fra en til fem, der én er lettest og fem er vanskeligst. Til slutt ble det satt en markør på hvert kort, som enten indikerte at arbeidsoppgaven var til brukergrensesnittet (grønn markør) eller til serveren (gul markør).



Figur 7.2: Trello ble brukt til å lage Kanban Board

7.1 Oppsummering

I prosessen måtte en velge mellom to typer utviklingsmetodikker, sekvensiell eller smidig. Ettersom det var behov for å gjøre endringer underveis i utviklingen, gikk utviklerne for smidig utviklingsmetodikk. Innenfor smidig utvikling var Scrum og Kanban de to mest kjente rammeverkene, og de hadde mange likheter. Begge rammeverkene brukte et Board for å sette opp hvilke oppgaver som skulle bli gjennomført. Valget falt på Kanban, da dette ble vurdert som mest effektivt og hensiktsmessig for et team på to personer.

Kapittel 8

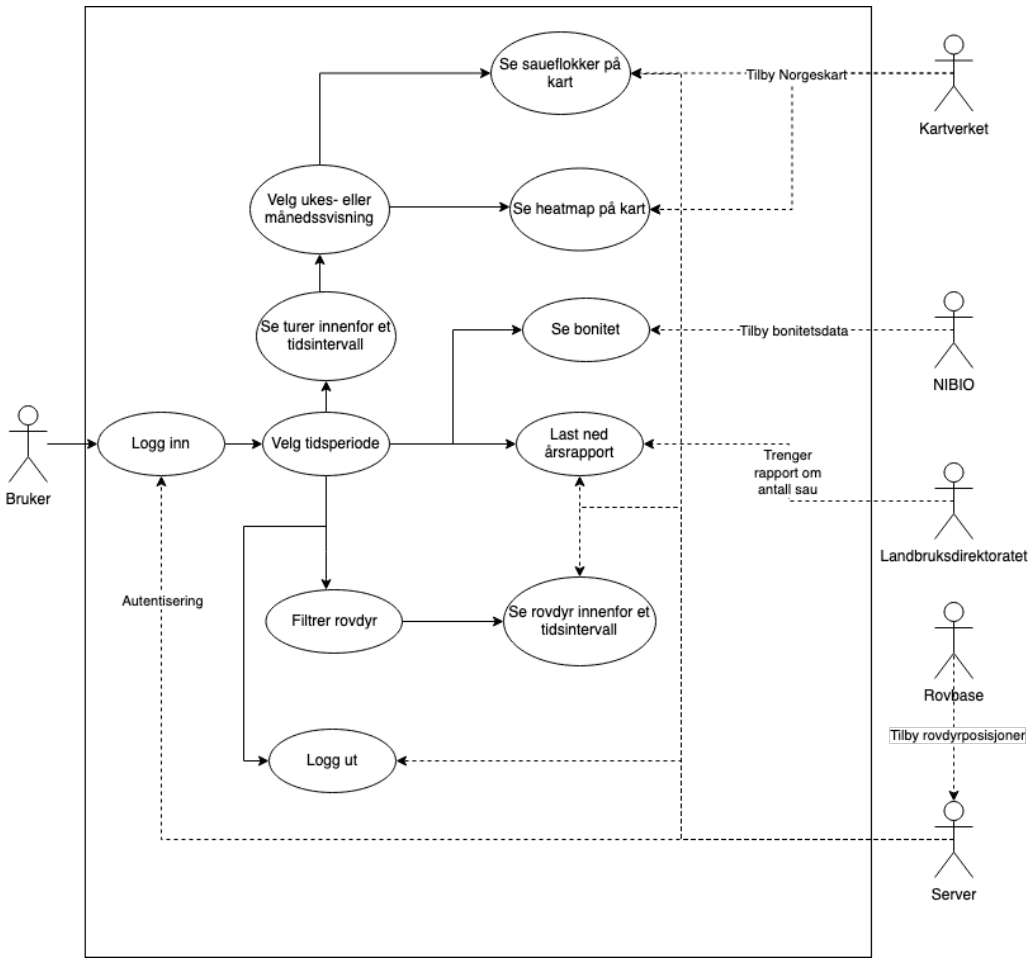
Programflyt og kravspesifikasjon

Før utviklingen startet var det viktig å få et felles bilde av den helhetlige funksjonen til *BeiteSpor*. Dette ble gjort gjennom Use Case diagram, etterfulgt av kravspesifikasjoner til funksjonalitetene i applikasjonen.

8.1 Use Case

Use Case diagram blir brukt for å beskrive funksjonaliteten til et system. Brukerne blir vist som strekfigurer, og pilene på diagrammet viser hvem som starter handlingen [41]. Use Case hjelper de involverte personene med å fokusere på de viktigste handlingene i systemet, men spesifiserer ikke hvilken metode en skal bruke for å lage løsningen. Videre hjelper Use Case diagram med å lage en avtale mellom kunde og oppdragsgiver, der diagrammet viser en oversikt over hvilke funksjoner systemet skal inneholde [42]. Dersom noen av partene har misforstått funksjonalitet i systemet, kommer det tydelig fram i Use Case diagrammet, og en kan løse dette før en går videre med implementasjon.

Fra figur 8.1 kan en se Use Case diagram for *BeiteSpor*. På venstre side av figuren ser en brukeren, som i dette tilfelle vil være bonden. Det eneste bonden kan gjøre når vedkommende går inn i webapplikasjonen, er å logge inn. Dersom det ikke eksisterer en bruker fra før, kan bonden opprette en, og deretter logge inn. Etterpå kan en velge tidsperioden for dataene som skal vises, som videre gir flere valgmuligheter. Bonden kan velge om webapplikasjonen skal vise turene i ukes- eller månedsvisning. Etter en har valgt hvordan en ønsker å se turene, kan en velge om en ønsker å representere turene i saueflokk- eller heatmapvisning. På høyre side av figur 8.1 kan en se at Kartverket tilbyr Norgeskart som kartlag når en skal se både saueflokkene og heatmap på kartet. På lik linje som Norgeskart tilbyr kartlag, tilbyr også Norsk institutt for bioøkonomi (NIBIO) et bonitetskartlag.



Figur 8.1: Use Case diagram for BeiteSpor

Det er blitt laget en ordning for tilskudd til bønder med småfe på utmarksbeite, som har blitt presentert i kapittel 4.4. For å få tilskuddet må bonden dokumentere alle fullførte tilsynsturer gjennom året. Dette må legges ved i søknadsskjemaet som sendes inn til Landbruksdirektoratet for å få tilskuddet. I midten av figur 8.1 ser en at det er mulig å laste ned en årsrapport. Denne kan bonden bruke som dokumentasjon på gjennomførte tilsynsturer.

Ettersom sau er utsatt for rovdyrangrep, er det relevant å vise hvor rovdyr befinner seg på kartet. Norsk institutt for naturforskning har dannet en selvstendig enhet som heter Rovdata, som registrerer rovdyr [43]. På 8.1 ser en at bonden kan filtrere rovdyr og vise de innenfor en bestemt tidsperiode. Tilgangen til rovdyrene blir gjort gjennom API-et til Rovdata, som heter Rovbase.

8.2 Kravspesifikasjon

Fra Use Case diagrammet som ble beskrevet i kapittel 8.1, ble det utarbeidet kravspesifikasjoner i samarbeid med veileder og domeneekspert, Hvasshovd. Kravene skal dekke den komplette funksjonaliteten til *BeiteSpor*.

8.2.1 Funksjonelle krav

Funksjonelle krav sier hva sluttløsningen skal inneholde, og hvilke funksjonaliteter som skal være implementert. Kravene vil være synlige i sluttproduktet, og kan bli testet av sluttbrukerne etter ferdig utviklet løsning [44]. Listen med krav ble laget før utviklingen begynte. I Tabell 8.1 kan en se de funksjonelle kravene til serveren.

ID	**Prioritet**	**Beskrivelse**
FK1	Høy	Mottakspunkt og lagring for data som sendes fra mobilapplikasjon
FK2	Høy	Hente data som er samlet inn fra en mobilapplikasjon om saueturer
FK3	Høy	Hente rovdyrhendelser (Bjørn, Gaupe, Jerv og Ulv) fra Rovbase
FK4	Høy	Lagre data i en skydatabase
FK5	Høy	Logge inn
FK6	Høy	Opprette bruker
FK7	Lav	Glemt passord
FK8	Høy	Generere saueflokker
FK9	Høy	Generere en PDF-årsrapport
FK10	Høy	Identifisere saueflokker fra tur til tur

Tabell 8.1: Funksjonelle krav for server

Serveren sin oppgave er å motta og lagre data som sendes fra mobilapplikasjonen, som innebærer gjennomførte turer (FK1), samt hente dataene som har blitt lagret på serveren og sende til webapplikasjonen (FK2). I tillegg henter serveren data fra Rovbase, for å vise rapporterte rovdyr på kartet (FK3). Videre må bøndene ha mulighet til å lagre nye data i databasen (FK4), som eksempelvis er å opprette nye brukere (FK6). Serveren skal gi en gyldig nøkkel til brukere som logger inn (FK5), og sende en epost dersom det trengs å opprette nytt passord (FK7). For å få nok data til å teste ut applikasjonen, genererer serveren saueflokker innenfor et bestemt område i et realistisk sauemønster (FK8). Mønsteret er laget ved hjelp av veileders innsikt i bevegelse til saueflokker. Etter en har generert data, skal serveren identifisere saueflokker etter hver tur (FK10). For å få en oversikt over dataene etter hvert år, kan serveren generere en PDF-rapport (FK9).

ID	Prioritet	Beskrivelse
FK11	Høy	Bruke Norgeskart som kartgrunnlag
FK12	Høy	Mulighet til å vise bonitet på kartet
FK13	Middels	Vise bondens tilsynsrute på kart
FK14	Høy	Vise posisjoner til saueflokker på kartet
FK15	Høy	Vise døde sauer
FK16	Høy	Klikke på en saueflokk, og se informasjon
FK17	Høy	Sammenslå posisjoner når en zoomer ut
FK18	Middels	Ulike ikon for observasjon, spor og døde rovdyr
FK19	Middels	Velge hvilke tidsintervall en vil se turer fra
FK20	Middels	Velge om en vil se saueflokkene for hver uke eller måned
FK21	Middels	Filtrere hvilke rovdyr en vil se på kartet
FK22	Middels	Velge hvilke turer en vil se
FK23	Lav	Ulik farge på posisjonsikonene
FK24	Lav	Fast navigeringsmeny

Tabell 8.2: Funksjonelle krav for brukergrensesnittet

Mye av funksjonaliteten på brukergrensesnittet er basert på at det eksisterer et detaljert kart av det norske landskapet (FK11). I tillegg ønsker bøndene å se hvor på kartet det er gode beiteforhold, med bruk av bonitet (FK12). På selve kartet ønsker en å se hvor saueflokker (FK14), og eventuelt døde sauer befinner seg (FK15). Det skal være mulig å klikke på de ulike flokkene, slik at en kan se en mer detaljert oversiket (FK16). Ettersom det blir lagt til en saueflokk hver uke eller måned (FK20), kan dette føre til at det på et tidspunkt blir mange saueflokker. En mulighet for å sammenslå posisjoner ved zooming kan gjøre systemet mer oversiktlig og raskere (FK17)). For å gjøre systemet mer brukervennlig, skal brukeren kunne velge tidsperiode for turene (FK19), og hvilke turer en vil se (FK22). I tillegg til å vise saueflokker, skal kartet ha mulighet til å vise rovdyr (FK21), med ulike farger (FK23) og ikoner på posisjonene (FK18). Til slutt er det ønskelig med en fast meny

som alltid er tilgjengelig (FK24).

8.2.2 Ikke-funksjonelle krav

Ikke-funksjonelle krav går ut på å måle kvaliteten til systemet. For å se om systemet oppfyller kravene kan en sette opp en liste med målbare krav [44]. Denne listen ble satt opp før utviklingen startet, slik at en hadde klart for seg hvilke krav applikasjonen skulle oppfylle.

Sikkerhet
Sikkerhet går ut på å sikre at data kun er tilgjengelig for personer med de rette tilgangene [45]. Sikkerhet inkluderer konfidensialitet og autentisering for å sikre at data er beskyttet.

1. Systemet skal kun være tilgjengelig for personer som har opprettet bruker
2. Hver bruker skal kun se sine egne turer
3. Data skal sendes kryptert

Vedlikehold
Vedlikehold går ut på å lage et system som skal være mulig å modifisere, konfigurere og utvide på en effektiv måte [45]. Et system som skal være vedlikeholdbart bør ha sterkt kohesjon (high cohesion), og løs kopling (loose coupling). Det betyr å lage komponenter som er uavhengige, der hver komponent har kun en oppgave som er veldefinert. Koden skal være godt dokumentert, og en skal enkelt kunne sette seg inn i koden som andre har skrevet.

1. Brukergrensesnittet skal bygges opp av komponenter, slik at deler av brukergrensesnittet kan skiftes uten å påvirke resten av applikasjonen
2. Serveren og brukergrensesnittet skal være to uavhengige deler
3. Systemet skal være godt dokumentert, og det skal være enkelt for andre utviklere å overta arbeidet

Brukbarhet
Brukbarhet handler om hvor enkelt det er å bruke et spesifikt produkt [46]. Dette inkluderer hvor enkelt det er å navigere mellom sidene i brukergrensesnittet, og hvor enkelt det er å forstå og bruke funksjonaliteten i applikasjonen.

1. Systemet skal være enkelt å bruke
2. Systemet skal være tilpasset datamaskin
3. Personer med lite teknologisk erfaring skal kunne bruke systemet
4. Brukeren skal kunne bruke systemet etter opplæring
5. Etter opplæring skal brukeren kunne:
    - Opprette bruker og logge inn
    - Velge periode i applikasjonen
    - Filtrere rovdyr
    - Navigere mellom turer i uke- og månedsvisning
    - Slå av og på, samt regulere synlighet av bonitet
    - Slå av og på heatmap og saueflokker
    - Laste ned PDF-rapport

Ytelse

Et system skal bygges slik at brukerne slipper å vente på at data blir lastet inn, at systemet blir overbelastet eller at ressursene blir brukt på en ineffektiv måte [45]. Ytelsen kan bli redusert dersom en kjører mange spørringer til serveren, eller om SQL-spørringene er kompliserte og dårlig gjennomtenkt. Ut fra dette har vi kommet frem til følgende krav for ytelse:

1. Applikasjonen skal kunne hente data fra serveren innen 3 sekunder med 30mb/s internetthastighet
2. 100 brukere skal kunne bruke systemet samtidig

8.3 Oppsummering

Det ble laget et Use Case diagram som beskrev alle de essensielle aktørene, og funksjonene i *BeiteSpor*. Med utgangspunkt i Use Case diagram, ble det utarbeidet både funksjonelle og ikke-funksjonelle krav i samarbeid med veileder Hvasshovd. Dette resulterte i 10 funksjonelle krav knyttet til serveren og 14 knyttet til brukergrensesnittet. De funksjonelle kravene for serveren dekket mottak, lagring og distribusjon av data fra tilsynsturer, hente data fra Rovbase og brukerhåndtering. Alle de nødvendige visualiseringene i applikasjonen ble dekket i de funksjonelle kravene for brukegrensesnittet. Til slutt ble det utarbeidet ikke-funksjonelle krav for sikkerhet, vedlikehold, brukbarhet og ytelse, for å kunne måle kvaliteten til systemet.

Kapittel 9

Valg av teknologi

I dette kapittelet vil teknologivalget for serversiden og brukergrensesnittet bli begrunnet. Det var viktig å velge utviklingsverktøy som kunne oppfylle alle kravene som ble beskrevet i kravspesifikasjonen.

9.1 Valg av teknologi til brukergrensesnittet

Første del av valgprosessen gikk ut på hvilke teknologier som skulle velges for brukergrensesnittet. Teknologiene skal dekke nødvendige funksjonalitet for visning i applikasjonen og bidra til en mest mulig effektiv utvikling.

9.1.1 Rammeverk til brukergrensesnittet

For å utvikle brukergrensesnittet var det ønskelig å bruke et JavaScript/TypeScript-rammeverk. Det eksisterer flere rammeverk som er mulig å velge, men det stod mellom React, Vue og Angular. Alle tre rammeverkene består av komponenter, men det er Vue som skiller seg mest ut i komponentoppbyggingen [47, 48, 49]. En Vue-fil er delt opp i tre deler, en for HTML, en for funksjonalitet og en for CSS [49]. Angular har en fast og bestemt måte en skal bygge opp komponenter og moduler, der alt skal følge en bestemt mal [48]. Vue og React er mer fleksible på hvordan en bygger opp komponenter, og hver utvikler har mer frihet til selv å velge hvordan koden struktureres [47, 49]. I forprosjektet, kapittel 5, ble det laget en mobilapplikasjon ved hjelp av React-Native som er bygget på React. Ettersom forprosjektet inneholdt mange komponenter som kunne komme til nytte i webapplikasjonen, falt valget på React. I tillegg er React det mest voksende og populære rammeverket i henhold til trender i *StackOverflow* [50].

TypeScript

For å utvikle i React kan en bruke både JavaScript og TypeScript [47]. Ettersom utviklerne ønsket å bruke et programmeringsspråk som benyttet seg av typesjekking, ble TypeScript valgt. TypeScript legger til ekstra syntaks, typer, som øker kvaliteten på koden, og som reduserer sjansen for at det oppstår kjøretidsfeil [51]. Etter en har skrevet kode i TypeScript, konverteres den til JavaScript. TypeScript er støttet i flere IDEer, og gir god støtte for refaktorering, automatisk-utfylling og typesjekking i sanntid.

Material UI

Et komponentbiblotek kan hjelpe med å øke effektiviteten i utviklingen, siden biblioteket inneholder ferdige komponenter som har funksjon- og designtilpasninger. Det er flere aktører som har bygd kompenentbiblotek i React [47]. Et krav var at komponentene er enkle å modifisere og tilpasse til eget behov. Valget ble derfor Material UI, som begge utviklerne hadde erfaring med fra før. Material UI tilbyr både gratisversjon, og betalende medlemmer [52]. Gratisversjonen har tilgang til alle de generelle komponentene, som gjør at utviklerne slipper å lage alt fra bunnen av.

9.1.2 Kartløsning

For å bestemme hvilken type kartløsning som skulle implementeres, måtte en velge et kart som oppfyller kravene FK11 og FK12 fra kravspesifikasjonen i kapittel 8.2. Fra kravene står det at Norgeskart skal være hovedkartlaget, med mulighet for å vise bonitetslag på kartet. Det betyr at kartløsningen som blir valgt, må ha mulighet til å legge på flere kartlag, der en kan legge til og fjerne lag under kjøring. Både Leaflet og Google Maps har kartløsninger i JavaScript. Ved å sammenligne funksjonaliteten til Leaflet og Google Maps, viser det seg at begge to tilbyr den funksjonaliteten som er viktig for webapplikasjonen [53, 54]. Leaflet tilbyr gratis bruk av APIet dersom en ikke skal tjene penger på det [53], der Google Maps på andre siden tilbyr en startkreditt, slik at en slipper å betale de første månedene [54]. I følge Wappalyzer som er anerkjent av blant annet Google, Amazon og Microsoft blir Google Maps brukt på nærmere 1 900 000 nettsider, i motsetning til Leaflet som blir brukt på rundt 300 000 nettsider [55]. Både Leaflet og Google Maps har dokumentasjon som er oversiktlig. Basert på argumentene, er begge gode alternativer som kartløsninger til bruk i webapplikasjonen. Valget ble Google Maps, ettersom deres kart er mest populært, og har en mer detaljert dokumentasjon sammenlignet med Leaflet.

9.2 Valg av teknologi til serveren

Serveren skal ta imot data fra mobilappliakasjonen og levere data til brukergrensersnitt. Valget av teknologier gikk derfor ut på finne rammeverk for å lage et API til mottak, lagring og distribuering av data, samt publisering av applikasjonen.

9.2.1 Programvarearkitektur

Det er ønskelig at serveren skal være uavhengig av både brukergrensesnittet i webapplikasjonen og mobilapplikasjonen. Oppgavene er å motta data fra mobilapplikasjonen og formidle data til webapplikasjonen, som Representational State Transfer (REST) tilfredsstiller. REST er en programvarearkitektur som beskriver hvordan en kan strukturere API til å være skalerbare, uniforme og uavhengige [56]. REST består av seks veiledende prinsipper [57]:

- **Uniformt grensesnitt:** Grensesnittet må identifisere alle ressursene som er involvert i samhandlingen mellom klient og server. Ressursene som blir brukt i responsen til serveren skal ha en uniform representasjon, og inneholde informasjon som er nødvendig for å forstå meldingen.

- **Klient-server:** En skal følge prinsippet som handler om å separere klient og server, slik at hver del kan bli utviklet uavhengig av den andre.
- **Tilstandsløs:** Hver forespørsel som blir sendt fra klient til server skal være uavhengig og inneholde all informasjon som er nødvendig for å forstå og fullføre forespørselen. Serveren skal ikke ha tilgang eller kjennskap til tidligere forespørsler som har blitt sendt.
- **Hurtigminne:** Hver respons skal implisitt eller eksplisitt markere om det er mulighet for å bruke hurtigminne eller ikke på inneholdet.
- **Lagdelt arkitektur:** Det er mulig å lage en arkitektur som består av flere komponenter, der hver komponent har sitt eget ansvarsområde.
- **Kode etter forespørsel (Valgfri):** Utvider klientens funksjonalitet med å tilby nedlasting og mulighet for å kjøre kode med skript.

9.2.2 Rammeverk til serveren

ASP .NET Core (ANC) er en *open source* versjon av Microsofts ASP .NET, som har over 100 000 bidrag på Github [58]. ANC kan bli utviklet på de største operativsystemene som MacOS, Windows, og Linux, og på tvers av operativsystemene. Rammeverket er fleksibelt med tanke på hvilken versjon en bruker, ettersom en server kan inneholde flere ulike versjoner av ANC. Pakkebehandleren til rammeverket er NuGet, som gjør det mulig å bruke pakker som andre utviklere har laget. ANC har god dokumentasjon, som er laget av Microsoft. Valget falt på ANC ettersom begge utviklerne hadde erfaring med rammeverket fra før, og det har god integrasjon med andre Microsoftløsninger.

Entity Framework Core

Entity Framework Core (EFC) er en *open source* kryssplattform som blir brukt til å aksessere databaser [59]. EFC er et bibliotek som kan bli lagt til i ANC ved hjelp av NuGet pakkebehandler. EFC er en *object-relational mapper* (ORM), som vil si at det blir laget .NET objekter av databasestrukturen. Dette fører til at en kun trenger å forholde seg til ANC sine objekter, og ikke databaseobjektene. For å opprette .NET objekter som er identiske med databasen finnes det to alternativ, generere de fra den eksisterende databasen, eller opprette objektene i kode først, og deretter generere databasen basert på koden. Etter at en modell er laget, kan Entity Framework Migrations (EFM) brukes til å holde kontroll på endringer i databasemodellen. EFM hjelper med å rulle ut nye versjoner av databasemodellen, eller gå tilbake til tidligere versjoner.

ASP .NET Core Identity

ANC har et eget rammeverk, Identity, som blir brukt til å håndtere brukere. Dette innebærer innlogging, utlogging, passord, profil, roller, token og epost [60]. Identity tilbyr et grensesnitt med innlogging, eller muligheten for å opprette sitt eget innloggingsfelt med brukernavn og passord. Det er også mulig å implementere sosiale medier som innlogging, slik som Facebook, Google eller Twitter.

9.2.3 Database- og publiseringsløsning

For både database- og publiseringsløsning ble det valgt Azure, som tilbyr løsninger som hjelper med å bygge, administrere og rulle ut programmer ved hjelp av en gruppe skytjenester [61]. I likhet med Azure er det flere aktører, eksempelvis Amazon og Google, som leverer skytjenester. Siden Azure er laget av Microsoft, i likhet med ANC, gjør det enklere å koble sammen løsningene. Derfor ble Azure SQL Database og Azure App Service foretrukket som henholdsvis database- og publiseringsløsning.

Azure SQL Database

Azure tilbyr relasjonsdatabaser i skyen, som en kan koble seg opp mot og aksessere data [62]. Databasene er skalerbare, der en kan styre hvor mye minne, prosesskraft og lagringsplass databasen skal få til enhver tid. Skyløsningen har et kontrollpanel der en får oversikt over sikkerheten og oppetiden til databasen. Azure har utviklet egen maskinlæringsmodell som skal hjelpe med å tilpasse den mest optimale ytelsen basert på forbruket [63]. For å koble seg til databasen, genererer Azure en streng som en kan bruke i EFC og ANC. For å aksessere data i databasen må en ha en gyldig IP-adresse, som en registrerer i kontrollpanelet til Azure [64].

Azure App Service

Azure App Service (AAS) tilbyr bygging, distribuering og skalering av applikasjoner på nettet [65]. AAS fungerer både for Microsofts egne produkter som .NET og .NET Core, samt med Node, Java og Python. Visual Studio, som er IDE-en til .NET Core, tilbyr direkte publisering av applikasjoner opp til Azure. Både før en publiser, og i ettertid er det mulig å legge til kontinuerlig integrasjon og kontinuerlig levering. En kan sammenkoble Azure-kontoen med *Github repository*, slik at hver gang en ny kode blir lastet opp til Github, vil den automatisk bli lastet opp i skyen.

9.3 Oppsummering

Brukergrensesnittet ble utviklet med rammeverket React, med TypeScript. Valget av React ble gjort på bakgrunn av popularitet, og at React-Native ble brukt som rammeverk i forprosjektet. Material UI ble brukt som komponentbibliotek, for å minske arbeidet med design og funksjoner i applikasjonen. For å vise kart i brukergrensesnittet, ble Google Maps brukt som kartløsning. Det eksisterte andre alternativer som tilbydde den nødvendige funksjonalitet, men valget falt på Google Maps på grunn av dens popularitet og gode dokumentasjon. Serveren ble laget med *open source* rammeverket ANC, som utviklerne hadde brukt tidligere. ANC har god integrasjon mellom tjenestene som tilbyr publisering, med bruk av AAS og databaseløsningen i Azure. Både brukergrensesnittet og serveren ble publisert med hjelp av AAS, slik at begge deler av applikasjonen ble publisert på samme måte.

Kapittel 10

Design

Kapittelet vil ta for seg designprosessen av brukergrensesnittet til webapplikasjonen. Det vil bli presentert hvilke verktøy som ble brukt, hvilke valg som ble tatt og resultatet av prosessen.

10.1 Brukeropplevelse

Før en skal lage en prototype, er det essensielt å vite hva brukeren ser etter, og hva flertallet av brukerne mener er et akseptabelt brukergrensesnitt. Nielsen [66] har skrevet ti generelle prinsipper for hva en bør tenke på når en skal designe et brukergrensesnitt. Nielsen er medgrunnlegger av Nielsen Norman Group som er verdensledende innenfor forskningsrelatert brukeropplevelse. De ti generelle prinsippene for brukeropplevelse er følgende:

BO1 **Synlighet av systemets status**: Brukerne skal alltid være informert om hva som foregår i systemet, og få tilbakemelding etter en gitt tid. Det blir lettere å stole på et system der en vet hva som skjer, basert på tidligere handlinger.

BO2 **Likhet mellom systemet og den virkelige verden**: Systemet bør bruke ord, uttrykk og konsepter som brukeren har kjennskap til, fremfor å kommunisere på maskinspråk. Informasjon bør komme i en naturlig og logisk rekkefølge, slik som brukeren er vant med fra virkeligheten. Det vil være lettere for brukeren å sette seg inn i det aktuelle brukergrensesnittet, dersom responsen fra en handling i virkeligheten gjenspeiler brukergrensesnittet.

BO3 **Brukerkontroll og frihet**: Ofte utfører brukere handlinger ved et uhell. Brukeren skal alltid ha en mulighet til å angre, eller gå tilbake til forrige steg uten å gå gjennom en omfattende prosess. Det vil skape en frihet hos brukeren når de vet at de alltid kan gå tilbake dersom de skulle gjøre en feil. Samtidig vil det hindre frustrasjon hos brukeren ettersom de slipper å stå fast på en side på grunn av at de ikke klarer å komme seg tilbake.

BO4 **Konsistent og standarder**: I et system skal ikke brukeren være i tvil om forskjellige ord, situasjoner eller handlinger betyr det samme. På grunn av at brukeren interagerer med flere brukergrensesnitt og andre nettsider, bør like handlinger på tvers av brukergrensesnitt være mest mulig generelle, for å unngå forvirring.

BO5 **Forebygging av feil**: Et godt system gir beskrivende meldinger, dersom det skulle oppstå feil, men det beste er å unngå at feil forekommer i utgangspunktet. Systemet må enten eliminere alle feil, eller tilby brukeren en ekstra bekreftelse på at en ønsker å utføre en bestemt handling. Konsekvensene av handlinger må komme tydelig frem.

BO6 **Forståelse fremfor memorisering**: En bør minimere hva brukeren skal memorere ved å lage alternativ, handlinger og elementer synlige for brukeren. En side bør alltid inneholde all nødvendig informasjon, slik at brukeren slipper å huske informasjon fra en tidligere side. Brukergrensesnittet bør være designet slik at en forstår hvordan en bruker det, fremfor å memorere hvordan en utfører en bestemt handling.

BO7 **Fleksibilitet og effektivt å bruke**: Snarveier kan hjelpe de mer erfarne brukerne med å navigere rundt på et nettsted. Disse snarveiene kan være gjemt, slik at de ikke forvirrer de uerfarne brukerne. Fleksible løsninger gjør at hver bruker kan velge hva som passer best for hver enkelt.

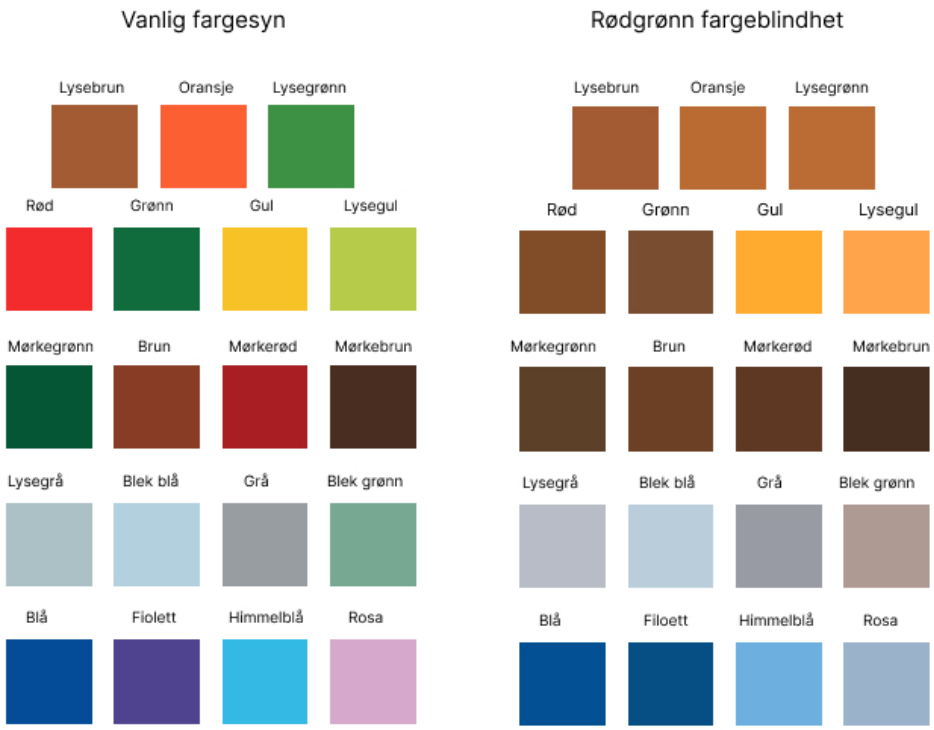
BO8 **Estetisk og minimalistisk design**: Informasjon som ikke er relevant for brukeren bør ikke være tilgjengelig på brukergrensesnittet. Ekstra informasjon som ikke er relevant vil konkurrere med den informasjonen som er relevant for brukeren. De visuelle elementene på nettsiden bør støtte brukerens mål på brukergrensesnittet.

BO9 **Gjenkjenne, diagnostisere og gjenopprette feil**: Feilmeldinger bør tydelig beskrive hva som er problemet, og på et språk som brukeren forstår. Feilmeldingen bør også si hvordan brukeren kan løse det oppståtte problemet.

BO10 **Dokumentasjon**: Noen ganger kan det være nødvendig å se på dokumentasjonen til et system, for å forstå hvordan en skal utføre en handling. Dokumentasjonen skal være lett tilgjengelig, presis og stegvis forklare hvordan en utfører en bestemt handling.

10.2 Fargevalg

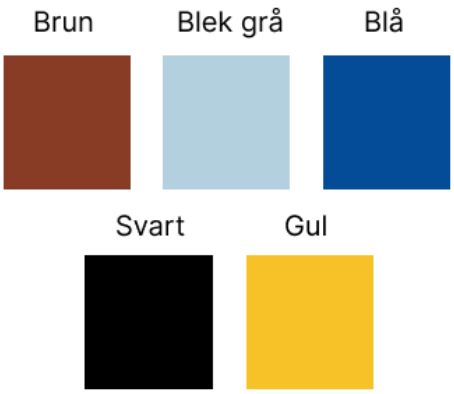
Web Content Accessibility Guildelines (WCAG) 2.0 er en standard som er laget av organisasjonen W3C, for å kunne tilby en rekke anbefalinger for hvordan en lager innhold på internett mer synlig og tydelig [67]. Retningslinjene er laget for de som har funksjonshemminger, som inkluderer dårlig syn og fargeblindhet. Det finnes både rødgrønn og blågul fargeblindhet, som er viktig å ta høyde for når en skal forme et brukergrensesnitt. I følge WCAG 1.4.1b, sies det at forklarende tekst eller andre visuelle virkemiddel skal gi samme informasjon som er formidlet gjennom bruk av farge [68].



Figur 10.1: Oversikt over hvordan farges sees med rødgrønn fargeblindhet [69]

I følge Norges blindeforbund er mellom 5 og 8 % av alle menn fargeblinde, og 0.3 % av kvinner er fargeblinde. Flesteparten av disse sliter med å skille mellom rødt og grønt, og har rødgrønn fargeblindhet [70]. For å kunne bedre forstå hvordan det er å se farger for en som er fargeblind, kan en se et fargepaletteksempel på figur 10.1. Her kan en se hvordan vanlige farger blir sett på fra personer med rødgrønn fargeblindhet. Figuer viser at de fleste varianter av rød, oransje, og grønn blir sett på som brune.

Basert på figur 10.1, ble det laget en fargepalett som skulle unngå den mest vanlige fargeblindheten, rødgrønn fargeblindhet [70]. Fra figur 10.2 kan en se fem farger som skiller seg ut og som ikke blir brune dersom en sliter med rødgrønn fargeblindhet.



Figur 10.2: Fargepalett for webapplikasjon

10.3 Prototype

Det finnes flere typer prototyper, fra papir til digitale varianter [71]. En kan velge hvor mye innsats en skal legge inn i hver prototype, om en kun vil teste en helt enkel flyt i løsningen, til mer avanserte løsninger som må inneholde spesifikke detaljer. Fordelen med å lage prototype er å spare tid, penger og å øke effektiviteten på utvikling.

For å designe prototypen stod valget mellom å bruke papirprototype eller å bruke et digitalt prototypeverktøy. Fordelen med å bruke papirprototype er at det er kjapt, effektivt og billig [72]. Hvem som helst kan lage forslag til papirprototype, både veileder og utvikler. Ulempen er at det blir lavnivå-skisser, med mindre detaljer, der en fokuserer mer på flyten i programmet. Ved test av papirprototyper må det foregå fysisk, der en person navigerer mellom skissene. Med digitale prototyper derimot, kan det testes over internett, og legge til automatisk linking mellom sider [71]. Skissene kan lages mer detaljerte, som kan være nærmest identiske med den originale løsningen. For å lage detaljerte skisser, vil det føre til at en må sette av lenger tid for å utvikle prototypene.

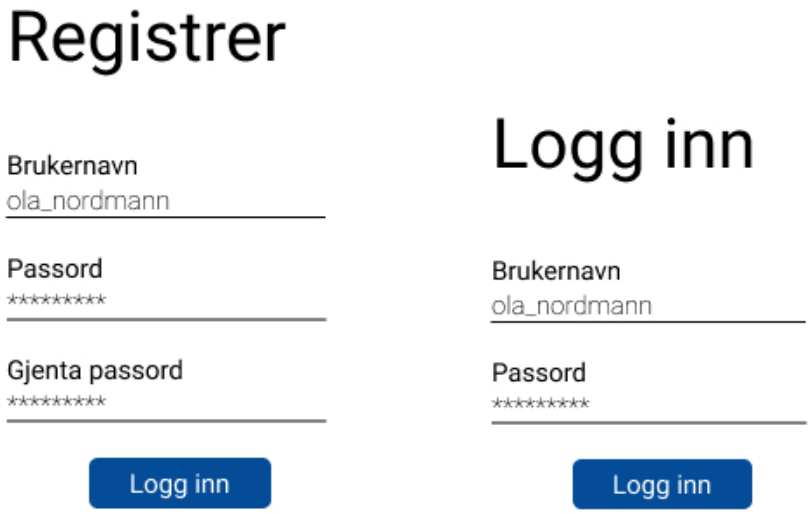
Ettersom vi ønsket å utvikle prototyper som var nærmest identiske med løsningen som skulle utvikles, falt valget på digital prototype. Valget stod mellom Adobe XD og Figma, siden begge programmene har gode løsninger for å jobbe i sanntid på samme prototype. Adobe har en gratis versjon som kun varer i syv dager [73]. Med Figma kan en opprette en gratisbruker, der en kan velge å aldri gå over til et betalende abonnement [74]. Figma ble derfor valgt som digitalt prototypeverktøy, ettersom vi ønsket muligheten til å endre prototypen i mer enn en uke uten å gå over til å et betalende abonnement. For å utvikle en prototype i Figma, ble Nielsen [66] ti punkter for brukeropplevelse brukt som grunnlag, som er beskrevet i kapittel 10.1.

10.3.1 Resultater

I dette delkapittelet vil vi ta for oss resultatene som kom fra designprosessen. Det ble utviklet en prototype til hver side i brukergrensesnittet, slik at utviklingsprosessen ville bli mer effektiv.

Autentisering

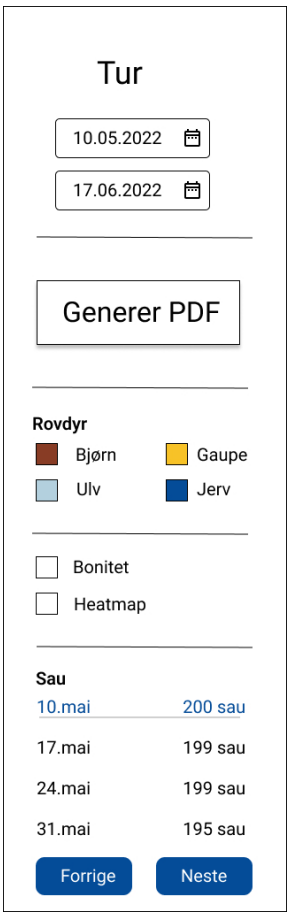
Fra Figur 10.3 kan en se brukergrensesnittet for registrering og innlogging av en bruker. For å opprette bruker må det registreres brukernavn, et passord og gjenta passord for å sjekke at passordet ble skrevet inn korrekt. Etter brukeren er opprettet, kan en logge inn med brukernavn og passord.



Figur 10.3: Prototype av registrering og innlogging av bruker

Meny

Menyen (figur 10.4) er en av de viktigste komponenten til nettsiden, ettersom mye av navigeringen foregår der. Menyen er delt inn i fem segmenter som er skilt med en midtstrek. Øverst på menyen er det en oversikt over hvilken tidsperiode som er valgt. Her kan en endre dato ved å klikke på kalender-symbolet. Nederst i menyen finner en oversikt over hvilke turer som hører til den valgte tidsperioden. Det vil alltid være én tur som er valgt, og som blir vist på kartet. En kan endre hvilken tur som er valgt ved å klikke på "neste"- eller "forrige"-knappen nederst på siden.



Figur 10.4: Prototype for meny

Den andre delen i menyen inneholder funksjonalitet for å laste ned årsrapport om sauene. Etter at en klikker på "Generer PDF" får en lastet ned en PDF, som er et eksempel på i figur 10.5. Rapporten viser en oversikt over hvor mange hvite, svarte og brune sauer som har blitt observert på hver tur, i tillegg til hvor mange sauer som er funnet døde. Videre i den tredje delen i menyen, kan en velge å filtrere rovdyr på kartet. Hvert rovdyr har ulik farge, som er beskrevet i kapittel 10.2. Hvert rovdyr kan vises og skjules ved å klikke på den tilhørende boksen til rovdyret. I fjerde seksjon har brukeren mulighet til å vise bonitet på kartet, som et ekstra lag. Sammen med bonitet kan brukeren også slå på heatmap visning av saueflokkene i dette segmentet.

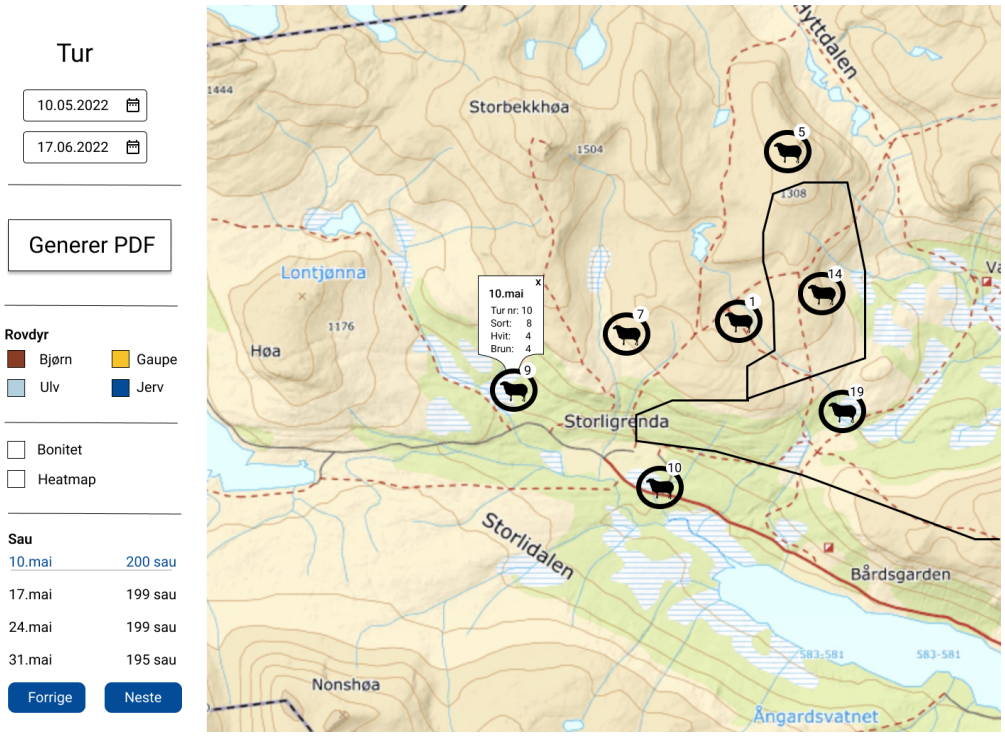
Rapport fra 10.05.2022 - 31.05.2022

Dato	Hvit sau	Svart sau	Brun sau	Død sau
10.05.2022	15	20	10	1
17.05.2022	13	19	8	5
24.05.2022	13	17	8	2
31.05.2022	12	17	8	1

Figur 10.5: Årsrapport over sauene

Tur

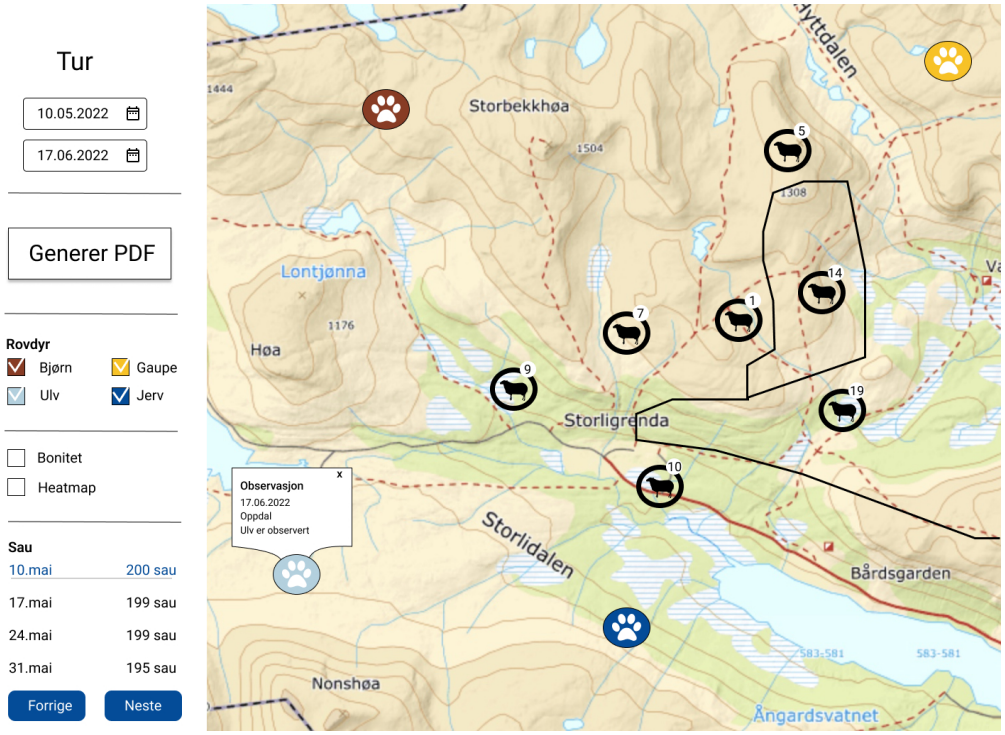
Ved å klikke på en spesifikk tur i menyen, kan en se hvor sauene ble observert på kartet. På figur 10.6 er det en representasjon av en gjennomført tilsynstur med observerte saueflokker. Her har hver saueflokk fått et lite hvitt tall oppe til høyre, for å symbolisere hvor mange sauer som er blitt observert på det bestemte området. Ved klikk på en saueflokk, vil brukeren få opp en hvit boks, som en ser på saueflokken lengst til venstre på figuren. Boksen inneholder informasjon om når turen fant sted, hvilket tur-nummer flokken hører til, og hvor mange sauer det er av hver farge. Den svarte linjen viser ruten brukeren har gått på tilsynsturen.



Figur 10.6: Prototype av en valgt tur

Rovdyr

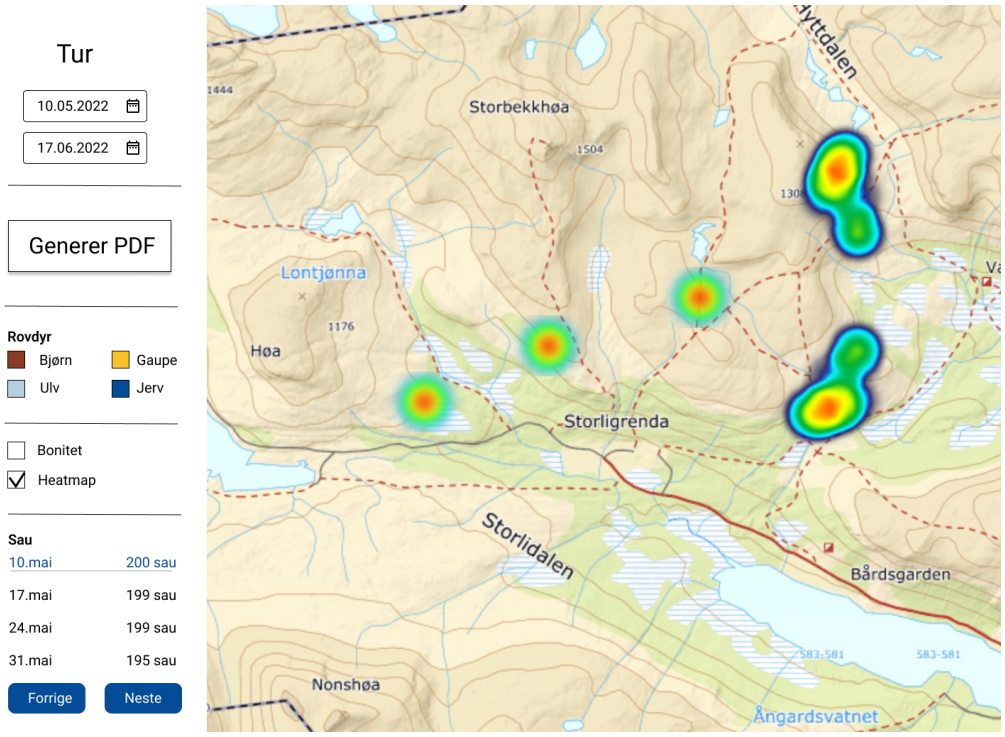
For å se rovdyr på kartet, klikker en på de fargede boksene med navnene til de ulike rovdyrene i menyen. Når et rovdyr er huket av, vil det bli vist på skjermen som et fotavtrykk i rovdyrets farge, slik en kan se på figur 10.7. Dersom en klikker på en spesifikk rovdyrhendelse, åpnes en informasjonsboks med attributter som dato for observasjon, hvilket type rovdyr det er, hvordan det ble observert og lokasjon.



Figur 10.7: Prototype av en tur på applikasjonen

Heatmap

Mellom rovdyr og informasjon om turene til saueflokkene, kan en velge funksjonaliteten heatmap. Ved å huke av heatmap, vil det bli vist hvor på kartet det er mest saueaktivitet (Figur 10.8). Heatmap vil bli vist istedenfor saueflokker på kartet. Fargegraderingen i et heatmap går fra rødt til blått, der rødt betyr mye aktivitet, mens blått betyr lite aktivitet.



Figur 10.8: Prototype av heatmap på kartet

10.4 Oppsummering

For å utvikle en digital prototype til webapplikasjonen, ble Figma brukt som designverktøy. Nielsens ti prinsipper for brukeropplevelse ble brukt som grunnlag for hvordan designet skulle se ut. I tillegg til å fokusere på brukeropplevelse, var det viktig å ta hensyn til personer med funksjonshemminger, som fargeblindhet. Derfor ble det laget en pallette med fem farger som tok hensyn til personer med rødgrønn fargeblindhet.

Kapittel 11

Implementasjon

I dette kapittelet vil struktur for databasen, serveren og brukergrensesnittet bli gjennomgått, samt implementasjonen av første versjon av webapplikasjonen.

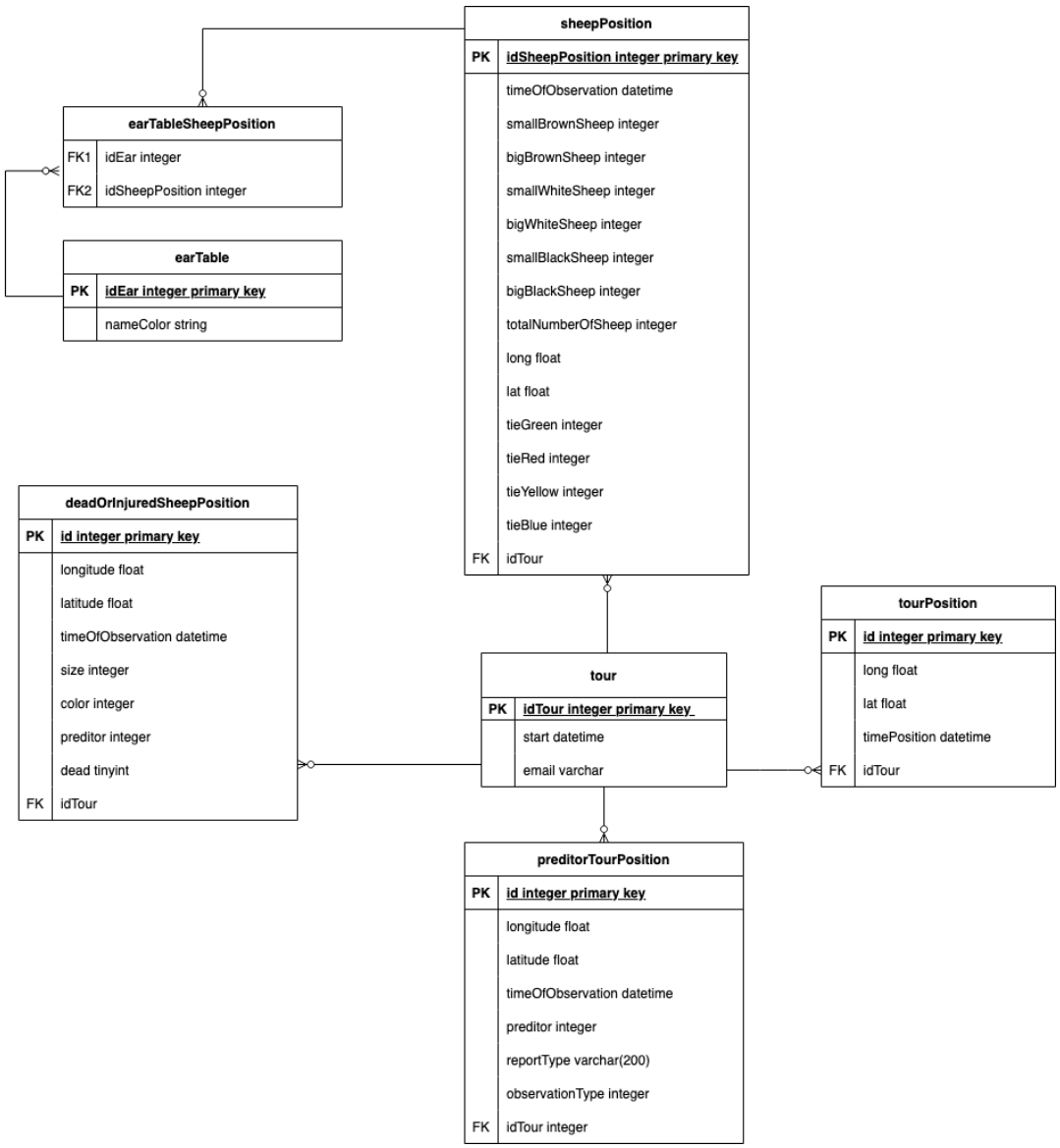
11.1 Database

11.1.1 Databasestruktur

Fra tabell 8.1 i kapittel 8.2.1 ble det beskrevet at serveren skal hente data fra en database, og i kapittel 9.2.3 beskrives det hvorfor det ble valgt Azure som skydatabase. For å kunne lage en database, er det nødvendig å forstå karakteristikken til dataene som brukes. ER-diagram består av entitet som er en klasse med personer, objekter, lokasjoner eller konsepter som kan bli brukt til å lagre data [75]. Klassene hjelper med å forstå relasjonene mellom dataene som er nødvendige for å sette opp databasen.

Fra figur 11.1 ser en ER-modellen som ble brukt til å lage databasen. Strukturen er basert på forprosjektet, men utvidet med noen tabeller for å tilrettelegge endringene som er spesifisert i kapittel 5.4. Det er syv tabeller, som alle tar utgangspunkt i *tour*-tabellen. Registreringen av en tur er knyttet til denne tabellen, og har en id, startpunkt og epostadresse som identifiserer hvem det er som opprettet turen. For å kartlegge hvor bonden går, er det laget en egen *tourPosition*-tabell som registrerer posisjonen til bonden hvert tiende sekund.

Når bonden observerer sauer på turen, blir de registrert i *sheepPosition*-tabellen. Posisjonen til sauene blir registrert på kartet der bonden observerer de, med lengde- og breddegrad. Her registreres det hvor mange søyer og lam som er brune, hvite og svarte. Hver sau har et bjelleslips som symboliserer hvor mange lam de har, som enten har grønn, rød, gul eller blå fargekode [12]. For å holde kontroll på hvem som er eiere av sauene i flokkene, kan brukeren legge til egne øremerkefarger, som blir registrert i *earTable*-tabellen. Ettersom en saueflokk kan inneholde sauer fra forskjellige eiere, ble det laget en mange-til-mange tabell, *earTableSheepPosition*.



Figur 11.1: ER-diagram for databasestruktur

I løpet av en tur kan det forekomme observasjoner av døde eller skadde sauer. Hendelsene registreres i en egen tabell, *deadOrInjuredSheepPosition*. Lokasjonen til søyene eller lammene blir registrert, sammen med hvilket rovdyr som har utført handlingen, hvis det er mulig å detektere. Dersom dyret er drept vil *dead*-kolonnen i *deadOrInjuredSheepPosition* være satt til 1, og ved skade vil kolonnen inneholde 0.

Dersom bonden ser et rovdyr i løpet av turen, registreres dette i *preditorTourPosition*-tabellen. Her er det mulig for å legge inn koordinatene til hvor rovdyret ble observert. Det skal være mulig å registrere jerv, gaupe, bjørn og ulv som rovdyr. Hvis bonden ikke klarer å se hvilke dyr det er, er det mulig å legge inn uidentifiserte rovdyr. Ettersom det er flere måter å oppdage et rovdyr på, kan bonden velge observasjonstype, som består av dyrespor, ekskrement, hår eller observasjon av selve rovdyret. Bonden velger om det skal registreres en rovdyrobservasjon eller om det er et dødt rovdyr.

11.1.2 Azure database

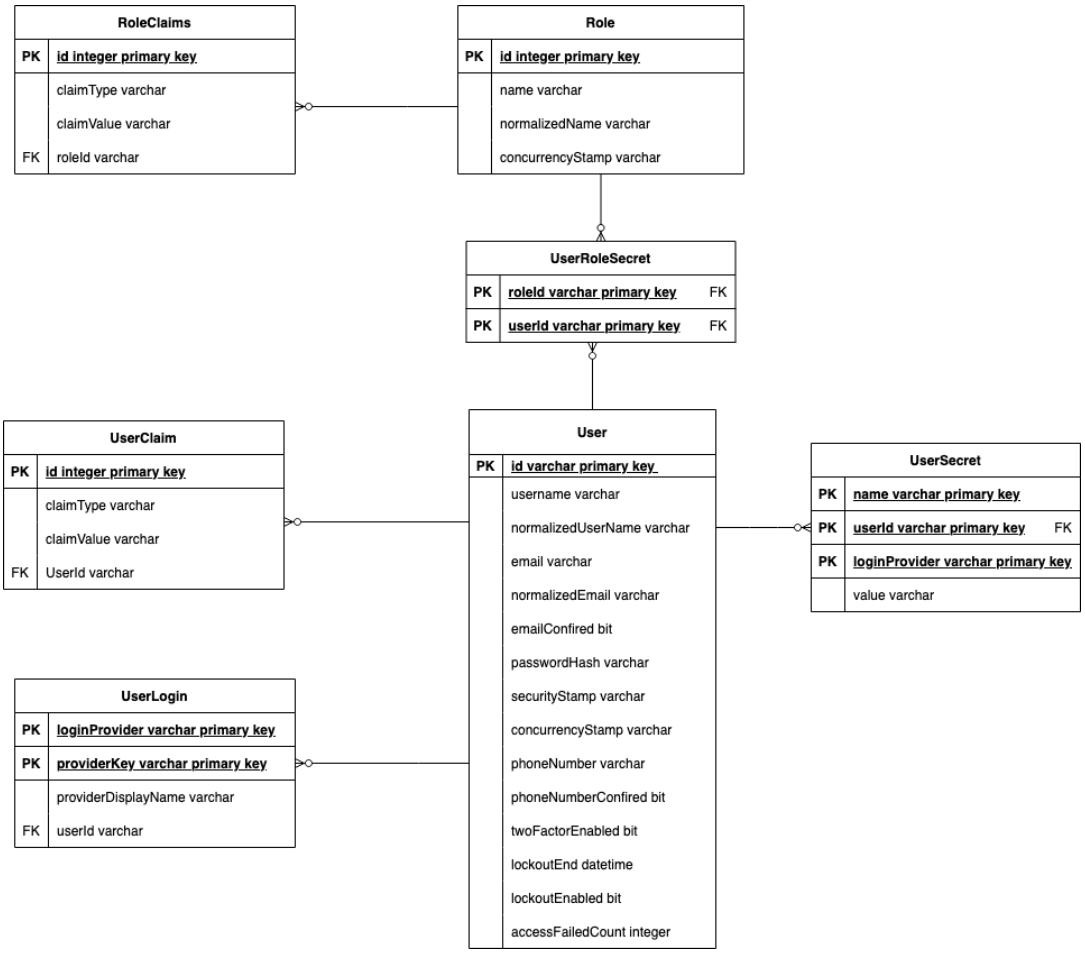
Etter ER-modellen ble laget, var neste steg å sette opp en SQL-database. Som beskrevet i kapittel 9.2.3, ble det brukt en Azure SQL-database. For å opprette databasen måtte en velge lokasjon og valget ble *Norway East*. Ettersom en ønsket lavest mulig responstid på databasen, ble lokasjonen nærmest NTNU valgt.

11.1.3 Entity Framework Core

For å sende spørringer til Azure SQL-databasen, måtte først databaseklasser lages ved hjelp av EFC. Databaseklassene inneholder de samme entitetene som er presentert i figur 11.1. For å koble relasjonene mellom databaseklassene sammen, ble det opprettet en egen fil, *SheepContext*. I denne filen ble alle klassene knyttet til *tour*-klassen, som er midtpunktet i figur 11.1. Det siste steget var å sette opp en kobling mellom serveren og Azure SQL-databasen. Dette ble gjort ved en tilkoblingsstreng som genereres av Azure [64]. Strengen inneholder blant annet brukernavn og passord til databasen.

11.2 ASP.NET Core Identity

For å logge inn, og autentisere brukere ble det brukt ASP.NET Core Identity, beskrevet i kapittel 9.2.2. For å holde oversikt over de registrerte brukerne, var det nødvendig med en brukerdatabase. En fordel med Identity er at biblioteket genererer en ferdig database-struktur, som en ser på figur 11.2.



Figur 11.2: ER-diagram for lagring av brukerinformasjon i database

I databasestrukturen sentrerer alle tabeller seg rundt *user*-tabellen. De viktigste entitetene på en bruker er brukernavn, epost og passord. Dette er den informasjonen en trenger for å logge inn på applikasjonen. Alle som skal bruke webapplikasjonen må opprette en bruker. *UserLogin*-tabellen har kontroll på om brukeren er logget inn eller ikke. *UserSecret*-tabellen inneholder en hemmelig streng som blir generert for å gjøre hver brukerinnlogging tilfeldig. *UserClaim*-tabellen inneholder informasjon om hvem den innloggede brukeren er, som eksempelvis kan være epostadressen til en bruker. I teorien kan en bruker ha flere roller, der hver rolle blir lagret i *Role*-tabellen. En rolle har også informasjon om den bestemte rollen, som blir lagret i *RoleClaims*. En rolle kan høre til flere brukere, og derfor ble det opprett en mange-til-mange relasjon som heter *UserRoleSecret*. I applikasjonen, *BeiteSpor*, er det derimot kun en rolle, der alle brukerne blir registrert som bonde. Det er muligheter for å legge til en administratortilgang i ettertid dersom det er nødvendig.

Ettersom webapplikasjonen krever innloggede brukere, er alle API-endepunktene for å hente data beskyttet bak autentisering. Brukeren må ha en gyldig *token* som blir tildelt

under innlogging, og denne sendes sammen med spørringene til autentisering i endepunktene. *Token* ble sendt fra brukergrensesnittet til de aktuelle API-endepunktene en ønsker å hente data fra. Autentisering på serveren ble gjort ved å sette *Authorize* på hvert endepunkt, som en ser på figur 11.3. *Authorize* sjekker om det er en gyldig *token* eller ikke. Dersom den ikke er gyldig vil serveren sende feilkoden 403 - forbudt.

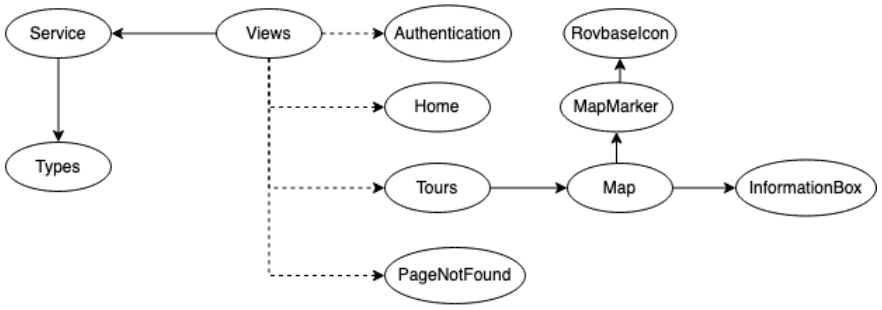
```
[Authorize]
[HttpGet("location")]
public async Task<ActionResult<IEnumerable<TourData>>> getTourLocations()
{
```

Figur 11.3: Autentisering på API-endepunkt

11.3 Struktur

11.3.1 Brukergrensesnitt

Som beskrevet i kapittel 9.1.1, ble det brukt React til å kode brukergrensesnittet. React er bygget opp av komponenter og ikke klasser, der hver komponent inneholder egne spesifikke oppgaver. Fra figur 11.4 ser en oppbyggingen til brukergrensesnittet. Figuren viser en overordnet oversikt, der hvert navn er en mappestruktur, og ikke en oversikt over alle komponentene. Ettersom en mappe kan inneholde flere titalls komponenter, blir det mer oversiktlig å vise mappene. På venstre side kan en se *Types*, som inneholder alle de ulike objekttypene som ble brukt i koden. *Service* inneholder spørringer som henter data fra serveren.



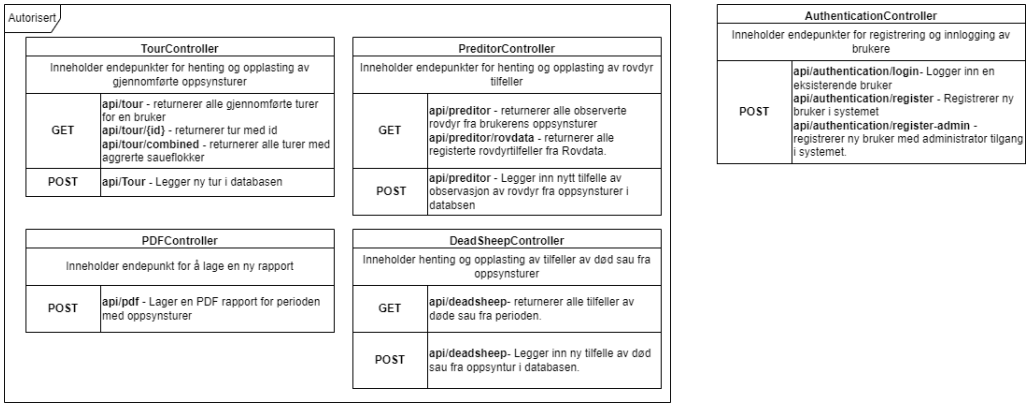
Figur 11.4: Overordnet struktur til brukergrensesnittet

Hovedmappen i brukergrensesnittet er *Views*, som igjen inneholder flere mapper. Fra figur 11.4 ser en at *Views* inneholder både *Authentication*, *Home*, *Tours* og *PageNotFound*. *Authentication* inneholder komponenter som tar for seg innlogging, registrering og utlogging av en bruker. *Home* viser fremsiden til applikasjonen, og eneste oppgaven er å sikre seg at brukeren ikke kommer videre før den har blitt logget inn ved hjelp av komponenter fra *Authentication*. Dersom en ikke er logget inn og går på en adresse en ikke har tilgang til, vil *PageNotFound* bli brukt, som sier at siden ikke ble funnet. *Tours* inneholder komponenter for hovedfunksjonene til webapplikasjonen, som er å velge turer i en meny,

oversikt over turer og vise sauer på kartet. *Tours* bruker videre komponenter fra *Map*, som inneholder all funksjonalitet for kartvisning. For å kunne vise markører på kartet, bruker *Map* komponenter fra *MapMarker*, som igjen henter ikoner til å vise i markørene fra *RovbaseIcon*. For å vise informasjon når en klikker på en markør, blir det hentet informasjon fra *InformationBox*.

11.3.2 Server

Serveren til applikasjonen har to funksjoner. Den første er å motta data fra gjennomførte tilsynsturer gjennom mobilapplikasjonen som ble laget i forprosjektet, som er FK1 i kravspesifikasjonen. Den andre funksjonen, FK2 i kravspesifikasjonene, er å hente data fra tidligere opplastede turer. Begge disse funksjonene er knyttet til en bruker av system og dermed har applikasjonen funksjoner for registrering og innlogging av brukere (FK5 og FK6).



Figur 11.5: Kart over endepunktet på serveren

Serveren fungerer som et API for henting og opplasting av data, og har endepunkter som tilfredsstiller kravspesifikasjonene FK1, FK2, FK5 og FK6. Kart over endepunktene i applikasjonen er vist i Figur 11.5. Hver av boksene i figuren representerer en kontroller med hver sitt ansvarsområde og tilhørende endepunkter i applikasjonen. *AuthenticationController* inneholder endepunktene som er knyttet til brukere i systemet som innbefatter registrering av ny bruker med endepunktet *api/authentication/register* og innlogging av eksisterende brukere med endepunktet *api/authentication/login*. Endepunktet for innlogging returnerer en JSON Web Tokens (JWT) token, som brukes i resten av systemet for å knytte henting og opplasting av data til en bruker. JWT token blir sendt i HTTP-headeren til endepunktene som markert innenfor autorisert i Figur 11.5.

Kontrollerne som er markert innenfor autorisert-boksen er basisfunksjonene i applikasjonen. Dette innebærer å laste opp og hente data, hente rovdyrdata og lage PDF-rapporter. *TourController* inneholder endepunkt for å laste opp en fullført tilsynstur gjennom *api/tour*(POST)-endepunktet. Videre kan en hente alle turer og en spesifikk tur på id gjennom de respektive endpunktene *api/tour*(GET) og *api/tour/id*(GET). I tillegg har det blitt laget et endepunkt som aggregerer nærliggende saueflokker i *api/tour/combined*(GET).

PreditorController og *DeadSheepController* er knyttet til rovdyrtilfeller. *PreditorController* har to GET-endepunkter, hvor rovdyrtilfeller oppdaget på tilsynsturer hentes gjennom

api/preditor endepunktet, mens rovdyrdata fra Rovbase hentes gjennom *api/preditor/rovdata* endepunktet. Ved observasjon av rovdyr på en tilsynstur brukes *api/preditor* (POST) som endepunkt for opplasting. *DeadSheepController* inneholder endpunkter for å hente og laste opp data av døde sauer fra rovdyrskade. Ved observasjon av rovviltskade på tilsynsturer brukes *api/deadsheep* (POST) til å laste opp, mens *api/deadsheep* (GET) brukes for å hente alle observerte hendelser til visning i brukergrensesnittet. Den siste kontrolleren inneholder et endepunkt, *api/pdf* (POST), for å lage en PDF-rapport fra tilsynsturene.

11.4 Norgeskart

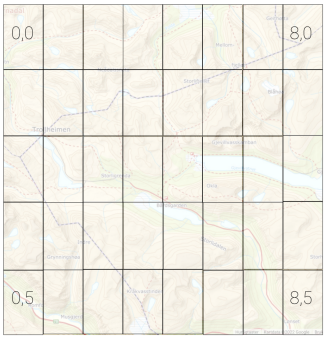
Det funksjonelle kravet FK11 i kapittel 8.2.1, spesifiserer at Norgeskart skal brukes som kartgrunnlag i *BeiteSpor*. Dette ble implementert ved hjelp av en kartmotor i bunnen, og deretter ble Norgeskart brukt som et kartlag over. Fra kapittel 9.1.2, ble det beskrevet hvorfor Google Maps ble valgt som motor. Ettersom det allerede finnes mange Google Maps pakker, lastet utviklerne ned pakken *GoogleMapReact*, fremfor å lage alt fra bunnen av. Når Google Maps skal regne ut en lokasjon i den virkelige verden til en lokasjon på webapplikasjon-kartet, må den oversette lengde- og breddegrad til en verdenskoordinat [76]. Dette ble gjort ved hjelp av Mercator projeksjon, som er standarden for navigering med kart. Mercator projeksjon er en sylinderprojeksjon, som benytter et sylinder som tangerer ekvator. Sylinderen brettes ut som et rektangel, der nord blir opp, og sør blir ned på kartet, uansett hvor en er i verden [77].

Når en legger Norgeskart over Google Maps, henter en mange små ruter fra Norgeskart og legger de sammen til et stort bilde som dekker hele skjermen. Norgeskart har en egen URL, der en kan sende inn x- og y-posisjon samt hvor mye en har zoomet inn på kartet. For å finne ut hvor mange ruter en trenger for å dekke over hele skjermen er en avhengig av å vite zoom-nivået, størrelsen på skjermen og hvilken lengde- og breddegrad en befinner seg på. Deretter måtte en regne ut topp- og bunnkoordinat for breddegrad, og venstre- og høyrekoordinat for lengdegrad. Dette ble gjort med følgende formler:[78]

$$Breddegrad: \frac{1 - \log(\tan(breddegrad * \frac{\pi}{180}) + \frac{1}{\cos breddegrad * \pi / 180}/\pi}{2 * 2^{zoom-level}} \tag{11.1}$$

$$Lengdegrad: \frac{lengdegrad * 180}{360 * 2^{zoom-level}} \tag{11.2}$$

For å regne ut hvor mange ruter en trenger i høyden bruker en formel 11.1, høyde = (breddegrad_topprute - breddegrad_bunnrute) + 1 [76]. Deretter bruker en formel 11.2, for å regne ut lengde = (lengdegrad_venstrerute - lengdegrad_høyrerute) + 1. Etter utregning av høyde og lengde, multipliserer en de i lag for å få totalt antall ruter. Resultatet av multiplikasjonen tilsier hvor mange ruter en må laste ned fra URL-spørringen til Norgeskart. På figur 11.6 kan en se et eksempel på rutenett på kartet. Rutene legges fra oppe til venstre og går ned mot høyre.



Figur 11.6: Fremstilling av rutekoordinat i Norgeskart

Fra figur 11.7 ser en to ulike kartlag, der figuren til venstre er Google Maps, mens figuren til høyre er Norgeskart. Norgeskart viser bedre linjer for naturområder, og gjør det lettere for bønder å se kontrurer i landskapet [79].

Figur 11.7: Kart med Google Maps og Norgeskart

11.5 Bonitet

I kapittel 11.4 blir det beskrevet hvordan en legger kartlag over Google Maps, ved bruk av Norgeskart. På *BeiteSpor* skal brukerne ha mulighet til å velge visning av bonitetslag på kartet, som det funksjonelle kravet FK12 tilsier. Bonitet er et begrep som blir brukt for å vise vekstpotensiale for trær, og blir målt fra lav til høy bonitet [80]. Vekstpotensiale avhenger av ikke-levende faktorer i skogen, slik som temperatur, næringstilgang og fuktighetsforhold. Høy bonitet på et område vil si at det befinner seg bergarter med viktige næringsinnhold, slik som kalkrike bergarter. NIBIO tilbyr et API med kartdata om bonitet, som viser flere forskjellige klasser av bonitet. Figur 11.8 viser alle klassene som er tilgjengelig fra NIBIO.

Figur 11.8: Oversikt over bonitetsklasser [81]

Web Map Service (WMS) er en av Open Geospatial Consortium sine standarder som tilbyr et HTTP-grensesnitt hvor en spør om geografiske kartbilder fra en eller flere lokasjonsdatabaser [82]. De geografiske lagene til et område blir definert i en WMS-forespørsel. Responsen består av et eller flere geografiske bilder, som kan være JPEG eller PNG. Det er mulig å sende inn en parameter i forespørselen som spesifiserer at bildene skal ha usynlig bakgrunn, slik at flere lag fra forskjellige servere kan bli lagt sammen til et stort bilde. Det vil si at WMS tilbyr skalerbare kartlag som en kan bruke til å legge over Google Maps.

For å kunne hente størrelsen på kartbilde til bonitetslaget som hentes i WMS-forespørsel, må Google Maps sine x- og y-koordinater beregnes om til bredde- og lengdegrad ved bruk av zoom-verdien i kartet. WMS tar inn to koordinatpunkter (x_min, y_min) og (x_max, y_max). Som forklart i 11.4 bruker Google Maps Mercator projeksjon, som gjør om hele kartet til et firkantet kart, der grenseverdiene er [-π * 6378237, π * 6378137]. For å finne disse parameterne brukte vi følgende formeler:

$$rutestørrelse = \frac{\pi * 6378137 * 2}{2^z} \tag{11.3}$$

$$x_min = -\pi * 6378237 + x * rutestørrelse \tag{11.4}$$

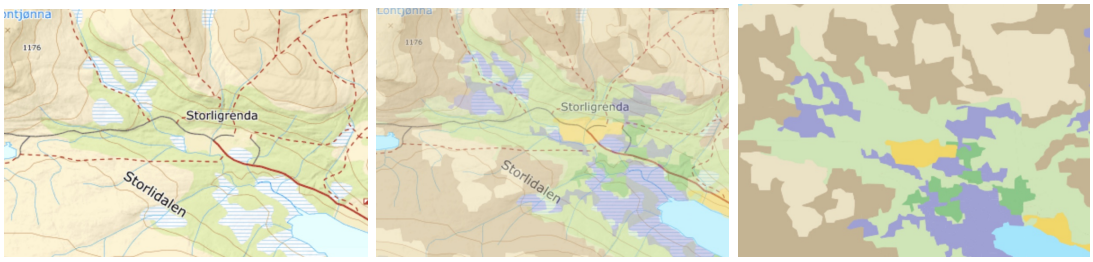
$$x_max = -\pi * 6378237 + (x + 1) * rutestørrelse \tag{11.5}$$

$$y_min = \pi * 6378237 - (y + 1) * rutestørrelse \tag{11.6}$$

$$y_max = \pi * 6378237 - y * rutestørrelse \tag{11.7}$$

Hvor x og y er koordinater, og z er nåværende zoom i kartet.

Etter omregninger ved bruk av formlene 11.3-11.7, blir parameterne sendt i en WMS-forespørsel. Resultatet av forespørselen er et rutenett med kartbilder som er forklart i kapittel 11.4. På figur 11.9 kan en se tre ulike figurer. Figuren til venstre er kun Norgeskart, figuren i midten er Norgeskart med bonitet som har en synlighet på 50%. Med en synlighet på 50 % blir konturer fra Norgeskart, som stedsnavn og høydekurver synlig sammen med bonitetskartlaget. Figuren til høyre viser 100% bonitetslag, og en ser tydelig de ulike bonitetsfargene, som er beskrevet i Figur 11.8. Selv om bildene kun viser bonitetslag med 0%, 50% og 100%, har brukeren mulighet til å velge synlighet fra 0 til 100 %.



Figur 11.9: Kart med Noregskart og bonitetskart

11.6 Identifisering av saueflokker

Fra de funksjonelle kravene i tabell 8.1 står det på punkt FK10 at en skal kunne identifisere saueflokker fra tur til tur. For å kunne identifisere saueflokker på neste tur, må en først definere hva en saueflokk er. Når bonden går tur med mobilapplikasjonen, beskrevet i kapittel 5, registreres det sauer etterhvert som bonden observerer de. Alle sauene som blir registrert på den samme lokasjonen, tilsvarer en saueflokk. Dersom saueflokkene er mindre enn 500 meter fra hverandre, skal flokkene bli aggregert til en større flokk. Avstanden mellom to saueflokker kan beregnes ved bruk av Haversine-formelen, som finner distansen mellom to koordinater [83].

$$
\begin{aligned}
a &= \sin(\Delta\phi/2) + \cos(\phi_1) \cdot \cos(\phi_2) \cdot \sin^2(\Delta\lambda/2) \\
c &= 2 \cdot atan2(\sqrt{a}, \sqrt{1-a}) \\
d &= R \cdot c
\end{aligned}
\tag{11.8}
$$

Hvor ϕ er breddegrad, λ er lengdegrad, og R er radiusen til jorda.

For å finne en aggregert saueflokk, starter en med å se på en tilfeldig saueflokk. Deretter sjekker en alle de andre saueflokkene på den samme turen, og finner distansen ved bruk av formel 11.8. Alle flokkene som har en avstand på mindre enn 500 meter blir lagt i den samme tabellen. Dette blir gjort for alle saueflokkene. For å få bedre oversikt over de aggregerte flokkene, regner en ut antall svarte, grå og hvite sauer, samt totalt antall. Plasseringen i kartet til den aggregerte saueflokken baseres på midtpunktet til koordinatene til de distinkte flokkene. For å beregne midtpunktet ble det brukt en funksjon i Google Maps API, som tar inn en liste koordinater og returnerer midtpunktet [84].

Etter en har aggregert saueflokkene fra en tur, er det ønskelig å finne ut hvor de aggregerte saueflokkene mest sannsynlig befinner seg på neste tur. Slik at en kan se et mønster mellom hver tur. Dette utført med lik framgangsmåte som hvordan en regnet ut de nærmeste saueflokkene fra en tur. En starter med en tilfeldig aggregert flokk på den nåværende turen, og regner ut distansen til alle andre flokker med formel 11.8. Forskjellen er at en kun tar den flokken som er nærmest, og ikke flere flokker. Metoden ble utarbeidet sammen med veileder og domeneekspert, Hvasshovd, som begrunnet valget med at sauer beveger seg ofte i det samme området.

11.7 Årsrapport

Fra kravspesifikasjonen i kapittel 8.2.1 står det på punkt FK9 at det skal være mulig å generere en PDF-årsrapport. Rapporten skal inneholde dato fra alle turene, og hvor mange hvite, brune og svarte sauer som ble observert på hver tur. I tillegg skal det være mulig å vise sauer som er skadet eller døde, og om det er observert noen rovdyr.

Årsrapporten ble generert på serveren ved hjelp av biblioteket *WkHtmlToPdf-DotNet*. Biblioteket konverter HTML med CSS til PDF-format [85]. Figur 11.10 viser et eksempel på en årsrapport. Her ser en syv kolonner som viser hvilke sauer og rovdyr som er observert på hvilken tur. Hver rad representerer hver tur, med tilhørende dato. Selv om det heter årsrapport, er det mulig å se rapport fra et bestemt tidsintervall.

Rapport fra 5/17/2019 - 6/7/2019

Dato	Hvit sau	Brun sau	Svart sau	Død sau	Skadet sau	Observert rovdyr
5/17/2019	15	20	24		1 hvit lam	1 jerv
5/24/2019	14	20	24	1 hvit lam		1 gaupe
5/31/2019	14	20	24			1 bjørn
6/7/2019	14	20	23	1 svart sau		2 ulv

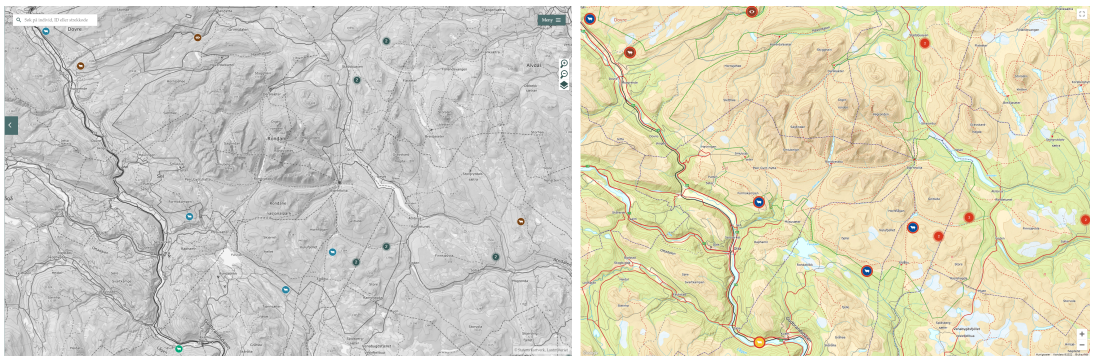
Figur 11.10: PDF-årsrapport over sau og rovdyr

11.8 Rovdyrskade

11.8.1 Rovbase

I kravspesifikasjonene fra kapittel 8.2 ble det satt opp kravet FK3, som spesifiserer at *BeiteSpor* skal hente data om rovdyr fra Rovbase. Rovbase er et verktøy som forvalter de store rovdyrene i Norge, Sverige og Finland [86]. Informasjonen er basert på opplysninger fra lokale kilder, i tillegg til organiserte registreringer gjennomført av feltpersonell. Integrasjon mot Rovbase i webapplikasjonen kan gi brukeren nyttig informasjon om hvilke tiltak som bør settes inn, dersom det er observert rovdyr i nærheten av bondens sauer.

I figur 11.11 sammenlignes Rovbase sin nettløsning, (rovbase.no), og *BeiteSpor* sin integrasjon mot rovbase. Skjermebildene viser registreringer av rovdyr fra samme periode og samme sted. Løsningene er i stor grad like, med noen små forskjeller i visning av ikone, der *BeiteSpor* har en rød ring rundt alle rovdyrregistreringer. Hensikten med å hente elementer fra Rovbase, er skape et gjenkjennelig design for brukeren. Det er sannsynlig at bønder, som er brukerbasen til webapplikasjonen, har interagert med Rovbase tidligere.



Figur 11.11: Skjermbilde med sammenligning av Rovbase sin nettløsning og integrasjon med Rovbase i våres løsning

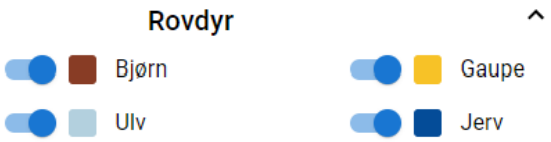
Integrasjon

Rovbase har et API for spørringer av data om rovdyr. Adressen til APIet er *https://www.rovbase.no/api/Feature*, og hentes gjennom en POST-spørring. Rovbase har lagt inn Cross-Origin Resource Sharing (CORS) regler i API-et, som vil si at spørringer som gjøres mot API-et må komme fra en nettleser med samme domene, altså rovbase.no. Måten vi håndterte dette på var å lage et eget endepunkt i server som spør om data fra Rovbase API-et, siden mekanismen kun er låst mot nettlesere.

API-et fra rovbase tar i mot et objekt som inneholder parametre som bestemmer hvordan en vil filtrere dataene en spør etter. Det er kun noen av parametrene som ble brukt i spørringene, resten står tomme. Parametrene som brukes er:

- Carnivore: Tar en liste med Id-er på rovdyr. Mulighetene er ulv (1), bjørn (2), gaupe (3) og jerv (4).
- CarnivoreDamage: Tar en liste med Id-er på skadede dyr. Mulighetene er sau (1), rein (2), hund (3), geit (4) og storfe (5).
- Observation: Tar en liste med Id-er på type observasjon av rovdyr. Mulighetene vi har lagt støtte for er synsobservasjon (2), spor (3), ekskrement (4) og hår (8).
- FromDate: Fra datoen rovdyrhendelser har blitt registrert.
- ToDate: Til datoen rovdyrhendelser har blitt registrert.

I webapplikasjonen er fra- og til datoene de eneste variable parametrene i spørringen. Hver gang perioden for visningen endres, blir det utført en ny spørring til Rovbase API-et. Alle valgmulighetene i de andre parametrene er alltid valgt. Fra en spørring kommer det en respons med en liste over alle rovdyrregistreringene i den valgte perioden. Det er fire hovedtyper av registreringer, rovvilt observasjoner, rovvilt skade, døde rovdyr og DNA-prøve.

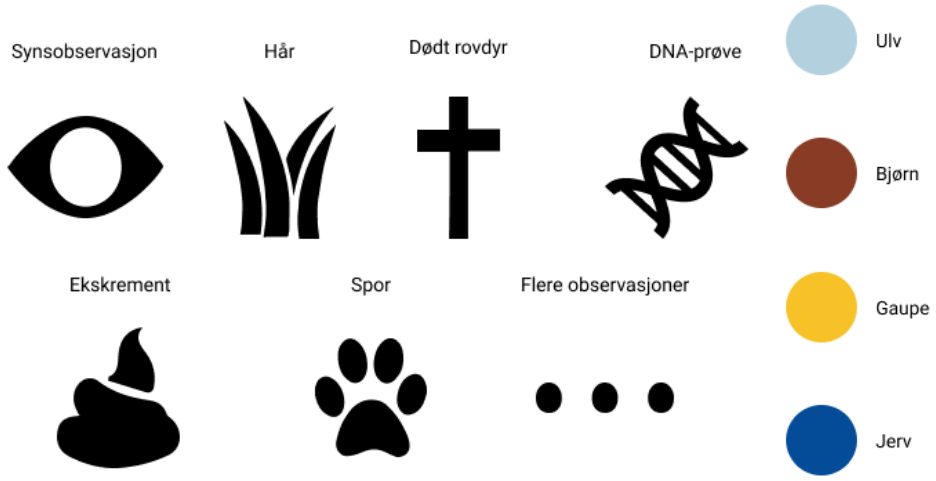


Figur 11.12: Skjermbilde av filtrering av rovdyrhendelser i webapplikasjonen

Alle rovdyrhendelsene blir hentet for en gitt periode, og blir filtrert i brukergrensesnittet basert på hva brukeren har valgt. Figur 11.12 viser skjermbilde av den implementerte rovdyr-filtreringskomponenten. De fire rovdyrene kan aktiveres og deaktiveres i kartet ved å slå bryterene av og på.

Ikoner og Fargekoder

Ikonene som brukes i markørene på kartet er hentet fra Rovbase. Fargekodene er ikke de samme, ettersom webapplikasjonen skulle ta hensyn til personer med fargeblindhet. Rovbase bruker forskjellige fargekoder og ikoner basert på hvilken type registrering som er blitt gjort.



Figur 11.13: Ikoner brukt for markører i webapplikasjonen, som er hentet fra Rovbase

Figur 11.13 har en visning av alle ikonene som er implementert i webapplikasjonen. Hvis registreringen er av typen rovviltobservasjon, vil markøren ha et ikon som avhenger av hvilken observasjon som er gjort. Hver rovviltobservasjon kan bestå av en eller flere observasjoner. Ikonene er synsobservasjon, hår, ekskrement, spor eller tre dotter for flere observasjoner. I tillegg til ikoner for rovviltobservasjoner viser applikasjonen et kors for dødt dyr og et DNA-ikon for gjennomført DNA-prøve.

Rovdata operer med fargekoder ut fra hvilken type rovdyr registreringen gjelder. *BeiteSpor* opererer på lik måte, der hvert rovdyr har en egen tilknyttet farge, men med farger som tar hensyn til fargeblindhet. Til høyre i Figur 11.13 er de fire rovdyrtypene listet opp med tilhørende fargekode. Markørene for ulv, bjørn, gaupe og jerv har henholdsvis lyseblå, brun, gul og blå farge.

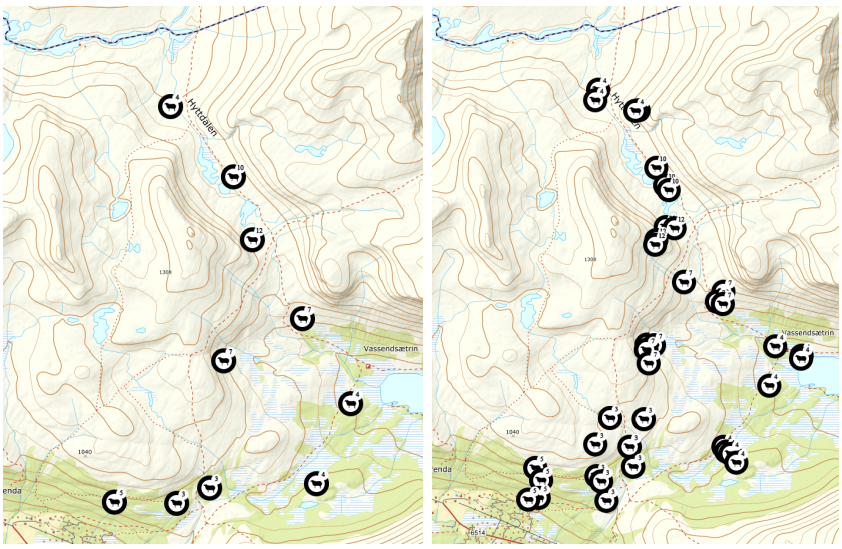
11.8.2 Registrert hendelse fra mobilapplikasjon

Krav FK15 fra kravspesifikasjonen i kapittel 8.2.1, går ut på at døde sauer skal bli vist i kartet. Ettersom det var mulig å registrere døde sauer på mobilapplikasjonen, og hente døde sauer fra Rovbase, ble det laget en felles markør. Fra figur 11.14 kan en se fire markører med ulik farge. Hver markør har en rød sirkel helt ytterst, som skal representere at dette er et dødt dyr. Videre har hver markør en bestemt farge, som skal vise hvilket rovdyr sauen ble drept av, der fargene er beskrevet i figur 11.13.

Figur 11.14: Markør for død sau på kartet

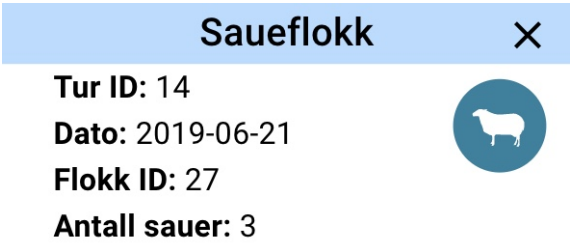
11.9 Saueflokk

En av de viktigste funksjonene til *BeiteSpor* er å vise saueflokker registrert på mobilapplikasjonen. Punkt FK14 i kapittel 8.2.1, sier at en skal vise saueflokker på kartet. Det står derimot ikke spesifisert noe om hvordan flokkene skal bli representert. Dette ble løst ved å bruke runde sirkler som inneholdt et ikon med en sau. For å videre symbolisere hvor mange sauer som befant seg i den bestemte saueflokken, ble det lagt til en liten sirkel med et tall. Fra figur 11.15 kan en se eksempler på saueflokker på kartet. Grunnen til at vi valgte å symbolisere saueflokkene på denne måten, var for å skille de i fra ikonene som kom fra rovbase, beskrevet i kapittel 11.8.1.



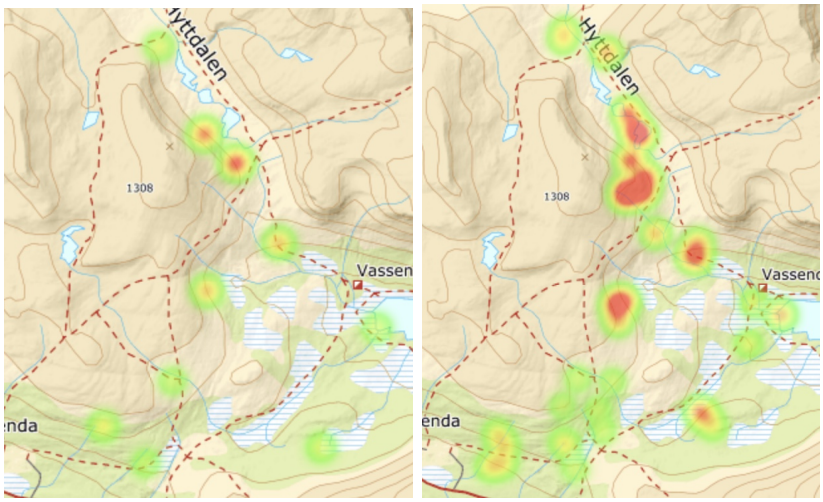
Figur 11.15: Oversikt over saueflokker på beite med ukes- og månedsvisning

Videre fra tabell 8.2 i kapittel 8.2.1 står det på punkt FK16 at det skal være mulig å klikke på en spesifikk flokk, for å kunne se mer informasjon om flokken. Skjermbilde av informasjonsboksen er representert i figur 11.16, som inneholder tur-id, dato fra når flokken ble registrert, flokk-id og til slutt hvor mange sauer det er i flokken. I løpet av utviklingsprosessen kom det frem at informasjonsboksen ikke trengte å inneholde hvilke farger flokken bestod av, i motsetning til skissene laget i kapittel 10.3.1.



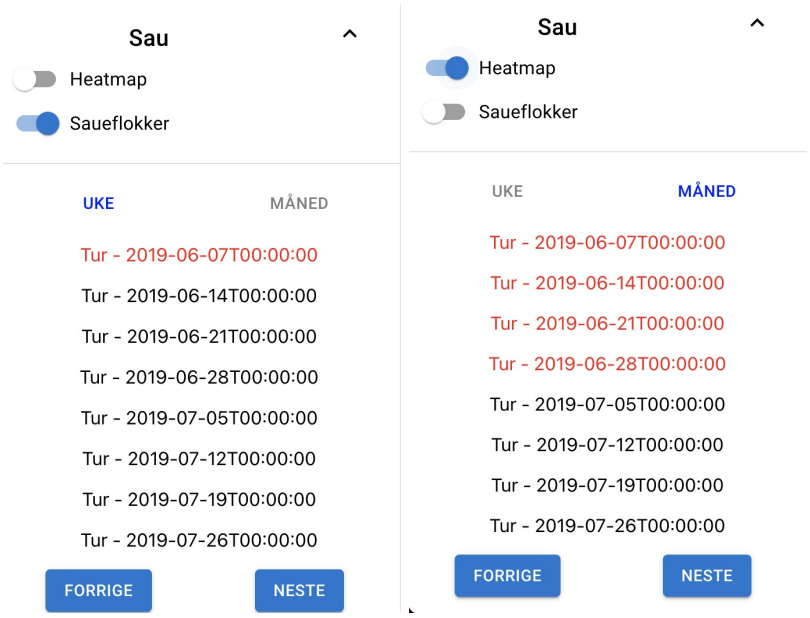
Figur 11.16: Informasjon om en bestemt saueflokk

Heatmap ble laget ved hjelp av Google Maps sin innebygde funksjon, *HeatmapLayer* [87]. For å lage heatmap, sendes det inn en liste med koordinater med tilhørende verdi, som tilsvarer posisjonen til saueflokken og antall sauer i flokken. Heatmap har en skala med mye aktivitet, fra rødt til grønt, der rødt er mye aktivitet og grønt er lite aktivitet. Høy aktivitet i kartet vil si at det er en flokk med mange sauer eller flere flokker som ligger i nærheten av hverandre. Figur 11.17 viser heatmap med ukesvisning til venstre og månedsvisning til høyre. Månedvisning har flere røde områder enn i ukevisning, som er naturlig når det i månedsvisning inneholder flere turer. Månedsvisning inneholder flere turer, som blir gjenspeilet i heatmapet med flere røde områder.



Figur 11.17: Heatmap med uke- og månedsvisning for saueflokker i *BeiteSpor*

Fra tabell 8.2 står det i punkt FK20 at det skal være mulig å velge mellom saueflokk for hver uke og måned. Dette ble gjort ved å legge alle turene i menyen. Fra figur 11.18 kan en se to figurer. Figuren til venstre viser ukevisning, der den aktive turen er markert med rødt. På venstre bilde i figur 11.15 og 11.17 ser en få datapunkter, ettersom en kun viser punkter fra en uke. På høyre bilde på figur 11.18 kan en se måned-visning. Figuren viser alle de aktive ukene i den valgte måned med rød farge. Videre kan en se på høyre bilde på figur 11.15 og 11.17 at det er mye aktivitet og mange datapunkter.



Figur 11.18: Oversikt over alle turene tilgjengelig innenfor en valgt tidsperiode

11.10 Oppsummering

Kapittelet tok for seg hvordan databasemodellen til *BeiteSpor* ble bygd opp, og hvordan databasen ble knyttet sammen med serveren. Det ble også automatisk opprettet en database ved hjelp av Identity-rammeverket, som håndterte brukerene til webapplikasjonen. Videre ble mappestrukturen til brukergrensesnittet, og alle endepunktene til serveren forklart. Implementasjon av kartlagene, Norgeskart og bonitet, ble gjennomgått, der en gikk gjennom hvordan en hentet rutene til kartene, og hvordan dette ble lagt inn i løsningen. Videre ble det beskrevet hvordan saueflokkene på kartet ble laget, og hvilke visninger som var tilgjengelig av saueflokkene. Basert på tidsperioden og hvilke turer som var valgt, kan det bli generert en årsrapport som tilfredsstiller kravene fra myndighetene. Det ble også beskrevet hvordan rovdyrene fra Rovbase ble implementert i kartløsningen, samt beskrivelse av tilhørende ikoner.

Del IV

Utførelse og resultat

I denne delen blir valg av metode for evaluering av *BeiteSpor*, gjennomføring av brukertestene og resultater fra brukertestene presentert.

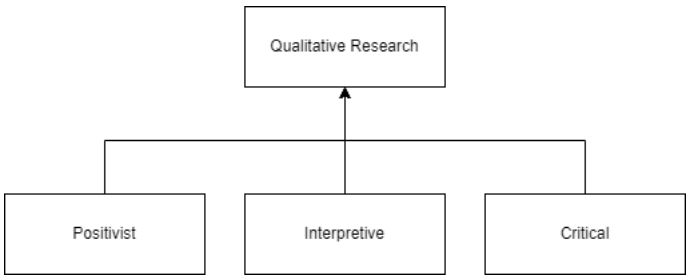
Kapittel 12

Brukertest

I kapittel 2.2 ble problemstillingen med tilhørende forskningsspørsmål presentert. Målet med masteroppgaven er å besvare forskningspørsmålene for å kunne finne ut om problemstillingen tilfredstilles. Problemstillingen er å lage en løsning med fremstilling av innsamlede data som kan gi bonden et bedre bilde på hvordan sauene beveger seg.

12.1 Kvalitativ analyse

Evalueringsgrunnlaget på det som har blitt implementert i masteroppgaven er gjennom en kvalitativ analyse basert på brukertester av personer i målgruppen. Bakgrunn for valg av type kvalitativ analyse er gjort basert på artikkelen til Myers og Avison [88] som tar for seg valgene av metode i en kvalitativ analyse av et informasjonsystem.



Figur 12.1: Myers og Avison [88] epistemologier for kvalitativ analyse

I kvalitativ analyse er det tre underliggende epistemologier som presentert i figur 12.1. De tre hovedepistemologiene er positivist, fortolkende (*interpretive*) og kritisk. Det er en forskjell mellom hvilken tilnærming som gjøres i kvalitative analyser. I våres oppgave er positivist og fortolkende mest relevant, fordi kritisk forskning fokuserer mer på motsetninger og konflikter i samfunnet. Positivist er en tilnærming hvor det antas at det er en objektiv oppfatning av realiteten, som er målbar og uavhengig av testpersonen [88]. Tilnærmingen brukes ofte i bevis eller motbevis av en teori. Den andre tilnæringen er fortolkende, som antar at testpersonen har en subjektiv oppfattelse av realiteten, og hver oppfattelse må analyseres hver for seg [88].

For den kvalitative analysen som foretas under masteroppgaven vil en fortolkende tilnærming være den mest hensiktsmessige. Forskjellen på tekniske kunnskaper og arbeidsmeto-

dene på bøndene som er i målgruppen, gjør at de vil ha et distinkt syn på hva som gjør jobben deres enklere. Neste steg i Myers og Avison [88] sin artikkel er valg av analysemetode. Den mest vanlige analysemetoden for datasystemer er *Case Study*, som handler om å "undersøke samtidsfenomener innenfor dens virkelige kontekst". Det vil si å lage en reell kontekst rundt brukertesten, som i tilfelle for masteroppgaven er å sette testpersonen i en kontekst der de bruker applikasjonen reelt.

12.2 Gjennomføring av brukertester

Den kvalitative analysen ble gjennomført ved brukertesting av bønder som er i målgruppen for masteroppgaven. Det var essensielt å velge testpersoner som passet målgruppen, ettersom *BeiteSpor* er laget for bønder. Bøndene har domenekunnskap om sau på utmarksbeite, som kan føre til at en ikke får testet bruksområde eller viktigheten av webapplikasjonen på personer utenfor målgruppen [89]. I tillegg får en testet om løsningen er teknisk vanskelig for brukerne, eller om den er tilpasset bøndens tekniske kompetanse. Brukertestene ble gjennomført som en kombinasjon av observasjoner og spørsmål knyttet til testpersonenes oppgaver og de forskjellige delene av applikasjonen. Testpersonene vil bli oppgitt som anonyme gjennom masteroppgaven, og resultatene fra disse testene ga grunnlaget for resultatet, diskusjonen og konklusjonen.

12.2.1 Før gjennomføring av brukertester

Før vi startet med brukertestene, ble det satt opp en plan for hvordan testene skulle bli gjennomført. Målet med brukertestene var å identifisere problemer med løsningen, og observere hvordan testpersonene interagerte med *BeiteSpor* [90]. Det ble valgt fem bønder som testpersoner, som ifølge Nielsen [91] er det som trengs for å avdekke mer enn 80% av alle feilene i et system.

Utviklerne hadde to ulike roller, den ene var leder og den andre var observatør. Lederen kommuniserte med testpersonene og forklarte hva oppgaven gikk ut på, etterfulgt av å stille spørsmål [90]. Observatørens rolle var å observere hva testpersonene sa underveis, og å notere ned hva de gjorde på brukertestene. Det varierte hvilke roller utviklerne hadde i de fem testene.

Hver brukertest var basert på at det eksisterte data som skulle tilsvare flere reelle tilsynsturer. Utviklerne genererte data for fire måneder, slik at alle testpersonene hadde likt datagrunnlag, og utgangspunkt før de startet med brukertesten. Videre var det ikke mulig å bruke systemet uten å ha en bruker. Ettersom målet med brukertestene ikke var å teste om bøndene klarte å opprette ny bruker, ble det forhåndslaget en bruker med tilhørende tilsynsturer.

testpersonen måtte tenke høyt under hele brukertesten slik at observatøren kunne skrive ned hva personen tenkte og hvorfor de utførte handlingene.

Før en startet å fortelle om bakgrunn for prosjektet og scenarioet til testpersonen, ble det gitt generell informasjon om hva en brukertest innebærer. Det ble sagt at det ikke var personen som ble testet, men systemet [92]. Testpersonen måtte tenke høyt under

hele brukertesten slik at observatøren kunne skrive ned hva personen tenkte og hvorfor de utførte handlingene [89]. Deretter ble testpersonen spurt om vedkommede kunne fortelle om bakgrunnen sin som bonde, slik som: hvor mange dyr en hadde, hvor lenge en hadde vært bonde og hvor ofte en gikk tilsynsturer. Leder for brukertestene startet med å lese opp bakgrunn for oppgaven til testpersonen [90], slik at de ble satt inn i det miljøet som var ønsket. Bakgrunnen var følgende:

> "I forprosjektet til masteroppgaven høsten 2021 ble det laget en mobilapplikasjon, som skal være et verktøy for bonden når en er ute på tilsynstur av sauer. I mobilapplikasjonen kan bonden registrere forskjellig karakteristikk ved saueflokker, som posisjon, antall sauer i forskjellige farger og størrelser, farge til bjelleslips og øremerker. Etter turen er ferdig, kan turen lastes opp i en skytjeneste som samler alle turene. I masteroppgaven har vi laget webapplikasjonen *BeiteSpor*, som samler inn dataene fra tilsynsturene med tilhørende visning i kart. Dataene som ligger inne i applikasjonen fra tilsynsturer er generert av oss, og er ikke reelle data."

Etter at bakgrunnen var opplest, passet en på at testpersonene forstod alt som var blitt presentert frem til nå. Deretter fikk de opplest et scenario som de skulle ta utgangspunkt i når de skulle gjennomføre brukertesten.

> "Over en lengre periode har du vært ute på tilsynsturer som har blitt lastet opp til webapplikasjonen. Du skal nå inn i den nye webapplikasjonen for å se data fra de gjennomførte tilsynsturene."

12.2.2 Under brukertesten

Under brukertesten leste testleder opp oppgaver som testpersonene skulle gjennomføre. Dette var ofte en liste med oppgaver. Det var viktig at testleder ikke ga hint eller veiledet testpersonen. Her skulle en observere hva brukeren gjorde. Dersom brukeren stod fast, kunne en prøve å omformulere spørsmålet slik at testpersonen kom på rett vei [90]. Etter hver oppgave kom det spørsmål som handlet om de oppgavene som brukeren hadde gjennomført, slik at de hadde oppgavene i minnet. Observatøren skrev ned alt som ble sagt og klikket på.

Oppgave 1 (Velg periode og rovdyr)

1. Velg en ny periode fra 12.juni til 1.august
2. Filtrer rovdyr, slik at du kun viser jerv i kartet
3. Finn ekstra informasjon om en hendelse av jerv i kartet
4. Velg i tillegg bjørn og ulv for å vise i kartet

Spørsmål 1

1. Kommer det tydelig frem hvilke rovdyr som hører til hver markør? Hva kunne vært

annerledes?
2. Er det tilhørende ikonet inni markøren forståelig?

Oppgave 2 (Oversikt over tilsynsturer)

1. Finn oversikten over alle tilsynsturer du har gått i den valgte perioden
2. Finn ekstra informasjon om en saueflokk

Spørsmål 2

1. Kommer det tydelig frem hva tallet på hver saueflokk betyr?
2. Savnet du noe informasjon når du klikket på en saueflokk?

Oppgave 3 (Måned- og ukevisning med heatmap og saueflokk)

1. Prøv å se på bevegelsene til sauene fra uke til uke i den valgte perioden
2. Gjør det samme igjen, men bruk nå heatmap som visning
3. Prøv å se på bevegelsen til sauene fra måned til måned i den valgte perioden, med å først bruke saueflokker og deretter heatmap

Spørsmål 3

1. Hvilke av de to visningene, heatmap eller saueflokk, syns du var best?
2. Var det noen forskjell på hvilken visning som var best på ukes- eller månedsvisning?
3. Kan denne funksjonaliteten hjelpe å holde saueflokkene borte fra områder med rovdyr?

Oppgave 4 (Bonitet og PDF)

1. Sett bonitet på kartet, med en synlighet på 0.7
2. Last ned en årsrapprt som PDF fra den valgte perioden

Spørsmål 4

1. Hvilken type sauerapportering gjør du, og hva skal med i rapporten?
2. Hjelper bonitetslagene deg med å forstå hvorfor sauene beveger seg slik de gjør?
   (a) Kommer det tydelig frem hva de ulike bonitetsfargene betyr?

12.2.3 Etter brukertesten

Etter alle oppgavene var gjennomført var det lagt opp til oppsummeringsspørsmål av brukertesten, som var følgende:

1. Hva bruker du per dags dato for å holde kontroll på sauene dine?
   (a) Bruker du en digital løsning?
2. Har du registrert sauer manuelt tidligere?
   (a) Registrerer du posisjonen til hver sau?
   (b) Registreres antall sauer, slips og øremerker?
   (c) Bruker du denne informasjonen til noe senere? Eventuelt hva?
   (d) Dersom du kunne sett alle dataene visuelt på en nettside i etterkant, hadde det vært til nytte?

3. Tror du denne løsningen som du har prøvd ut nå kan hjelpe deg til bedre å forstå hvordan saueflokkene beveger seg?
4. Var det noe som manglet for å kunne se bevegelsene til sauene?
   (a) Er det noe tilleggsinformasjon om saueflokkene eller en annen type visning som kunne forbedret løsningen?
5. Kan løsningen gjøre jobben din som sauebonde enklere?

12.2.4 Digital gjennomføring

På grunn av pandemien, COVID-19, og at flere av bøndene ikke befant seg i Trondheim, ble brukertestene gjennomført digitalt. Det finnes flere typer digital testing, der en kan velge å sende alle oppgavene til testpersonene og vente på at de gjennomfører oppgavene alene uten å bli observert [89]. Deretter sender de inn resultatet. Et annet alternativ er å gjennomføre brukertesten som om det hadde vært i virkeligheten. Der en snakker sammen på en digital plattform, og testpersonene deler skjermen sin, slik at det er mulig å se hva de klikker på. I tillegg kan det være en fordel å bruke kamera, slik at en kan se personene en snakker med [93]. Vi valgte å gå for alternativ nummer to, der vi brukte Microsoft Teams som vår digitale plattform, og testpersonene kunne dele skjermen sin. Ved bruk av Microsoft Teams blir brukertesten mer fysisk, og en får en følelse av å være i samme rom som testpersonen.

12.2.5 Pilottest

For å passe på at brukertesten inneholdt alt som var nødvendig, og at spørsmålene var godt formulerte og utvetydige, ble det gjennomført en pilottest. Brukertesten er designet av utviklingsteamet, og slik som det er med alt annet som er designet, må det testes før det brukes. En vanlig strategi er å utføre en pilottest på noen som enten er med i teamet, eller som har kjennskap til teamet eller oppgaven [90]. Det ble derfor utført en digital pilottest på veileder Hvasshovd, som testet at alt fungerte som ønsket, før brukertestene på bøndene begynte.

12.2.6 Oppsummering

Evalueringsmetoden for implementasjonen til *BeiteSpor* ble gjort ved en kvalitativ analyse. Fremgangen i kvalitative analysen ble gjort basert på artikkelen til Myers og Avison [88], der det ble valgt en fortolkende tilnæring med *Case Study* som analysemetode. Den kvalitative analysen ble gjort gjennom observasjoner og spørsmål i brukertestene av *BeiteSpor*. Brukertestene skulle dekke brukevennligheten for funksjonalitetene i applikasjonen, samt subjektive mener rundt funksjonalitetene. Fem testpersoner deltok i brukertestene, som ble gjennomført digitalt med Microsoft Teams. Før brukertestene ble gjennomført på testpersonene, kjørte vi en pilottest på veileder for å se om at alt fungerte som ønsket.

Kapittel 13

Resultat brukertest

I dette kapittelet skal vi ta for oss resultatene som kom fra brukertestene som ble gjennomført på fem testpersoner.

13.1 Bakgrunnsinformasjon

Fire av bøndene hadde beiteområde på Vestlandet, mens den ene bonden holdt til i Trøndelag. Det varierte hvor mange sauer bøndene hadde, og variasjonen var fra 130 til 230 sauer. Det var også interessant å få vite hvor lenge de hadde vært bønder, slik at en kan vite om de har mye erfaring, og hvilke løsninger de har brukt for å holde kontroll på sauene. Bonden med minst erfaring hadde jobbet med sauer i 12 år, og den med mest erfaring hadde jobbet som bonde i 27 år. Alderen på testpersonene var over et stort spekter, der den yngste var i trettiårene, mens den eldste var i sekstiårene. Beiteområdene som de slapp sauene ut på kunne variere fra et par hundre mål, til flere tusen mål.

13.2 Oppgave 1 (Velg periode og rovdyr)

13.2.1 Brukertest observasjoner

De fleste testpersonene klarte å velge tidsperiode. Hvordan de løste dette var ulikt, noen valgte å skrive inn datoen i tekstfeltet til kalenderen, mens andre klikket på kalenderikonet og navigerte seg frem til korrekt dato. Det var en som ikke så menyen til venstre, ettersom vedkommende kun fokuserte på kartet, og det som foregikk der. Etter testleder spesifiserte at en skulle velge dato i menyen, fant personen frem, og da gikk det fint å velge dato. Etter dette kommenterte vedkommende: “neste gang jeg bruker applikasjonen vet jeg at det eksisterer en meny, og jeg vil ikke gå i samme felle igjen”.

Alle personene klarte å filtrere rovdyr, enten det var å kun velge jerv, eller å legge til bjørn og ulv i ettertid. Deretter var neste oppgave å finne ekstra informasjon om en hendelse av jerv i kartet. Her slet flere testpersoner. Et problem var at de ikke hadde zoomet langt nok ut, og fikk ikke sett noen hendelser av jerv på kartet. Et annet problem var at de ikke forstod hvilken markør en skulle klikke på for å se informasjon om jerv. De slet med å forstå forskjellen på markørene som viste sau, og markørene som viste rovdyr.

13.2.2 Spørsmålsbesvarelse

Kommer det tydelig frem hvilke rovdyr som hører til hver markør?
Flere av personene brukte rovbase til vanlig, og kjente derfor igjen markørene, og visste fra før hva disse betydde. En av personene har tilgang til den interne rovbasen, og er ansatt av statens naturvernforbud. Vedkommende forstod alt som hadde med rovdyr å gjøre, og klarte å navigere gjennom alt, og visste hva alle markørene betydde. For andre som ikke hadde brukt rovbase, var det forvirrende med fargene. Etter testleder forklarte fargene, ble det tydeligere for dem. Når en zoomet ut så ble markørene slått sammen til en klynge der det stod antall sammenslåtte markører. Her var det flere som ikke forstod hva det summerte tallet betydde med en gang.

Er det tilhørende ikonet inni markøren forståelig?
De som hadde brukt rovbase før, forstod hva ikonene betydde. De personene som ikke hadde brukt rovbase slet med å forstå hva de ulike ikonene inni markørene betydde. Slik som sau inni markøren forstod folk flest at betydde død sau. De slet litt mer med å forstå forskjellen på spor, hår, ekskrement og DNA-prøve. Etter samtlige av testpersonene hadde klikket seg rundt på applikasjonen, forstod de alle ikonene, og visste hva ikonene betydde til neste gang de skulle bruke applikasjonen.

13.3 Oppgave 2 (Oversikt over tilsynsturer)

13.3.1 Brukertest observasjon

Det var delt mellom testpersonene når de skulle finne oversikt over alle tilsynsturene. Flere personer gikk først på årsrapport, og trodde det var der de fikk sett alle turene. Andre slet generelt med å finne tilsynsturene i menyen. En testperson sa at det var misvisende når det stod "Sau" og ikke "tilsynstur" i menyen. En annen person lurte på om det kun var bonden som kunne registrere tilsynstur, eller om hvem som helst kunne registrere turer. Resten av testpersonene fant frem på første forsøk.

De fleste testpersonene slet med å finne ekstra informasjon om en saueflokk. Mange var usikre på om det var rovdyrene eller en saueflokk de skulle klikke på. Noen personer lurte på om det var i menyen eller på kartet de fant ekstra informasjon om en saueflokk. I noen tilfeller måtte testleder tipse personene om å filtrere bort alle rovdyrene som var valgt, slik at de kun så de aktive saueflokkene. Etter hintet klarte alle personene å finne ekstra informasjon om saueflokkene.

13.3.2 Spørsmålsbesvarelse

Kommer det tydelig frem hva tallet på en saueflokk betyr?
En person var usikker på om det var antall voksne dyr, eller om det var summen av lam og søye. Resten av testpersonene antok at dette var totalt antall sau (søye + lam) i flokken, noe som var korrekt.

Savnet du noe informasjon når du klikket på en saueflokk?

Flere påpekte at de bruker slipsfargene til søyene, og deretter regnet ut antall lam basert

på slipsene. De ønsket dermed at det stod hvor mange søyer det var, hvor mange lam en antok at det skulle være og hvor mange lam det faktisk var. Når det er rovdyr ute så tar de oftest lammene, og en ønsket derfor å ha en oversikt som symboliserte om det manglet noen lam.

13.4 Oppgave 3 (Måneds- og ukesvisning med heatmap og saueflokk)

13.4.1 Brukertest observasjon

Det var variert hvor enkelt testpersonene klarte å navigere seg til å se på bevegelsene på sauene. Mange slet med å finne ut hvordan en gikk fra tur til tur, uavhengig om det var fra ukes- eller månedsvisning. Et par testpersoner prøvde å klikke på den bestemte turen i listen, noe som ikke var mulig. Etter hvert forstod alle at de måtte navigere mellom turene med å bruke neste- og forrige-knappen. En person klarte dette på første forsøk.

Alle personene klarte å navigere mellom heatmap og saueflokk. Det var heller ikke noe problem for testpersonene å variere mellom ukes- og månedsvisning. En person ønsket å kunne se grenseområde for beite på kartet, slik at en kunne se om en sau hadde beveget seg utenfor ønsket område. En annen person påpekte at det var vanskelig å memorere hvor den forrige saueflokken stod, når en navigerte seg mellom turene. Vedkommende kom med et forslag om å vise den forrige turen med svakere farge, eller legge igjen noe form for spor, slik at en enklere kunne se bevegelsesmønsteret.

13.4.2 Spørsmålsbesvarelse

Hvilken av de to visningene, heatmap eller saueflokk, syns du var best?
Flesteparten av testpersonene foretrakk å bruke saueflokkvisningen, ettersom en fikk opp antallet på hver saueflokk. Med saueflokkvisning kan en se hvordan de beveger seg, og om de er innenfor et bestemt område. En person syns saueflokkvisningen var rotete, og det var vanskelig å få fullstendig oversikt, ettersom flokkene gikk om hverandre. Vedkommende begrunnet med at heatmap gjorde det visuelt enklere å se hvor det var mye og lite aktivitet.

Var det noen forskjell på hvilken visning som var best på ukes- eller månedsvisning?
Det var to svar som gikk igjen på dette spørsmålet. Noen syns at saueflokkvisning var best på både uke og måned. Det andre svaret var at saueflokkvisning var best på uke, og heatmapvisning var best på måned, ettersom det ble for mange sauer på måned dersom en brukte saueflokkvisning. De som foretrakk saueflokk på begge visningene, så kun på om sauene var innenfor et område. De brydde seg ikke om det var rotete, eller uoversiktlig. De som foretrakk heatmapvisning på måned, konsentrerte seg om bevegelsen til sauene, og hvor det var mest aktivitet på hver tur.

Kan denne funksjonaliteten hjelpe å holde saueflokkene borte fra områder med rovdyr?
Flere av personene som hadde beiteområde på Vestlandet har ikke hatt problem med rovdyr, og kunne ikke svare på dette spørsmålet. Testpersonen med beiteområde i Trøndelag har hatt mye problemer med rovdyr. Vedkommende svarte at dersom *BeiteSpor* viste

rovdyr, ville personen sette inn ekstra tilsyn, for å passe på dyrene sine. I tillegg ville vedkommende snakke med andre bønder både for å vite hvor de har sett rovdyr, og advare om hva som ble observert på applikasjonen. Testpersonen ville også tatt grep som innebærer å flytte dyrene til en annen lokasjon dersom det ble observert mange rovdyr. En av bøndene fra Vestlandet har hatt problemer med jerv tidligere, men kommenterte at det var lite å gjøre for å hindre jerven. Hovedgrunnen var at sauene var spredt over flere tusen mål, og bonden hadde ikke kapasitet til å beskytte hele beiteområde for jerv. Vedkommende mente at eneste løsning var å ta dyrene av beite.

13.5 Oppgave 4 (Bonitet og PDF)

13.5.1 Brukertest observasjon

Alle testpersonene klarte på første forsøk å ta på bonitetslaget på kartet. Noen var usikre på hvordan en valgte synlighet av bonitet på kartet, men alle klarte dette innen kort tid. På grunn av problemer med nedlasting av rapport, ble denne tilsendt på Microsoft Teams til testpersonene. Alle kommenterte at det ikke var nødvendig å skrive hvor mange sauer det var av hver farge. Istedenfor ønsket de at rapporten viste hvor mange lam og søyer det var, og muligheten for å se slipsfarger. Det var også motsigende resultater fra brukertesten. Noen ønsket at en skulle notere inn individnummer på hver enkelt tur, som igjen skulle vises på rapporten. Andre personer mente at dette ikke var mulig, ettersom en må være svært nære for å se individnummeret, noe en sjeldent er på et beite med flere hundre mål.

13.5.2 Spørsmålsbesvarelse

Hvilken type sauerapportering gjør du, og hva skal med i rapporten?
Alle testpersonene sier at de rapporterer inn antall søyer og lam. En av testpersonene sier at det blir rapportert til statsforvalteren. De andre sier at de skriver inn i sauekontrollen, som har et eget skjema for tilskudd. Sauer blir registrerte på starten av året, der en må skrive inn hvor mange en planlegger å sende på utmarksbeite. Basert på innsendingen, kan en få generert lister som gir oversikt over hvilke sauer som er på utmarksbeite. Listene blir også brukt for å hjelpe med tilsynsturene. Etter sesongen er ferdig skriver de igjen ned hvor mange sauer de har. Ingen av testpersonene skriver ned farge på sauene, kun hvor mange søyer og lam de har.

Hjelper bonitetslagene deg med å forstå hvorfor sauene beveger seg slik de gjør?
Flere av testpersonene kjenner til bonitet, og bruker bonitet til å velge hvor sauene skal beite. Videre kommenterte de at fargene hjelper med å forstå hva de ulike områdene betyr, men de som brukte det til vanlig kunne fargene utenat. Andre mener at bonitet ikke er viktig, og at sauene selv velger å gå der de finner mat. Flere sa at selv om de visste om bonitet, trengte de ikke å se på et bonitetskart, ettersom dyrene går i de samme beitene om igjen, og en vet hvor det er gode beiteforhold.

13.6 Spørsmål etter gjennomført brukertest

13.6.1 Hva bruker du nå for å holde kontroll på sauene dine?

Alle utenom en av testpersonene brukte en eller flere digitale løsninger for å holde kontroll på sauene. De brukte ulike digitale løsninger, men enten FindMy, Telespor eller Smartbjella ble brukt. En person brukte både FindMy og Telespor, ettersom den ene var avhengig av mobildekning og den andre brukte satellitt til kommunikasjon. I tillegg til bruk av digitale løsninger, gikk alle tilsynsturer, som vil bli dekket i neste spørsmål.

13.6.2 Har du registrert sauer manuelt tidligere?

Alle testpersonene går minst en tilsynstur i uken, der de registrerer sauene manuelt. Det varierte mye hvordan de ulike testpersonene gjennomførte en tilsynstur. En person sjekket hvor dyrene var på Telespor, og planla turen basert på deres siste registrerte posisjon. Vedkommende registrerte dyrene i en tilsynsbok. Et annet alternativ var å skrive ut listen av dyr som hadde blitt registrert i sauekontrollen. Dersom personen fant et dødt lam ble dette registrert i sauekontrollen. Noen av testpersonene gikk på tilsynstur i fjellet for å se om noe var galt, enten om et lam gikk uten moren eller om det hadde vært rovdyr. Grunnen var at fjellbeite bestod av flere tusen mål, og en så ikke alle dyrene som var ute på beite. Flere av testpersonene memorerte antall sauer de hadde observert i hodet. Her nevnte en person at en ble usikker på hvor mange sauer som ble telt, og det var ikke alltid at totalt antall sauer ble helt nøyaktig.

13.6.3 Tror du denne løsningen som du har prøvd ut nå kan hjelpe deg til å bedre forstå hvordan saueflokkene beveger seg?

Flertallet av testpersonene mener at det hadde vært en god idé å bruke en mobilapplikasjon til å registrere sauer, og deretter se dataene på en webapplikasjon i ettertid. Visualisering av dataene gir et helhetssyn på hvor sauene er. De spesifiserer derimot at for at en skal kunne bruke et slikt system, må det være enkelt og brukervennlig, ettersom de ikke hadde mye erfaring med digitale løsninger. En person kommenterte at en kunne studere dataene for å vite nøyaktig hvor det burde gjødsles på slutten av sesongen, slik at beite blir klart til neste sesong. Videre kunne en studere dataene for å effektivisere innsamling av sau, ettersom en vet hvor de sist befant seg. Mobilapplikasjonen til *BeiteSpor* måtte være tilgjengelig uten internett, på grunn av at store deler av beite ikke hadde internett-dekning. Det var en person som ikke støttet bruk av webapplikasjonen. Vedkommende begrunnet påstanden med at det ble brukt FindMy, og at personen var skeptisk til bruk av teknologiske løsninger til beitebruk.

13.6.4 Var det noe som manglet for å kunne se bevegelsene til sauene?

Et ønske var å dele opp informasjonen om en saueflokk i søyer og lam, istedenfor å kun skrive totalt antall sauer, der en ikke hadde kjennskap til fordelingen. I tillegg var det et ønske om å legge inn flere observasjoner når en var ute på tilsynstur, der et alternativ var å skrive inn tvillinger. En person ønsket å se forrige tur på kartet, samtidig som den nåværende, slik en kunne se bevegelsen mellom turene. Det siste ønsket var å integrere

sauekontrollen inn i webapplikasjonen.

13.6.5 Kan løsningen gjøre jobben din som sauebonde enklere?

Flertallet tror at webapplikasjonen kan forbedre og effektivisere jobben som sauebonde. Det er ønskelig at applikasjonen er brukervennlig og enkel å bruke for personer som ikke har en teknologisk bakgrunn. Registreringen på mobilapplikasjonen må være enkel, slik at en sparer tid, og at det er mer effektivt enn å notere med penn og papir. Det var en person som ikke støttet digitalisering, og mente at det hadde blitt mer arbeid enn fortjeneste. Vedkommende sier at han har FindMy, men bruker ikke webapplikasjonen som hører til. Det eneste vedkommende bruker FindMy til er å motta varsel dersom et dyr ikke har beveget seg på flere timer.

13.7 Oppsummering

Det ble gjennomført brukertester på fem testpersoner, der fire var fra Vestlandet, og en fra Trøndelag. Oppgavene og spørsmålene fra brukertestene er grunnlaget for resultatene. Webapplikasjonen fikk mange gode tilbakemeldinger, som at brukergrensesnittet var for det meste oversiktlig, og de forstod mye av funksjonaliteten. Det ble ikke avdekket kritiske feil i løpet av brukertesten, men det kom frem flere mindre feil. Testpersonene slet med å se forskjell på markørene for sauer, og markørene for rovdyr. Videre når markørene ble slått sammen til klynger, ble testpersonene usikre på hva tallet i midten av klyngen betydde. Flere testpersoner slet også med å finne oversikten over tilsynsturer.

Del V

Diskusjon

Basert på planlegging og implementasjon av løsningen, samt gjennomføring og resultater fra brukertesten, skal denne delen diskutere brukertestene, prosessen og løsningen, og sammenligne løsningen med eksisterende løsninger. Til slutt skal vi se på hvordan resultatene besvarer forskningsspørsmålene.

Kapittel 14

Resultat brukertest

Med utgangspunkt i resultatene som kom fra brukertestene i kapittel 13, skal dette delkapittelet diskutere de fremtredende resultatene. Enten med løsninger til et problem eller et fokusområde som vil være viktig i en potensiell viderutvikling av løsningen.

14.1 Turoversikt i applikasjonen

Fra observasjoner i kapittel 13.3.1 kom det frem at flere av testpersonene var usikre på navigasjonen rundt turer med visning av saueflokker for den aktuelle perioden. Det første problemet var at testpersonen ikke forstod hvor i menyen oversikten over turene var plassert. For en testperson var tittelen misvisende for nedslagsmenyen med turoversikt, og kom med et forslag om å bytte fra "Sau" til "Tilsynsturer". Dette er mer forståelig og informativt, som reduserer risikoen for misforståelser. I den nåværende versjonen av *BeiteSpor* er nedslagsmenyen til turoversikt plassert nederst i menyen. Menyen til venstre består i hovedsak av nedslagsmenyer som kan ekspanderes. Hvis flere av nedslagsmenyene over "Sau"-menyen er ekspandert, vil denne gjemmes og brukeren må skrolle ned for å se hele turoversikten. Derfor kan enten en høyere plassering av nedslagsmenyen eller å plassere den i en separert meny føre til økt forståelse.

14.2 Navigasjon mellom oppsynsturer

Et problem flere av testpersonene hadde i brukertesten var navigasjonen mellom tilsynsturene, som ble presentert i kapittel 13.4.1. Fra resultatene kom det frem at det var vanskelig å forstå navigeringen mellom turene med forrige- og nesteknappen. Derfor er det et potensial for forbedring av navigasjonen mellom turene. Flere av testpersonen prøvde å klikke på turene i listen for å velge en spesifikk tur, som kan vise seg å være en naturlig navigasjonsmetode for målgruppen. Det kom også et ønske fra en av testpersonen om visning av nåværende og forrige tur i kartet, slik at det ble enklere å se bevegelsesmønstret fra tur til tur. En løsning på problemet kan være å vise forrige tur med lavere synlighet enn den nåværende turen på kartet.

14.3 Informasjon om en saueflokk

For å få ekstra informasjon om en saueflokk i applikasjonen, må brukeren klikke på en av flokkene i kartet. Funksjonen ble testet i oppgave 2 i bruketest og observasjonene ble beskrevet i kapittel 13.3.1. Det første problemet testpersonene støtte på, var å skille mellom rovdyrikon og saueflokkikon. I den nåværende versjonen av webapplikasjon er det lite skille mellom ikonene for hendelse der sauer er drept av rovdyr og ikonet for observasjon av saueflokker. Ofte er det et flertall av rovdyrhendelser i kartet, som gjør at saueflokkene kan bli borte i mengden. Derfor vil et mer adskilt ikon for saueflokkene gjøre det enklere å oppdage de.

Når testpersonene fant ut hvilke ikoner som tilhørte saueflokkene, forstod de fleste at ikonet måtte trykkes på for å vise ekstra informasjon. Innholdet i informasjonboksen var lagt lite fokus på under utviklingen av løsningen. Det var derfor nyttig med tilbakemeldinger på denne delen av applikasjonen. Den nåværende løsningen inneholdt kun antall i flokken som metainformasjon. Fra spørsmål knyttet til oppgave 2 i brukertestene, beskrevet i kapittel 13.3.2, kom det frem at viktig informasjon om flokkene manglet. Antall lam og søyer ble beskrevet som en viktig tilleggsinformasjon i flokken siden det er viktig at søyer og lam går sammen. Dette er informasjon som blir registrert i mobilapplikasjonen og er tilgjengelig i databasen, men som ikke blir vist i dagens versjon.

14.4 Visning av saueflokker

I applikasjonen er det blitt implementert to versjoner av visning for saueflokkene, som er blitt beskrevet i 11.9. Disse funksjonene ble testet og stilt spørsmål rundt i oppgave 3 i brukertesten. Resultatene av spørsmålene knyttet til oppgaven ble presentert i kapittel 13.4.2. Et flertall av testpersonene foretrakk visningen av saueflokker som markører fremfor heatmap, men det var forskjell på om det var måneds- eller ukesvisning. For månedsvisning var det flere som syntes heatmap var en mer oversiktlig visning. Dette avhenger av hva brukeren bruker applikasjonen til. En av testpersonene brukte kartet til å se om sine egne sauer holdt seg innenfor beiteområde, som var enklere å observere ved at saueflokkene var visualisert som markører. Andre mente heatmap var en bra løsning for å kunne se hvor sauene oppholdt seg mest i løpet av den valgte perioden. Derfor vil en løsning slik det er i dag, med mulighet for å slå av og på visningstypene være det optimale.

14.5 Viktighet av rovbase

I kapittel 13.2 ble observasjoner og spørsmål for oppgave 1 presentert. I denne oppgaven ble integreringen mot rovbase testet. Flere av testpersonene hadde interagert med rovbase tidligere og hadde derfor kjennskap til markører som er gjenbrukt i denne delen av applikasjonen. Fra brukertestene kom det også frem en forskjell på hvor utbredt rovdyrskade var. Fire av testpersonene drev med sauedrift på Vestlandet, mens en av testpersonene drev i Trøndelag. Fra oppfølgingsspørsmål rundt rovdyr, kom det frem at det var et mindre problem av rovdyr hos testpersonene fra Vestlandet. En av testpersonene fra Vestlandet hadde opplevd rovdyrangrep, men det var på grunn av at beiteområde gikk inn på Har-

dangervidda. Kapittel 4.2 viste at det var forskjeller i utbredelse av rovdyrangrep på sauer i landet. Dette samsvarer med resultatene fra brukertestene som var knyttet til rovdyrangrep. Generelt hadde testpersonene fra Vestlandet et mindre problem enn testpersonen fra Trøndelag. Presentert i kapittel 4.2 og vist i figur 4.3 hadde Trøndelag et mye større tap av sauer til rovdyr enn Vestland i 2021. Derfor vil integrasjon mot rovbase være aktuelt for de av brukerne som er fra områder med høyere trussel fra rovdyr. I applikasjonen er dette løst ved bruk av filtrering av rovdyrene, som er vist i figur 11.12. Alle rovdyr kan filtreres bort hvis det ikke er relevant for den aktuelle brukeren.

14.6 Genering av PDF-rapport

Under første pilottest kom det frem at funksjonen som skulle generere PDF-rapport ikke fungerte som ønsket. Generering av rapporter fungerte når applikasjonen ble kjørt lokalt, men ikke når den ble publisert til produksjonserveren. I brukertesten ble en rapport som ble generert lokalt sendt over til testpersonen på PDF-format. Funksjonen hvor brukeren generer rapporten i applikasjonen ble derfor ikke testet. Siden det eneste som ikke ble testet var at brukeren selv trykket på en knapp for å genere rapporten, anses det som liten betydning for gjennomføringen av brukertesten. For en potensiell videreutvikling og produksjonssetting av applikasjonen, må funksjonen fikses.

14.7 Oppsummering

Hensikten med brukertestene var å avdekke hva som var bra med *BeiteSpor*, og hva som kunne forbedres. Fra brukertestene kommer det frem at det var flere mindre feil, der brukergrensesnittet ikke var intuitivt eller at det var mangel på informasjon. De samme feilene gikk igjen hos flere testpersoner. I pilottesten kom det frem at PDF-genereringen ikke fungerte i webapplikasjonen i produksjon. Dette førte til at en rapport ble sendt til hver testperson. Det var delte meninger om implementasjonen av Rovbase, ettersom det var mindre rovdyr på Vestlandet sammenlignet med Trøndelag.

Kapittel 15

Evaluering av prosess

I dette kapittelet blir det diskutert gjennomføring av de tre prosessene design, utvikling og gjennomføring av brukertestene.

15.1 Design

Prototypen ble utviklet ved bruk av prototyperammeverket Figma. Dette ble gjort tidlig i prosessen og i en egen designfase, slik at prototypen var ferdig før utviklingen skulle begynne. Det ble ikke gjennomført en brukertest på designet, kun på den ferdig utviklede løsningen. For å optimalisere designet, burde det blitt gjennomført en iterasjon til på prototypen, ettersom det er mer effektivt å endre prototypen, fremfor å endre designet i webapplikasjonen.

15.2 Utvikling

Ettersom en fullstendig prototype ble laget i Figma før utviklingen, hadde utviklerne en klar plan for hvordan designet skulle se ut. Dette økte effektiviteten på utviklingen av brukergrensesnittet til webapplikasjonen. Bruken av Kanban-board fungerte optimalt til utviklingsteamet, ettersom en hadde en oversiktlig plan for hva som skulle gjøres til enhver tid, og hvem som gjorde hva. På grunn av at teamet bestod av to utviklere, var det ikke nødvendig med en mer avansert metodikk enn Kanban. Teamet kommuniserte kontinuerlig under hele utviklingsfasen, og ved å ha alle oppgavene på et Kanban-board kunne en se fremgang etterhvert som hver oppgave ble flyttet mot høyre, som betød utført oppgave. Ved bruk av en agile metode, kunne utviklerne gjøre endringer underveis, og legge til eller fjerne oppgaver fra Kanban-board.

15.3 Gjennomføring av brukertest

For å sikre at brukertesten ble gjennomført på en god og effektiv måte, ble det laget en plan som er beskrevet i kapittel 12.2. Planen ble testet flere ganger av utviklerne, og en gang av veileder, for å teste på noen som ikke hadde like god kjennskap til løsningen som utviklerne. Eventuelle kritiske feil ble oppdaget i denne gjennomgangen. En feil som ble

oppdaget var at PDF-nedlastningen ikke fungerte på serveren med AAS. Løsningen på problemet var at testpersonene simulerte at de klikket på PDF-nedlasning, og ble tilsendt en PDF på Microsoft Teams.

Brukertestene ble gjennomført digitalt, som forklart i kapittel 12.2.4. Fra pilottesten med veileder, kom det frem at en digital brukertest kunne by på problemer. På grunn av at flere av testpersonene hadde lite teknologisk erfaring, ble det satt av ekstra tid til brukertestene, dersom tekniske problemer skulle oppstå. Nesten alle personene brukte opp den allokerte tiden, ettersom fire av fem testpersoner slet med Microsoft Teams. De fleste slet med å dele skjerm, noe utviklerne hadde forutsett som det største problemet. Løsningen var at testleder sendte bilder på Microsoft Teams, som stegvis forklarte hvordan en skulle dele skjerm. Et annet problem som oppstod hos et par av deltagere var at mikrofonen ikke fungerte. Dette problemet var vanskeligere å navigere testpersonene gjennom, på grunn av at det ofte var maskininnstillinger som var problemet, og hvert operativsystem har ulike måter å fikse dette på. Løsningen ble å snakke via mobiltelefonen, og dele skjerm ved bruk av Microsoft Teams. Etter alle de tekniske problemene var løst, ble brukertestene gjennomført uten videre problemer.

15.4 Oppsummering

Prototyperammeverket Figma ble brukt for å utforme designet til applikasjonen. Designet til applikasjonen ble utformet før utviklingen startet, men sett i etterkant kunne det vært en fordel med en designiterasjon til. Med et ferdig utviklet design i Figma og sammenkjøring gjennom Kanban-boardet, gjorde at det ble en effektiv utvikling av løsningen. Etter planlegging og pilottest, ble bruketestene gjennomført. Det ble satt av ekstra tid for gjennomføringen, siden brukertesten foregikk digitalt, og flere av testpersonene hadde lite erfaring med Microsoft Teams fra tidligere. Nesten alle testpersonene brukte opp tiden som hadde blitt satt. Dette er noe som unnebygger at målgruppen har lite teknisk erfaring, og at det er et høyt krav om brukervennlighet i applikasjonen.

Kapittel 16

Evaluering av løsningen

I dette delkapittelet vil vi evaulere hvordan utviklingen av serveren og brukergrensesnittet gikk for seg.

16.1 Server

Microsofts tjenester ble brukt for å utvikle serveren ved bruk av ANC og AAS for å hoste serveren, og Azure Database. Ettersom alle tjenestene var laget av Microsoft, var de godt integrerte og det var enkelt å sammenkoble de ulike delene. Til å autentisere brukere ble det brukt ANC Identity, som håndterte all brukerdata. Identity opprettet en egen brukerdatabase med all brukerinformasjon, og opprettet gyldige token som ble brukt for å logge brukere inn og ut. Ettersom Identity er godt integrert i ANC, var det effektivt og enkelt å autentisere hvert endepunkt. Ved bruk av REST API på serveren, forklart i kapittel 9.2.1, får en strukturert arkitekturen og deler serveren inn i uavhengige deler. Dette gjorde at det var enkelt å gjøre endringer senere i prosessen.

16.2 Brukergrensesnitt

For å forme brukergrensesnittet ble det brukt React med TypeScript, som hindret typefeil ved kjøring. Ettersom hele React er bygget opp av komponenter, var det naturlig at hver komponent hadde sin egen arbeidsoppgave og sitt eget ansvarsområde. På denne måten kan en ved vedlikehold skifte ut eller endre på enkelte deler av koden, uten at dette skal påvirke andre deler. Brukergrensesnittet ble hostet i AAS, på lik linje som serveren. Fordelen med å bruke AAS i begge tilfellene, var at en kunne bruke erfaringen fra første publisering av serveren til å opprette den neste. Material UI ble brukt som komponentbibliotek for brukergrensesnittet. Ved bruk av dette biblioteket slapp en å bruke tid på å lage basiskomponentene i webapplikasjonen.

Kapittel 17

Kravspesifikasjon

Dette delkapittelet vil gå gjennom både de funksjonelle og ikke-funksjonelle kravene som ble presentert i kapittel 8.2, for å se hvilken grad de har blitt oppnådd.

17.1 Funksjonelle krav

Dette delkapittelet handler om de funksjonelle kravene som står beskrevet i kapittel 8.2.1. De fleste funksjonelle kravene ble implementert og står beskrevet i kapittel 11. De kravene som ikke har blitt oppfylt vil bli dekket i dette delkapittelet.

17.1.1 FK7 - Glemt passord

Det funksjonelle kravet FK7, "Glemt passord", ble ikke oppnådd. Grunnen til at dette kravet ikke ble gjennomført var at det hadde lav prioritet, og ville ikke påvirke resultatene som kom frem i kapittel 13. I delkapittel 12.2.1 kommer det frem at hver testperson fikk opprettet en forhåndsdefinert bruker. Det var dermed ikke relevant å klikke "Glemt passord", ettersom dette ikke var en del av brukertesten eller interessant for forskningsspørsmålene.

17.1.2 FK13 - Vise bondens tilsynsrute på kart

Det funksjonelle kravet FK13, "Vise bondens tilsynsrute på kart", ble først utviklet i webapplikasjonen. Etter samtaler med veileder Hvasshovd, kom det frem at hovedfokuset var på bevegelsene til saueflokkene. For å fokusere kun på sauene og bevegelsesmønsteret, ble turruten til bonden fjernet, ettersom det kom inn flere faktorer som gjorde webapplikasjonen mer komplisert enn nødvendig.

17.1.3 FK9 - Generere en PDF-årsrapport

Diskutert i kapittel 14.6 var det problemer med generering av PDF-rapporter, som gjør at FK9 ikke er tilfredsstilt. Generering av rapporter fungerte når webapplikasjonen ble kjørt lokalt, men ikke når den ble kjørt i produksjon. Med hensyn til tidsbruk og liten påvirkning på brukertesten, ble problemet nedprioritert.

17.2 Ikke-funksjonelle krav

17.2.1 Sikkerhet

Ved bruk av ANC Identity, beskrevet i delkapittel 11.2, sørger en for at systemet kun er tilgjengelig for personer som har opprettet bruker. En tur er knyttet til en bestemt bruker, og er kun synlig for denne brukeren. Data blir sendt sikkert fra serveren ved bruk av HTTPS.

17.2.2 Vedlikehold

I delkapittel 9.1.1 og 9.2.2, står det om React og ANC som er delt inn i brukergrensesnitt og server. React ble brukt for å lage brukergrensesnittet, og er bygget opp av komponenter, som er enkle å endre på i ettertid uten å endre på hele kodebasen. ANC ble brukt for å lage API-et, og er delt inn i modeller og kontrollere for å skille mellom databasemodeller og endepunkter.

17.2.3 Brukbarhet

For å teste brukbarheten til systemet, ble det utført brukertester, som står beskrevet i kapittel 13. I bakgrunnen til testpersonene kommer det frem at det er varierende teknologiske kunnskaper, både lite og mye kunnskap. Basert på resultatene kommer det frem at webapplikasjonen er enkel å bruke. Det er noen personer som sliter med noen av oppgavene, men etter forklaring fra testleder forstår de hvordan den aktuelle oppgaven skal utføres. Denne forklaringen fra testleder kan sammenlignes med en opplæring, og en kan dermed si at personene klarer å bruke webapplikasjonen og dens funksjoner etter opplæring.

17.2.4 Ytelse

En gratisversjon av Microsofts AAS ble brukt til å kjøre serveren. Gratisversjonen hadde noen ulemper som satte begrensninger for ytelsen. Dersom serveren ikke var brukt på en stund, ville den gå i dvale. Den første forespørselen ville ta noen sekunder, ettersom serveren måtte våkne. Derfor er testing av ytelsen til applikasjonen først relevant når applikasjon eventuelt skal publiseres til en produksjonserver. Gratisversjonen har blitt testet gjennom brukertestene og av utviklerne, men dette har blitt gjort med få brukere samtidig.

17.3 Oppsummering

De fleste funksjonelle kravene ble implementert, utenom tre krav. Dette var på grunn av at kravet hadde lav prioritet, eller at kravet ikke var nødvendig for å gjennomføre brukertestene. Det var begrensninger for å teste ytelsen til systemet, på grunn av at det ble brukt en gratisversjon til AAS. De resterende kravene til ikke-funksjonelle krav var godkjent.

Kapittel 18

Sammenligning med eksisterende løsninger

I kapittel 6 står det beskrevet tidligere arbeid som var gjort innenfor digitalisering av sauer. Telespor, FindMy og Smartbjella tilbyr sporingsenheter som blir festet fysisk på en sau, og sender posisjonen minst en gang i døgnet. På denne måten får en kartlagt hvor sauene er hver dag. Ifølge regelverket som står beskrevet i kapittel 4.1, står det at bonden skal gå minst en tilsynstur i uken. Det vil si at webapplikasjonen utviklerne har laget vil komme i tillegg til sporingsenhetene som allerede eksisterer, ettersom den handler om å digitalisere tilsynsturer.

En applikasjon som har et annet bruksområde enn sporingsenhetene, som står beskrevet i kapittel 6, er BeiteSnap. Denne applikasjonen kan sammenlignes med noe av funksjonaliteten som ble beskrevet i forprosjektet, kapittel 5, der en kan registrere et skadet dyr, et dødt dyr eller om en har observert et dyr. Videre blir posisjonen til dyrene vist på et kart, som kan ha noen likheter til *BeiteSpor* som utviklerne har laget. Begge applikasjonene er basert på at personer skriver inn hva de har observert, på BeiteSnap er dette tilfeldig turgåere, og i *BeiteSpor* er det bonden selv som har registrert dataene. Fordelen med *BeiteSpor* er at bonden vet at dataene som ble registrert er korrekte, i motsetning til dataene som ble registrert av ukjente forbipasserende. Videre har BeiteSnap ikke mulighet til å sammenslå flokker, navigere mellom ulike turer, vise om det er en søye eller lam som går alene, eller mulighet til å vise saueflokker og heatmap på kart.

En applikasjon som har mulighet til å vise heatmap av dyreaktiviteten, er ZooMonitor. I likhet med *BeiteSpor*, ble det gått jevnlige tilsynsturer av dyrene i dyrehagen, som ble registrert av dyrepasserne. I ZooMonitor kunne en laste opp et kartbilde av områdene til dyrene, slik at en kunne plassere hvert dyr nøyaktig på kartet. Videre ble heatmap-analysen brukt til å se på bevegelsesmønsteret, som førte til at de la inn tre matstasjoner for å øke bevegelsene til dyrene. Heatmap-funksjonaliteten i *BeiteSpor* ble laget for at bøndene skulle få en oversikt over bevegelsene. Både BeiteSnap, ZooMonitor og *BeiteSpor* har mulighet for å laste ned en rapport som viser en oversikt over alle dataene som har blitt registrert.

Kapittel 19

Prosjektevaluering

Delkapittelet vil ta for seg hvert av de tre forskningsspørsmålene som ble laget til problemstillingen, og diskutert hvordan de har blitt besvart gjennom resultatene.

19.1 FS1.1 - Kan et digitalt system fremstille innsamlet data fra tilsynsturer på en mer oversiktlig og effektiv måte sammenlignet med penn og papir?

BeiteSpor har blitt utviklet i samarbeid med veileder Hvasshovd, med bakgrunn i tidligere arbeid på feltet, eksisterende løsninger og hvilke krav myndighetene setter for tilsynsturer. Løsningen tilbyr bøndene en visning av hvor sauene er blitt registrert, hvor mange som ble registrert, når de ble registrert, hvilken tur de ble registrert på, hvor rovdyrene var observert og bonitetslag på kart. I tillegg sammenslår webapplikasjonen mindre saueflokker til en stor flokk, og gir brukeren mulighet til å navigere mellom ulike tilsynsturer som har blitt registrert. Webapplikasjonen tilbyr fremstilling av dataene i form av PDF-rapport, som viser all nødvendig informasjonen basert på kravene for å få tilskudd fra myndighetene. Fra brukertestene kommer det frem at mye av funksjonaliteten til webapplikasjonen er enkel og intuitiv å bruke, selv for bønder med lite teknologisk erfaring. Ingen av testpersonene brukte noen form for teknologiske verktøy for å registrere sauer på tilsynsturer, eller analyserte tidligere data for å se hvor sauene kunne befinne seg. Flertallet av testpersonene kommenterte at *BeiteSpor* ville forenkle og effektivisere tilsynsturer sammenlignet med dagens løsning med penn og papir.

19.2 FS1.2 - Hvilke faktorer bidrar til en bedre forståelse av bevegelsesmønsteret, og øker effektivitet i videre tilsynsturer?

Hensikten med *BeiteSpor* er å hjelpe bonden med å få et bedre bilde av hvordan sauene beveger seg for å kunne sette inn tiltak dersom det er nødvendig. Det har blitt implementert flere funksjoner i webapplikasjonen som skal bidra til at bonden får en bedre forståelse for bevegelsesmønsteret til sauene. Gjennom brukertestene kom det frem et variert syn på

hvilke faktorer som bidro mest til å forstå bevegelsesmønsteret. En av testpersonene ville sjekke om sauene holdt seg innenfor sitt eget beiteområde, mens en annen testperson ville sammenligne bevegelsen til flokkene fra tur til tur.

Implementasjonen av bonitetslag i kartet og integrasjon mot Rovbase mente testpersonene hadde varierende betydning for å forstå bevegelsemønsteret til sauene. Diskutert i kapittel 14.5 varier trusselen for rovdyrangrep fra område til område. Det vil derfor variere hvor viktig bruken av Rovbase er for brukeren i *BeiteSpor*. Resultater fra brukertesten viser også at bruken av bonitet er varierende blant testobjektene. Noen har brukt bonitet aktivt tidligere for å se på beiteforhold, mens andre mente at egne erfaringer om beiteområde var nok informasjon for å vite hvor sauene bevegde seg for å finne mat. Resultatene viser at det ikke er en bestemt faktor som økter forståelsen av bevegelsesmønsteret, men flere faktorer som kan være med å påvirke bevegelsesmønsteret. Hvilke faktorer varierer fra bonde til bonde, basert på hva bonden mener er viktig for å forstå mønsteret til sauene.

19.3 FS1.3 - Kan et system integrert med rovdata hindre eller redusere tap av søye og lam til rovdyr?

For å tilby brukerne mer informasjon og gjøre *BeiteSpor* mer komplett, ble det integrert rovdyr fra Rovbase. Tidligere har bøndene måtte bruke Rovbase som en selvstendig applikasjon, og sammenligne med hvor de hadde sett sauene sine når de gikk en tilsynstur. Ved å samle alt på en plass, kan bøndene få bedre oversikt over hvor sauene deres er registrert, og hvor rovdyrene er observert.

Fra brukertestene var det delt mellom testpersonene om rovdyroversikten ville redusere eller hindre tap av sauer. Argumentene som støttet påstanden gikk ut på at beitelag og bonden selv ville øke tilsynsturene basert på hvor det er blitt observert rovdyr. Som et tiltak kan dyrene enten flyttes til en annen posisjon, eller gjennomføre en tidligere innsanking enn planlagt. Motargumentet var at beiteområdet til sauene var for stort til at bonden klarte å hindre rovdyrene i å angripe sauene. Som nevnt i kapittel 4.2, var det mer rovdyr i Trøndelag, enn på Vestlandet. Testpersonen som var imot bruk av rovbase, hadde mindre erfaring med rovdyr og var usikker på hvilke tiltak som kunne innføres, ettersom vedkommende kun hadde opplevd rovdyr en gang. Basert på dette, kan BeiteSpor i teorien forhindre eller redusere tap av søye eller lam dersom bonden har mulighet til å innføre tiltak. For å finne ut mer i praksis bør en gjennomføre flere brukertester på bønder som holder til i Midt-Norge og Trøndelag.

Del VI

Konklusjon og videre arbeid

Som en helhetsvurdering av prosjektet presenteres konklusjonen i denne delen. Det etterfølges med fremlegging av videre arbeid som kan gjøre *BeiteSpor* til en enda bedre løsning.

Kapittel 20

Konklusjon

Målet med prosjektoppgaven er å finne ut hvordan en kan fremstille innsamlet data fra tilsynsturer for å bedre forståelsen av bevegelsesmønsteret til sauene, som er beskrevet i problemstillingen, PB1, i kapittel 2.2.

Webapplikasjonen *BeiteSpor* gir bønder muligheter til å se data som har blitt registrert fra tilsynsturer ved bruk av mobilapplikasjonen som ble utviklet i forprosjektet, kapittel 5. Funksjonaliteten til *BeiteSpor* ble utviklet basert på erfaringene til professor Hvasshovd, med bakgrunn i eksisterende løsninger og tidligere arbeid som har blitt utført på feltet. Ved bruk av kart, kan en vise nøyaktig posisjon på alle saueflokkene som tilsynsansvarlig har registrert, og se detaljert informasjon for hver flokk. Videre kan en velge om en vil se turene med saueflokk- eller heatmapvisning, og om en ønsker å se turene gruppert i ukes- eller månedsvisning. For å få tilskudd til sau på utmarksbeite, må bøndene sende inn informasjon basert på krav fra myndighetene. Ved bruk av *BeiteSpor* kan en få generert all den nødvendige informasjonen til en PDF-rapport.

Fra brukertestene kom det fram at målgruppen for *BeiteSpor* har liten teknisk erfaring, som gir et større krav for brukervennlighet i webapplikasjonen. Samtlige av testpersonene greide å gjennomføre oppgavene i brukertesten, selv om det var forskjeller innad i testgruppen. Etter å ha utforsket brukergrensesnittet eller fått veiledning, forsto samtlige funksjonalitetene i applikasjonen. Resultatene viste at faktorene som var nødvendige for å bedre forståelsen av bevegelsesmønsteret til sauene, varierte mellom testpersonene. Variasjonene stod mellom navigering av saueflokk- og heatmapvisning, visning av bonitetslag eller å se på rovdyrposisjon sammenlignet med saueposisjon. *BeiteSpor* tilbyr filtreringsmuligheter på funksjoner, slik at brukeren selv kan velge hvilke visningstyper som er aktuelle for de.

Løsningen vil ikke være et substitutt for de eksisterende løsningene TeleSpor, Smartbjella eller FindMy, men vil komme som et ekstra digitalt verktøy. Sammenlignet med bruk av tilsynsbok eller notering på papir, vil *BeiteSpor* forenkle og tydeliggjøre innsamlet data fra tilsynsturene, øke forståelsen av bevegelsesmønsteret og tilby nødvendig informasjon som myndighetene krever. Likevel er det fortsatt behov for å videreutvikle løsningen og teste *BeiteSpor* på flere testpersoner.

Kapittel 21

Videre arbeid

Kapittelet handler om hva som kan gjøres for å forbedre *BeiteSpor*, enten å gjøre endringer basert på brukertestene eller legge inn funksjonalitet som ble nedprioritet i utviklingsfasen.

21.1 Turvisning

Etter å ha observert testpersonene navigere seg gjennom *BeiteSpor*, kommer det frem at oversikten over alle turene, med navnet "sau", burde skiftes til "tilsynsturer". Et annet problem var at denne oversikten var langt nede på nettsiden. En kan derfor flytte oversikten over tilsynsturer øverst på siden ettersom dette er den viktigste funksjonaliteten til nettsiden, ettersom den er mer synlig øverst.

Fra resultatene av brukertestene kommer det frem at det var vanskelig å memorere hvor sauene befant seg på den forrige turen. En forbedring kan være å legge til en knapp hvor en kan velge å se forrige tur sammen med nåværende tur. Den forrige turen kan vises i en annen farge enn nåværende tur, eller i svakere farge, for å symbolisere at dette ikke er den nåværende turen. Selve navigeringen mellom turer var det også flere som hadde problemer med. Nederst på siden var det en "neste"- og "forrige"-knapp som skulle navigere mellom de ulike turene. En forbedring på dette er å legge til mulighet for å klikke på hver enkelt tur, slik at en slipper å bruke knappene nederst på siden. Det må vurderes om en skal fjerne knappene nederst, og kun ha muligheten til å klikke på en spesifikk tur, eller om en ønsker å ha begge mulighetene for å navigere gjennom turene.

21.2 Glemt passord

På grunn av mangel på tid og nedprioritering av oppgaver som hadde lav prioritet, ble ikke glemt passord implementert i *BeiteSpor*. Dersom en skal gjøre webapplikasjonen klar for produksjon må en legge til mulighet for å klikke på glemt passord, og få sendt en link på epost. På den nåværende webapplikasjonen har ikke brukerne mulighet til å gå på innstillinger for å endre passord, selv om de ikke har glemt det. Et alternativ kan være å lage en side som inneholder brukerinnstillinger, der de eksempelvis kan endre passord, navn, epost og andre tilpassinger av webapplikasjonen.

21.3 Integrering av sporingsenhet

Ettersom de fleste av testpersonene brukte sporingsenheter til å holde kontroll på sauene, kan en integrere dataene fra sporingsenhetene med de manuelle dataene som ble samlet inn på tilsynsturene. På denne måten får en samlet dataene på en plass, og en slipper å navigere mellom den digitale løsningen til sporingsenheten og *BeiteSpor*. Et filter kan legges inn i webapplikasjonene, der visning av sporingsenhetene kan slås av og på. Applikasjonen kan vise data fra manuelle tilsynsturer og sporingsenheter, som en kombinasjon eller hver for seg.

21.4 Sauer på kart

På kartvisningen kan en se både saueflokker som er registrert av bonden, og døde sauer som er integrert fra Rovbase. Et par testpersoner kommenterte at de slet med å se forskjell på hva som var deres saueflokk, og hva som var hentet fra Rovbase. En forbedring av *BeiteSpor* er å tydeliggjøre hvilke ikoner som hører til egne registreringer og hvilke som er hentet fra rovbase. Videre når en klikket på en saueflokk som var registrert av bonden, fikk en opp ekstra informasjon. Her ønsket testpersonene å få mer spesifikk informasjon, som fortalte hvor mange søyer og lam det ble registret, og eventuelt hvor mange slipsfarger som var registrert. Et alternativ er å automatisk regne ut hvor mange lam det skal være, basert på slipsfarger, og sammenligne dette med hvor mange lam som faktisk ble registrert. I følge brukertestene, kommenterte testpersonene at det var lammene som var viktigst å ha kontroll på, og de som var mest sårbare.

21.5 Integrasjon mot Sauekontrollen

Litteratursøk og resultater fra brukertesten viser at en stor andel av sauebøndene er medlem av Sauekontrollen. Beskrevet i dagens situasjon i kapittel 4.3, var rundt 40 % av alle sauebønder medlem av Sauekontrollen. I tillegg var samtlige av testpersonene medlem av Sauekontrollen. En integrasjon i *BeiteSpor* mot Sauekontrollen hadde sammenslått et ledd for innrapportering fra tilsynsturer. Diskutert i kapittel 15.3, har målgruppen for applikasjonen liten teknisk erfaring. Redusere antall ledd for innrapportering av tilsynsturer vil gjøre det enklere for brukerne. Sauekontrollen har egne tilsynsrapporter som kunne vært nyttig å integrert i webapplikasjonen, slik at rapportene kunne fylles automatisk inn i Sauekontrollen.

21.6 Implementere beitelag

Det er vanlig at bøndene organiserer seg i beitlag for tilsyn av sauene som er på utmarksbeite. Fordelene med å organisere seg i beitelag er at ressursene samles, og bøndene i beitelaget kan fordele arbeidet med tilsynsturer. For å få tilskudd for organisert beitebruk er det krav om at bonden er en del av et beitelag, beskrevet i kapittel 4.1. I dagens versjon av *BeiteSpor* er tilsynsturene kun knyttet til en bruker. En forbedring av webapplikasjonen kan være å opprette et beitelag, som består av flere brukere. Brukerne som er en del av et beitelaget kan registrere tilsynsturer som er gjennomført for beitelag, og kan se alle

gjennomførte turer i applikasjonen. En slik løsning kan gi grunnlag for et bedre samarbeid blant bøndene.

21.7 Legge inn beiteområde

I brukertesten kom det frem at noen av testpersonene var kun opptatt av å bruke kartet til å finne ut om sauene holdt seg innenfor sitt eget beiteområde. En forbedring kan være å gi brukeren mulighet til å legge inn beiteområde i *BeiteSpor*, slik at en enklere kan se om sauene er utenfor området. Videre kan det være mulig å legge inn et varsel på markørene til saueflokkene dersom de blir observert utenfor det valgte beiteområde.

21.8 Testing

Flere av testpersonene kommenterte at det var viktig at webapplikasjonen var brukervennlig for personer med lite teknologisk erfaring. Det ble gjennomført en iterasjon med fem brukertester på webapplikasjonen. Selv om testpersonene var i en alder fra trettiårene til sekstiårene, burde en teste på flere personer, ettersom det omtrent var en testperson per tiår. Etter at utviklerne hadde fikset tilbakemeldingene som ble gitt på den første iterasjonen med brukertest, burde det blitt utført en til brukertest, for å kvalitetssikre produktet slik at det er enkelt og intuitivt å bruke for bønder.

Referanser

[1] *Beitestatistikk*. nb. URL: https://beitestatistikk.nibio.no/nb/norge (sjekket 27.04.2022).

[2] Steen A.G. og Ådnanes S. *Manuell oppfølging av sau, forprosjekt*. Forprosjekt "Datateknologi, fordypningsprosjekt". 2021.

[3] *Forskrift om tilskott til organisert beitebruk - Lovdata*. URL: https://lovdata.no/dokument/LTI/forskrift/2003-04-01-427?fbclid=IwAR2SjsFrvIKu-j_EEIXwH22A97D_QJ6zcYAxGxn1U30Sv8fm8WJjM8Je6f8 (sjekket 28.04.2022).

[4] *Forskrift om velferd for småfe - Lovdata*. URL: https://lovdata.no/dokument/SF/forskrift/2005-02-18-160#KAPITTEL_4 (sjekket 20.04.2022).

[5] *Tilsynsturer - Norsk Sau og Geit*. no. Aug. 2015. URL: https://www.nsg.no/oppland/lokallag/favang-sau-og-geit/nyheter/tilsynsturer (sjekket 28.04.2022).

[6] *Rovbase*. URL: https://www.rovbase.no/erstatning/sau (sjekket 26.04.2022).

[7] *Rovdata - Jerv - Bestandsstatus*. URL: https://rovdata.no/Jerv/Bestandsstatus.aspx (sjekket 26.04.2022).

[8] *Sauekontrollen*. nb-NO. URL: https://www.animalia.no/no/Dyr/husdyrkontrollene/sauekontrollen/ (sjekket 26.04.2022).

[9] *Gardsbruk, jordbruksareal og husdyr*. nn. URL: https://www.ssb.no/jord-skog-jakt-og-fiskeri/jordbruk/statistikk/gardsbruk-jordbruksareal-og-husdyr (sjekket 26.04.2022).

[10] *Tilskot til drift av beitelag (OBB)*. nn. URL: https://www.landbruksdirektoratet.no/nn/ordningar/jordbruk/tilskot-til-drift-av-beitelag-obb (sjekket 28.04.2022).

[11] *Brukerveiledning*. URL: https://www.animalia.no/no/Dyr/husdyrkontrollene/sauekontrollen/brukerveiledning (sjekket 09.05.2022).

[12] *Bjelleslips - Norsk Sau og Geit*. no. Apr. 2011. URL: https://www.nsg.no/beitebruk/utmarksbeite/merking-av-smafe/bjelleslips/ (sjekket 10.05.2022).

[13] Telespor AS. *Telespor Produkter*. URL: https://telespor.no/produkt/ (sjekket 15.03.2022).

[14] *Radiobjella - Telespor*. URL: https://nettbutikk.telespor.no/products/radiobjella-inkl-batteri (sjekket 10.05.2022).

[15] *Telespor*. nb-NO. URL: https://telespor.no/informasjon/telespor-systemet/ (sjekket 10.05.2022).

[16] *Findmy — GPS sporing av husdyr på utmarksbeite - uten mobildekning*. URL: https://www.findmy.no/ (sjekket 10.05.2022).

[17] *Produkt*. nb. URL: https://www.findmy.no/nb/produkt (sjekket 10.05.2022).

[18] Findmysheep AS. *FindMy Funksjoner*. URL: https://www.findmy.no/nb/funksjoner (sjekket 15.03.2022).

[19] *Nyhet! Smartbjella 2 - opptil 10år batteri**. nb-NO. URL: https://smartbjella.no/ (sjekket 22.03.2022).

[20] *Smartbjella - om oss*. nb-NO. URL: https://smartbjella.no/om-oss/ (sjekket 10.05.2022).

[21] *Smartbjella - Bestill*. nb-NO. URL: https://smartbjella.no/butikk/ (sjekket 10.05.2022).

[22] *Smartbjella - produkt*. nb-NO. URL: https://smartbjella.no/produkt/ (sjekket 10.05.2022).

[23] *Beitesnap - Revolusjonerende verktøy for husdyr på beite!* nb-NO. URL: https://www.beitesnap.no/ (sjekket 25.04.2022).

[24] Ola Tande. *Enklere tilsyn med BeiteSnap*. nb-no. Section: nyheter. Mai 2017. URL: https://www.gardsdrift.no/enklere-tilsyn-med-beitesnap/184792 (sjekket 25.04.2022).

[25] Wark J.D. mfl. *Monitoring the Behavior and Habitat Use of Animals to Enhance Welfare using the ZooMonitor App*. en. URL: https://www.animalbehaviorandcognition.org/article.php?id=1175 (sjekket 21.03.2022).

[26] *Waterfall Methodology - A Complete Guide — Adobe Workfront*. URL: https://www.workfront.com/project-management/methodologies/waterfall (sjekket 30.03.2022).

[27] Atlassian. *Is the Agile Manifesto Still a Thing?* en. URL: https://www.atlassian.com/agile/manifesto (sjekket 09.05.2022).

[28] Atlassian. *Get started with agile project management*. en. URL: https://www.atlassian.com/agile/project-management (sjekket 09.05.2022).

[29] Atlassian. *Get started building an agile workflow*. en. URL: https://www.atlassian.com/agile/project-management/workflow (sjekket 09.05.2022).

[30] *What is Scrum?* en. URL: https://www.scrum.org/resources/what-is-scrum (sjekket 30.03.2022).

[31] Atlassian. *Scrum - what it is, how it works, and why it's awesome*. en. URL: https://www.atlassian.com/agile/scrum (sjekket 09.05.2022).

[32] Atlassian. *Sprints*. en. URL: https://www.atlassian.com/agile/scrum/sprints (sjekket 09.05.2022).

[33] *What Is Kanban? Explained in 10 Minutes — Kanbanize*. URL: https://kanbanize.com/kanban-resources/getting-started/what-is-kanban (sjekket 31.03.2022).

[34] Atlassian. *Kanban - A brief introduction*. en. URL: https://www.atlassian.com/agile/kanban (sjekket 09.05.2022).

[35] Atlassian. *What is a Kanban Board?* en. URL: https://www.atlassian.com/agile/kanban/boards (sjekket 09.05.2022).

[36] Atlassian. *Working with WIP limits for kanban*. en. URL: https://www.atlassian.com/agile/kanban/wip-limits (sjekket 09.05.2022).

[37] *What's the Agile Methodology and How Can It Benefit Your Enterprise?* en. URL: https://www.wearemarketing.com/blog/what-is-the-agile-methodology-and-what-benefits-does-it-have-for-your-company.html (sjekket 31.03.2022).

[38] *Kanban vs. Scrum boards: 11 major differences.* en-US. Jun. 2017. URL: https://miro.com/blog/scrum-kanban-boards-differences/ (sjekket 01.04.2022).

[39] *Trello.* URL: https://trello.com (sjekket 01.04.2022).

[40] Atlassian. *What is Agile?* en. URL: https://www.atlassian.com/agile (sjekket 30.03.2022).

[41] *What is Use Case Diagram?* URL: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/ (sjekket 09.05.2022).

[42] Kurt Bittner og Ian Spence. *Use Case Modeling.* en. Google-Books-ID: zvxfXvEcQjUC. Addison-Wesley Professional, 2003. ISBN: 978-0-201-70913-1.

[43] *Om Rovdata.* URL: https://rovdata.no/Rovdata/OmRovdata.aspx (sjekket 28.03.2022).

[44] Geeksforgeeks. *Functional vs Non Functional Requirements.* URL: https://www.geeksforgeeks.org/functional-vs-non-functional-requirements/ (sjekket 25.02.2022).

[45] Owen Phelps. *Guide to non-functional requirements: types and examples.* URL: https://www.boxuk.com/insight/guide-to-non-functional-requirements-types-and-examples/ (sjekket 02.03.2022).

[46] Indeed Editorial Team. *9 Nonfunctional Requirements Examples.* URL: https://www.indeed.com/career-advice/career-development/non-functional-requirements-examples (sjekket 02.03.2022).

[47] *React – A JavaScript library for building user interfaces.* en. URL: https://reactjs.org/ (sjekket 21.04.2022).

[48] *Angular.* URL: https://angular.io/ (sjekket 21.04.2022).

[49] *Vue.js - The Progressive JavaScript Framework — Vue.js.* en-US. URL: https://vuejs.org/ (sjekket 21.04.2022).

[50] *Stack Overflow Trends.* URL: https://insights.stackoverflow.com/trends?tags=angular%2Creactjs%2Czurb-foundation%2Csemantic-ui%2Cjquery%2Cvue.js%2Cember.js%2Cbackbone.js (sjekket 31.05.2022).

[51] *JavaScript With Syntax For Types.* en. URL: https://www.typescriptlang.org/ (sjekket 21.04.2022).

[52] *MUI: The React component library you always wanted.* en. URL: https://mui.com/ (sjekket 21.04.2022).

[53] *Leaflet — an open-source JavaScript library for interactive maps.* en. URL: https://leafletjs.com/SlavaUkraini/index.html (sjekket 21.04.2022).

[54] *Google Maps Platform Documentation — Maps JavaScript API — Google Developers.* en. URL: https://developers.google.com/maps/documentation/javascript (sjekket 21.04.2022).

[55] *Leaflet vs. Google Maps feature and pricing comparison - Wappalyzer.* URL: https://www.wappalyzer.com/compare/leaflet-vs-google-maps (sjekket 21.04.2022).

[56] *Fielding Dissertation: CHAPTER 5: Representational State Transfer (REST)*. URL: https://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm (sjekket 03.05.2022).

[57] *What is REST*. en-US. URL: https://restfulapi.net/ (sjekket 03.05.2022).

[58] *What is ASP.NET Core? — .NET*. en-US. URL: https://dotnet.microsoft.com/en-us/learn/aspnet/what-is-aspnet-core (sjekket 22.04.2022).

[59] *Overview of Entity Framework Core - EF Core*. en-us. URL: https://docs.microsoft.com/en-us/ef/core/ (sjekket 22.04.2022).

[60] Rick-Anderson. *Introduction to Identity on ASP.NET Core*. en-us. URL: https://docs.microsoft.com/en-us/aspnet/core/security/authentication/identity (sjekket 24.03.2022).

[61] *Bli kjent med Azure — Microsoft Azure*. nb. URL: https://azure.microsoft.com/nb-no/overview/ (sjekket 22.04.2022).

[62] *Azure-databaser – databasetyper på Azure — Microsoft Azure*. nb. URL: https://azure.microsoft.com/nb-no/product-categories/databases/ (sjekket 22.04.2022).

[63] *Azure SQL Database – Administrert skydatabasetjeneste — Microsoft Azure*. nb. URL: https://azure.microsoft.com/nb-no/products/azure-sql/database/ (sjekket 11.05.2022).

[64] *Connect and query - Azure SQL Database & SQL Managed Instance*. en-us. URL: https://docs.microsoft.com/en-us/azure/azure-sql/database/connect-query-content-reference-guide (sjekket 11.05.2022).

[65] *App Service – bygg og drift nettapper — Microsoft Azure*. nb. URL: https://azure.microsoft.com/nb-no/services/app-service/ (sjekket 22.04.2022).

[66] Jakob Nielsen. *10 Usability Heuristics for User Interface Design*. en. URL: https://www.nngroup.com/articles/ten-usability-heuristics/ (sjekket 05.04.2022).

[67] *Web Content Accessibility Guidelines (WCAG) 2.0*. URL: https://www.w3.org/TR/WCAG20/ (sjekket 04.04.2022).

[68] *Testprosedyrar for nettstader — Tilsynet for universell utforming av ikt*. nn. URL: https://www.uutilsynet.no/regelverk/testprosedyrar-nettstader/709 (sjekket 04.04.2022).

[69] *UniColor Pro — What is Color Universal Design? — EIZO*. URL: https://www.eizoglobal.com/products/coloredge/unicolor_pro/index2.html (sjekket 05.04.2022).

[70] *Fargeblind / fargesvak*. URL: https://www.blindeforbundet.no/oyehelse-og-synshemninger/fargeblindhet (sjekket 05.04.2022).

[71] *What is Prototyping?* en. URL: https://www.interaction-design.org/literature/topics/prototyping (sjekket 06.04.2022).

[72] *What is Paper Prototyping?* en. URL: https://www.interaction-design.org/literature/topics/paper-prototyping (sjekket 05.04.2022).

[73] *Adobe XD Pricing — XD Trial & Plan Info — Adobe XD*. en. URL: https://www.adobe.com/no/products/xd/pricing/individual.html (sjekket 06.04.2022).

[74] *Pricing for Figma's Free, Professional, and Organization plans.* en. URL: https://www.figma.com/pricing/ (sjekket 06.04.2022).

[75] Qing Li og Yu-Liu Chen. *Modeling and Analysis of Enterprise and Information Systems*. en. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. ISBN: 978-3-540-89555-8 978-3-540-89556-5. DOI: 10.1007/978-3-540-89556-5. URL: http://link.springer.com/10.1007/978-3-540-89556-5 (sjekket 17.03.2022).

[76] Google Maps Platform. *Map and Tile Coordinates*. URL: https://developers.google.com/maps/documentation/javascript/coordinates (sjekket 04.03.2022).

[77] *kartprojeksjon*. nb. Feb. 2021. URL: http://snl.no/kartprojeksjon (sjekket 27.05.2022).

[78] CFIS. *Google Maps Deconstructed*. URL: https://cfis.savagexi.com/2006/05/03/google-maps-deconstructed/ (sjekket 04.03.2022).

[79] *Kartgrunnlag Fastlands-Norge*. nn-NO. URL: https://kartverket.no/api-og-data/kartgrunnlag-fastlands-norge (sjekket 18.03.2022).

[80] Øyvind Stranna Larsen. *Bonitet*. URL: https://snl.no/bonitet (sjekket 11.03.2022).

[81] institutt for bioøkonomi Norsk. *Bonitetfarger*. URL: https://wms.nibio.no/cgi-bin/ar250?version=1.1.1&service=WMS&request=GetLegendGraphic&format=image/png&layer=Bonitet (sjekket 11.03.2022).

[82] Open Geospatial Consortium. *Web Map Service*. URL: https://www.ogc.org/standards/wms (sjekket 10.03.2022).

[83] Mangesh Nichat. «Landmark based shortest path detection by using A* Algorithm and Haversine Formula». I: (apr. 2013). (Sjekket 18.03.2022).

[84] Google Maps. *Class LatLngBounds*. URL: https://googlemaps.github.io/v3-utility-library/classes/_googlemaps_jest_mocks.latlngbounds.html (sjekket 14.03.2022).

[85] HakanL. *WkHtmlToPdf-DotNet*. URL: https://github.com/HakanL/WkHtmlToPdf-DotNet (sjekket 16.03.2022).

[86] Miljødirektoratet. *Rovbase*. URL: https://rovbase.no/om (sjekket 15.03.2022).

[87] Google Maps Platform. *Heatmap layer*. URL: https://developers.google.com/maps/documentation/javascript/heatmaplayer (sjekket 16.03.2022).

[88] Michael D. Myers og David Avison. *Qualitative Research in Information Systems: A Reader*. en. Google-Books-ID: NzrlEZMflZ0C. SAGE, mai 2002. ISBN: 978-1-4462-7581-8.

[89] World Leaders in Research-Based User Experience. *Usability Testing 101*. en. URL: https://www.nngroup.com/articles/usability-testing-101/ (sjekket 25.04.2022).

[90] James R. Lewis. «Usability Testing». en. I: *Handbook of Human Factors and Ergonomics*. Section: 49 _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/0470048204.ch49. John Wiley & Sons, Ltd, 2006, s. 1275–1316. ISBN: 978-0-470-04820-7. DOI: 10.1002/0470048204.ch49. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/0470048204.ch49 (sjekket 25.04.2022).

[91] Jakob Nielsen. *Why You Only Need to Test with 5 Users*. en. URL: https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/ (sjekket 19.05.2022).

[92] *What is Usability Testing?* en. URL: https://www.interaction-design.org/literature/topics/usability-testing (sjekket 26.04.2022).

[93] World Leaders in Research-Based User Experience. *Remote Moderated Usability Tests: How to Do Them*. en. URL: `https://www.nngroup.com/articles/moderated-remote-usability-test/` (sjekket 26.04.2022).

Tillegg A

Brukertester

A.1 Brukertest A

Brukertest av masteroppgave

Bakgrunn

I forprosjektet til masteroppgaven høsten 2021 ble det laget en mobilapplikasjon, som skal være et verktøy for bonden når han/hun er ute på tilsynstur av sau. I applikasjonen kan bonden registrere forskjellig karakteristikk ved saueflokker, som posisjon, antall sauer i forskjellig farger og størrelser, slipsfarger og øremerker. Etter turen er ferdig kan turen lastes opp i en skytjeneste som samler alle turene.
I masteroppgaven har vi laget werbapplikasjonen som samler inn dataene fra tilsynsturer med tilhørende visning i kart. Dataene som ligger inne i applikasjonen fra tilsynstur er generert av oss, og er ikke reell data.

Scenario:
Over en lengre periode har du vært ute på tilsynsturer, som har blitt lastet opp til applikasjonen. Du skal nå gå inn i den nye webapplikasjonen for å se data fra de gjennomførte tilsynsturene.

Før vi starter:
Dette er ikke en test av deg, men en test av applikasjonen.
Foretrekker du å være anonym, eller kan vi nevne navn dersom det skulle være nødvendig?

Bakgrunnsinformasjon:
Giftet seg med en bonde for 17 år siden, og drevet bondegård hele tiden. Har 130 sau per dags dato, men hadde på det meste 250. Driver også med storfe. Har fjellbeite, og har startet opp igjen med fjellbeite. Har brukt sporing på fjellbeite. Bruker Smartbjella, og har sauene på utmarksbeite vest i hardangervidden.

Oppgave (Velg periode og rovdyr)

1. Velg en ny periode fra 12. juni til 1. august.
2. Filtrer rovdyr, slik at du bare viser jerv i kartet.
3. Kan du finne ekstra informasjon om hendelse av jerv i kartet.
4. Velg i tillegg bjørn og ulv for å vise i kartet.

Svar:
Gikk fint å velge periode, og skjønte at hen måtte trykke på søk.
Greide fint å filtrere på rovdyr.
Syns det var litt vanskelig å skjønne hva som var rovdyr hendelser på kartet.

Spørsmål

1. Kommer det tydelig frem hvilke rovdyr som hører til hvert ikon? Noe som kunne vært annerledes?

   Forstod ikke at ikonet for sau som var blått var knyttet til Jerv. Koblet ikke at øye-ikonet var rovdyr. Det ville hen skjønt etter ganske kort bruk.
2. Er det tilhørende ikonet inni markøren forståelig?

   Det var litt vanskelig å skjønne hva som var rovdyr

Oppgave (Oversikt over Tilsynsturer)

1. Finn oversikt over alle tilsynsturene du har gått i den valgte perioden
2. Finn ekstra informasjon om en saueflokk

Svar:
Var litt vanskelig å finne ut hva som var hver tur.
Forstod ikke helt forskjellen mellom hva som var rovdyr og registrerte saueflokker.

Spørsmål

1. Kommer det tydelig frem hva tallet på en saueflokk betyr?

   Det var antall dyr, som hen antok tidlig.
2. Savnet du noe informasjon når du klikket på en saueflokk?

   Savner registrerte slips på hver flokk.

Oppgave (Måned- og ukesvisning med heatmap og saueflokk)

1. Prøv å se på bevegelsene på sauene fra uke til uke i den valgte perioden.
2. Gjør det samme igjen, men bruk da heatmap som visning.
3. Prøv å se på bevegelsene på sauene fra måned til måned i den valgte perioden, med å bruke først saueflokker og deretter heatmap.

Svar:
Hen sliter litt med å få oversikt.
Hen prøver å trykke på en tur i oversikten for å velge en tur, men det fungerer ikke.
Hen skjønte til slutt av seg at hen måtte trykke på neste- eller forrige-knappen.
Det hadde vært nyttig å hatt med grenseområde for beite, og ID-en til sauen i tillegg. Det hadde vært stor verdi å kunne registrere nummeret på bjella.

Spørsmål

1. Hvilke av disse to visningene syns du var best?
   a.

2. Var det noe forskjell på hvilken visning (saueflokk eller heatmap) som var best på om det var ukes eller månedsvisning?
    Foretrakk saueflokk-visningen fremfor heatmap. Det viktigste er å vite om de er innenfor beiteområde, slik at de ikke går utenfor beite.
3. Kan denne funksjonaliteten hjelpe å holde saueflokkene vekke fra områder med rovdyr?
    Det gjelder ikke så mye for hen. Hen tipper at svaret er nei, men hen sier at hen kanskje ikke rett person å spørre, siden de har lite problem med

Oppgave (Bonitet og PDF)

1. Sett bonitet på kartet, med synlighet 0,7.
2. (Vi viser en PDF som de liksom har lastet ned)

svar:
Hen skjønte fort hvordan hen valgte bonitet i kartet og endre synlighet.

Spørsmål

1. Hvilken type rapportering gjør du til staten? Hva skal med i denne rapporten?
    Sender søknad om produksjonstilskudd. antall fødte lam, slakta lam. Bruker verktøyet Sauekontroll. Sender inn del 1 på vinter/vår og del 2 på høsten.
2. Hjelper bonitet deg til å forstå hvorfor sauene beveger seg slik de gjør?
    a. Kommer det tydelig frem hva de ulike bonitetsfargene betyr?
    Det er ofte kjente områder, så de vet egentlig fra før av hva som er gode beiteområder. Det kan kanskje være nyttig hvis det er et nytt beiteområde.

Samling Spørsmål

1. Hva bruker du nå for å holde kontroll på sauene dine? Bruker du en digital løsning, eller foregår alt på papir?
    Bruker både Smartbjella og manuelt gjennom tilsynsturer. Registrer posisjon og nummer på sauene gjennom en manuell oppsynstur. Hen bruker informasjonen til å vite hvilket sauer som ikke går ute av beiteområde.
2. Har du registrer sau manuelt tidligere?
    a. Registrer du posisjon?
    b. antall sau, slips og øremerke?
    c. Bruker du denne informasjonen til noe senere? Hva?
    d. Dersom du kunne sett alt dette visuelt på en nettside i etterkant (våres applikasjon), hadde det vært til nytte?
3. Tror du denne løsningen med en slik visning hjelpe deg til å bedre forestå hvordan saueflokkene beveger seg på?
    Det er viktig at løsningen ikke trenger å ha dekning, og kunne laste opp når man har dekning. Hen bruker et skjema som er laget i excel. Vil ha med individopplysninger. Bruker gjerne mat som gjør at de kommer nærmere, slik at de kan registrere flere ting.
4. Var det noe som manglet for å kunne se på bevegelsene til sauene?

    a. Er det noe tilleggsinformasjon av saueflokkene eller en annen type visning som kunne gjort det bedre?
       Sauene går ikke i statiske flokker, derfor er individnummer er viktig. Mulighet til å legge inn andre observasjoner. F.eks hvis det er tvillinger.
5. Kan løsningen gjøre jobben din som sauebonde enklere?
    a. Kunne brukt et elektronisk verktøy. Må fungere uavhengig av mobildekning. Gjerne helt ned på individnivå. For det kan være nyttig å kunne følge et individ fra tur til tur.
    b. I Smartbjella bruker hen mest individ visningen, men bruker også heatmap.

A.2 Brukertest B

Brukertest av masteroppgave

Bakgrunn

I forprosjektet til masteroppgaven høsten 2021 ble det laget en mobilapplikasjon, som skal være et verktøy for bonden når han/hun er ute på tilsynstur av sau. I applikasjonen kan bonden registrere forskjellig karakteristikk ved saueflokker, som posisjon, antall sauer i forskjellig farger og størrelser, slipsfarger og øremerker. Etter turen er ferdig kan turen lastes opp i en skytjeneste som samler alle turene.
I masteroppgaven har vi laget werbapplikasjonen som samler inn dataene fra tilsynsturer med tilhørende visning i kart. Dataene som ligger inne i applikasjonen fra tilsynstur er generert av oss, og er ikke reell data.

Scenario:
Over en lengre periode har du vært ute på tilsynsturer, som har blitt lastet opp til applikasjonen. Du skal nå gå inn i den nye webapplikasjonen for å se data fra de gjennomførte tilsynsturene.

Før vi starter:
Dette er ikke en test av deg, men en test av applikasjonen.
Foretrekker du å være anonym, eller kan vi nevne navn dersom det skulle være nødvendig?
Kan ta med alt du vil!!

Bakgrunn:
Hen er 33 år, gift og hatt gård i 12 år. Har 3 barn.
Gården er 100 mål, og hen har 7 ammekyr og 150 sau. Fleire tusen mål med fjellbeite.
Sender sauene til fjells på hardangervidda. fra sankthans til midten av november.
Hen har erfaring med datamaskin

Oppgave (Velg periode og rovdyr)

1. Velg en ny periode fra 12. juni til 1. august.
2. Filtrer rovdyr, slik at du bare viser jerv i kartet.
3. Kan du finne ekstra informasjon om hendelse av jerv i kartet.
4. Velg i tillegg bjørn og ulv for å vise i kartet.

Svar:
Klarer fint å velge periode.
Klarer fint å filtrere jerv.
Litt usikker på kor hen finne informasjon om ein jerv.

Spørsmål

1. Kommer det tydelig frem hvilke rovdyr som hører til hvert ikon? Noe som kunne vært annerledes?

   Syns fargekodene var greie, men alle de tallene som står på klyngene var hen mer usikker på. Forstod dette etter Anders forklarte. Syns det var likt som Rovbase, så det gidde mening.

2. Er det tilhørende ikonet inni markøren forståelig?

   Syns dette er forståelig. Men hadde kanskje hatt problemer dersom ingen hadde forklart dette.

Oppgave (Oversikt over tilsynsturer)

1. Finn oversikt over alle tilsynsturer du har gått i den valgte perioden
2. Finn ekstra informasjon om en saueflokk

Svar:
Gjekk først på årsrapport. Deretter gikk hen på sau, men var litt usikker her.
Spørsmål fra hen: “er det kun bonden som kan registrere, eller hvem som helst”

Spørsmål

1. Kommer det tydelig frem hva tallet på en saueflokk betyr?

   Nei. lurer på om det er antall voksne dyr, eller om det er alle dyrene (lam + sau).

2. Savnet du noe informasjon når du klikket på en saueflokk?

   Grunnen til at en har oppsyn er for å finne et eksakt antall. Jerv dreper ofte lam. Så når ein har tilsyn kan en godt telle de voksne sauene. Viktig at en teller antall lam og antall voksne. Dersom en har 5 sauer, og 3 lam, så seie det at det er noe gale. Jerven gjemmer ofte lammene, så en må gå over store områder før en finner de.

Oppgave (Måned- og ukesvisning med heatmap og saueflokk)

1. Prøv å se på bevegelsene på sauene fra uke til uke i den valgte perioden.
2. Gjør det samme igjen, men bruk da heatmap som visning.
3. Prøv å se på bevegelsene på sauene fra måned til måned i den valgte perioden, med å bruke først saueflokker og deretter heatmap.

Svar:
Forstår ikke hvordan hen ser på bevegelsen fra uke til uke. Anders må forklare at her må hen klikke på neste og forrige for å navigere.
Forstår fint hvordan hen skifter til heatmap.

Spørsmål

1. Hvilke av disse to visningene syns du var best?
   Absolutt saueflokken. Der fekk du opp antallet, og følte det ble mer oversiktlig. Hadde en bedre oversikt over hva en faktisk hadde sett
2. Var det noe forskjell på hvilken visning (saueflokk eller heatmap) som var best på om det var ukes eller månedsvisning?
   Ukesvisning ble mer oversiktlig, men samtidig så er det kjekt å se i fra måned til måned, hvor går sauene. Går de mest på et tidspunkt tidlig i sesongen? Månedsvisning er best på heatmap. Ukesvisning sa ikke så mye, si det var så få punkter.
3. Kan denne funksjonaliteten hjelpe å holde saueflokkene vekke fra områder med rovdyr?
   Det er fare for jerv på hardangervidden. Har hatt jerv 2 år på hardangervidda. Dersom hen hadde sitt jerv så hadde hen ikke flyttet sauene. Dersom jerven dreper mange av dyrene, så blir de tatt av beite og med hjem. To år på rad mistet hen 30% av lammene sine. Gikk tilsynstur og fant flere lam som var døde. Fant 25-30 lam som var død. Vet ikke om det er rovdyrikonene hen hadde brukt nettsiden til. Tenker kanskje at det er mer aktuelt i midt-norge der det er mange rovdyr. Stoler ikke helt på dei som legger inn data på rovbase.

Oppgave (Bonitet og PDF)

1. Sett bonitet på kartet, med synlighet 0,7.
2. (Vi viser en PDF som de liksom har lastet ned)

Svar:
Litt usikker på hvordan en setter synlighet på 0.7, men finner det ut relativ fort.

Spørsmål

1. Hvilke type rapportering gjør du til staten? Hva skal med i denne rapporten?
   Er med i sauekontrollen. Periodevis veldig flink å registrere her. Begynner med årssyklusen med januar og februar, og legger inn lysing (ultralyd) og så registrerer en lamming og paring. Dersom en er flink til å registrere lamming så har en alle individer operative. Da bruker en å registrere beite og binge, alle individer en sender til fjells. Får generert lister av hvilke sauer en har i fjellet. Vet at en kan lage tilnærma lik rapport som det vi har i sauekontrollen. For min del dersom jeg skulle brukt et program som dette, ville jeg registrert på slips istedenfor fargene. Vil ha gul, grønn og rød slips farge i rapporten istedenfor hvit, brun og svart sau. Uansett hvor lite innhegning sauene går på, så klarer en ikke å lese individnummer. En er nødt til å observere ut i fra noe, og da er det lettere med slips. Har 149 kvite og 1 svart. Har slips på dei som går med 1 lam og 3 lam. Mener at en skal være 7,5 meter i fra sauen for å kunne lese individnummer.
2. Hjelper bonitet deg til å forstå hvorfor sauene beveger seg slik de gjør?
   a. Kommer det tydelig frem hva de ulike bonitetsfargene betyr?

Ja, det hjelper. Får en bedre oversikt av hvor en er i verden. Ser lettere hva det er en ser på, og hvilke type område det er. Det er enkelt å se fargene på bonitet, men dette er fordi hen kjenner det fra før. Hen kan det utenom.

Samling Spørsmål

1. Hva bruker du nå for å holde kontroll på sauene dine? Bruker du en digital løsning, eller foregår alt på papir?

   Når vi går tilsynsturer i fjellet så ser en mer etter hva som kan være galt. Lam som går uten moren. Men det blir rent visuelt. Det går ikke ann å telle. Det går over for store områder. Ser kun 10% av sauene på utmarksbeite. Men på et beite hjemme, med fluemark og sjukdommer, er en helt nødt til å telle. For min del hadde denne applikasjonen vært helt ideelt. Hen memorerer antall sauer i hovedet, og det er fort for å bli i tvil. Skriver ikke ned noe og ser ikke på noe tidligere historikk. Skriver ned hvor mange dyr hen har på hver plass, og dette blir notert på notater på telefonen. Ser absolutt for meg at denne applikasjonen hadde vært noe å testet ut. Dette må være en enkel app. Det bønder liker minst er som regel noe med data og teknologi. Følte nettsiden var veldig enkel.
2. Har du registrer sau manuelt tidligere?
   a. Registrer du posisjon?
   b. antall sau, slips og øremerke?
   c. Bruker du denne informasjonen til noe senere? Hva?
   d. Dersom du kunne sett alt dette visuelt på en nettside i etterkant (våres applikasjon), hadde det vært til nytte?
3. Tror du denne løsningen med en slik visning hjelpe deg til å bedre forestå hvordan saueflokkene beveger seg på?

   Tror denne hadde vært veldig interessant å se en oversikt. Hadde plutselig fått et helt bilde over hvor sauene er. Vil gjerne vite hvor sauene går i slutten av sauesesongen, slik at hen vet hvor det skal gjødsles i august.
4. Var det noe som manglet for å kunne se på bevegelsene til sauene?
   a. Er det noe tilleggsinformasjon av saueflokkene eller en annen type visning som kunne gjort det bedre?

      Jeg føler vi må dele det opp i vaksne sau og lam. Totalt antallet på selve ikonet kunne vært det samme, men vil se sau og lam når en klikker på mer informasjon. Det er lammene som betyr mest for de. I en normalsituasjon så ser du den voksne sauen og lammene på siden. Økonomien ligger i lammene.
5. Kan løsningen gjøre jobben din som sauebonde enklere?

   Absolutt. Lurer på om hen kan leggge inn hvor mange sauer hen har på et bestemt området før en går ut på tur. Slik at hen kan sjekke om tallet stemmer, og får opp en prosentvis på hvor mange sauer hen ser på hvert beite. Mener det hadde vært en god ide og integrere mot sauekontrollen.
6. Hvor mye beitemark har du?

A.3 Brukertest C

Brukertest av masteroppgave

Bakgrunn

I forprosjektet til masteroppgaven høsten 2021 ble det laget en mobilapplikasjon, som skal være et verktøy for bonden når han/hun er ute på tilsynstur av sau. I applikasjonen kan bonden registrere forskjellig karakteristikk ved saueflokker, som posisjon, antall sauer i forskjellig farger og størrelser, slipsfarger og øremerker. Etter turen er ferdig kan turen lastes opp i en skytjeneste som samler alle turene.
I masteroppgaven har vi laget werbapplikasjonen som samler inn dataene fra tilsynsturer med tilhørende visning i kart. Dataene som ligger inne i applikasjonen fra tilsynstur er generert av oss, og er ikke reell data.

Scenario:
Over en lengre periode har du vært ute på tilsynsturer, som har blitt lastet opp til applikasjonen. Du skal nå gå inn i den nye webapplikasjonen for å se data fra de gjennomførte tilsynsturene.

Før vi starter:
Dette er ikke en test av deg, men en test av applikasjonen.
Foretrekker du å være anonym, eller kan vi nevne navn dersom det skulle være nødvendig? Går helt fint

Bakgrunn:
Voksen mann på 65år, og har drevet med sau siden hen var 40, og vokst opp med sau. Går masse tilsynsturer. Brukt findMy og Telespor. Smartbjella har eg ikke erfaring med. Bruker både FindMy og telespor, telespor er avhengig av mobildekning, og FindMy bruker GPS.

Oppgave (Velg periode og rovdyr)

1. Velg en ny periode fra 12. juni til 1. august.
2. Filtrer rovdyr, slik at du bare viser jerv i kartet.
3. Kan du finne ekstra informasjon om hendelse av jerv i kartet.
4. Velg i tillegg bjørn og ulv for å vise i kartet.

Svar:
Sliter litt med å finne tidsperiode til venstre.
Klarer fint å filtrere rovdyr.
Lurte litt på hva “Øyet” betydde, men fant det ut når hen klikket på informasjonen.

Spørsmål

1. Kommer det tydelig frem hvilke rovdyr som hører til hvert ikon? Noe som kunne vært annerledes?

Det gjekk fint etter hen hadde testet litt. Det er ikke alltid alt er intuitivt når hen begynner å dra på årene.

2. Er det tilhørende ikonet inni markøren forståelig?

   Sau var greitt, men visste ikke hva øyet betydde. Kjente igjen klyngene, fordi det var sånn på findMy.

Oppgave (Oversikt over oppsynsturer)

1. Finn oversikt over alle oppsynsturer du har gått i den valgte perioden
2. Finn ekstra informasjon om en saueflokk

Svar:
Finner turene fint.
Usikker på om det er rovdyr som vises eller saueflokk.
Filtrerer vekk alle rovdyrene, det gikk fint.

Spørsmål

1. Kommer det tydelig frem hva tallet på en saueflokk betyr?
   Ja det kommer tydelig fram, tenkte seg frem til dette
2. Savnet du noe informasjon når du klikket på en saueflokk?
   Lam som var skadet, og om alle sauene har lammene sine.

Oppgave (Måned- og ukesvisning med heatmap og saueflokk)

1. Prøv å se på bevegelsene på sauene fra uke til uke i den valgte perioden.
2. Gjør det samme igjen, men bruk da heatmap som visning.
3. Prøv å se på bevegelsene på sauene fra måned til måned i den valgte perioden, med å bruke først saueflokker og deretter heatmap.

Svar:
Ser at det er litt bevegelse i kartet, men husker ikke hvordan forrige bildet var. Tenker om det er noe mulighet for å legge spor på kartet. Et problem, en tur gikk en sau alene, neste gang gikk den med en annen sau, dette kan føre til sorteringsproblematikk. Det er ikke sikkert at det er samme flokk fra gang til gang. Må på individnivå på de voksne sauene. Mest stabilt å se på hvem som går i lag.

FInner heatmap fint når hen forstår hva vi sier.
Foreslår at en kan klikke på turene, slik at en slipper å klikke på neste/forrige.
Klarer fint å klikke på måned med heatmap.
Ser hvor det er mest intens

Spørsmål

1. Hvilke av disse to visningene syns du var best?
   “Saueflokk” var det veldig mange, og ikke helt lett å se. Fargekoding var visuelt, det var lett å se hvor det var lite og mye sauer.
   Med “saueflokk” på måned så blir det veldig mange flokker oppå hverandre.
   Med heatmap så var det enklere.
2. Var det noe forskjell på hvilken visning (saueflokk eller heatmap) som var best på om det var ukes eller månedsvisning?
   På uke så var det lettere å bruke “saueflokker”. For voldsomt på måned.
3. Kan denne funksjonaliteten hjelpe å holde saueflokkene vekke fra områder med rovdyr?

Oppgave (Bonitet og PDF)

1. Sett bonitet på kartet, med synlighet 0,7.
2. (Vi viser en PDF som de liksom har lastet ned)

Svar:
Finner fint ut av bonitet.

Spørsmål

1. Hvilke type rapportering gjør du til staten? Hva skal med i denne rapporten?
   Organisert beitelag og felles sammenslutning, slik at hen får litt tilskudd fra offentlige.
   Sender inn antall tilsynsdager + antall dyr slept til fjells, vaksne og lam + hvor mye tap + vaksne og lam.
   Det blir telt opp manuelt hos hver bonde og sendt inn. Sauekontrollen har mye data fra hele sauelivet. Melde inn alle dyr, som er tapt eller som blir slaktet. Melder ut dyrene manuelt. Da skal en få en opptelling, hvor mange en hadde i starten og i slutten.
   Hvor mange en har slept til fjells av vaksne og lam og hvor mange som kommer hjem.
   På en tilsynstur så skriver en kun antall.
   Antall personer som er med blir regnet som antall tilsynsturer.
2. Hjelper bonitet deg til å forstå hvorfor sauene beveger seg slik de gjør?
   a. Kommer det tydelig frem hva de ulike bonitetsfargene betyr?
      Dyrene går der det er mat. Dersom det er gode forhold så er det logisk at de går der som det er god jord. Trur kanskje jeg ville funnet ut litt flere bevis på dette her. På sommeren så går de på plasser en ikke skulle trodd var mulig. Ser tydelig at fargene er forskjellig. Lurte på om en skal bruke fargekodene i menyen.

Samling Spørsmål

1. Hva bruker du nå for å holde kontroll på sauene dine? Bruker du en digital løsning, eller foregår alt på papir?
    a.
2. Har du registrer sau manuelt tidligere?
    a. Registrer du posisjon?
    b. antall sau, slips og øremerke?
    c. Bruker du denne informasjonen til noe senere? Hva?
    d. Dersom du kunne sett alt dette visuelt på en nettside i etterkant (våres applikasjon), hadde det vært til nytte?

        Har skrive ut listen som jeg tar med meg når jeg går på tur. Sauekontrollen genererer en liste som kan skrives ut på papir som hen tar med seg. Her har en individnummer til sauene. Har en app til sauekontrollen, mest til registrering, ikke tilsyn av sauer. Registrerer for å ha oversikt over sauene sine. Dersom det er et lam som mangler, og da registrerer hen dette i sauekontroll.

3. Tror du denne løsningen med en slik visning hjelpe deg til å bedre forestå hvordan saueflokkene beveger seg på?
    a. Tror dette kunne vært til nytte. Alt må være veldig brukervennelig, må være så enkelt som mulig.
    Bruker du Telespor og Findmy sine løsninger?

        Bruker sporingsenhetene stort sett til å se hvor de befinner seg. Ser på beitemønsteret, og hvor har de gått. Noen spor blir til en svart flekk, mens andre har streker på kryss og tvers. Interessant å se på om de vandrer mye eller om de står i ro. Bruker dette til en skal samle inn sauene.
        Det er veldig stor forskjell i bevegelsesmønsteret til sauene.
        Hen har kun kvit sau. 70% av norsk sau er kvit.

4. Var det noe som manglet for å kunne se på bevegelsene til sauene?
    a. Er det noe tilleggsinformasjon av saueflokkene eller en annen type visning som kunne gjort det bedre?
    Lage spor i kartet, hvor sauene går i fra tur til tur. Fordi hen ikke husker forrige tur. Men må på individnivå dersom hen skal ha videre sporing.
5. Kan løsningen gjøre jobben din som sauebonde enklere?
    Dersom dere får til noe som er lett å bruke. Bare taste det inn på telefonen, for da kan dette spare masse arbeid sammen med registrering. Viktig at det er brukervennlig.

A.4 Brukertest D

Brukertest av masteroppgave

Bakgrunn

I forprosjektet til masteroppgaven høsten 2021 ble det laget en mobilapplikasjon, som skal være et verktøy for bonden når han/hun er ute på tilsynstur av sau. I applikasjonen kan bonden registrere forskjellig karakteristikk ved saueflokker, som posisjon, antall sauer i forskjellig farger og størrelser, slipsfarger og øremerker. Etter turen er ferdig kan turen lastes opp i en skytjeneste som samler alle turene.
I masteroppgaven har vi laget werbapplikasjonen som samler inn dataene fra tilsynsturer med tilhørende visning i kart. Dataene som ligger inne i applikasjonen fra tilsynstur er generert av oss, og er ikke reell data.

Scenario:
Over en lengre periode har du vært ute på tilsynsturer, som har blitt lastet opp til applikasjonen. Du skal nå gå inn i den nye webapplikasjonen for å se data fra de gjennomførte tilsynsturene.

Før vi starter:
Dette er ikke en test av deg, men en test av applikasjonen.
Foretrekker du å være anonym, eller kan vi nevne navn dersom det skulle være nødvendig?

Bakgrunnsinformasjon:
Driver med sau, og har rundt 300 sau. Holdt på med sau siden 1995.

Oppgave (Velg periode og rovdyr)

1. Velg en ny periode fra 12. juni til 1. august.
2. Filtrer rovdyr, slik at du bare viser jerv i kartet.
3. Kan du finne ekstra informasjon om hendelse av jerv i kartet.
4. Velg i tillegg bjørn og ulv for å vise i kartet.

Svar:
Fant datovelger med en gang, og klikket søk av seg selv med en gang.
Forstod hvor rovdyr var, og forstod hvordan switchen funket.
Greide å zoome på kartet.
Forstod at hen måtte klikke på en markør for å få opp informasjon.

Spørsmål

1. Kommer det tydelig frem hvilke rovdyr som hører til hvert ikon? Noe som kunne vært annerledes?

   Forstod at fargene hadde betydning for hvilke typer dyr det er.
2. Er det tilhørende ikonet inni markøren forståelig?

Forstod ikke helt hva som var betydningene av de forskjellige ikonene. Etter å ha klikket litt rundt skjønte hen betydningene av ikonene.

Oppgave (Oversikt over tilsynsturer)

1. Finn oversikt over alle tilsynsturer du har gått i den valgte perioden
2. Finn ekstra informasjon om en saueflokk

Svar:
Forstod at turene lå under sau.
Forstod ikke helt at det var i kartet man måtte lete etter ekstra informasjon av en saueflokk.
Tok litt tid, men skjønte til slutt at man måtte trykke på en saueflokk

Spørsmål

1. Kommer det tydelig frem hva tallet på en saueflokk betyr?
   Forstod at det betyr at det var antall sau i flokken.
2. Savnet du noe informasjon når du klikket på en saueflokk?
   Vil ha hvor mange sauer som er voksne og hvor mange som er lam i flokken.

Oppgave (Måned- og ukesvisning med heatmap og saueflokk)

1. Prøv å se på bevegelsene på sauene fra uke til uke i den valgte perioden.
2. Gjør det samme igjen, men bruk da heatmap som visning.
3. Prøv å se på bevegelsene på sauene fra måned til måned i den valgte perioden, med å bruke først saueflokker og deretter heatmap.

Svar:
Forstod med en gang at man måtte trykke på neste og forrige.
Forstod at man måtte switche av saueflokk og på med heatmap med en gang.
Forstod hvor månedsvisningen var.

Spørsmål

1. Hvilke av disse to visningene syns du var best?
   Bedre med saueflokker. Ser både på om de er innenfor et område og hvordan bevegelsene er.
2. Var det noe forskjell på hvilken visning (saueflokk eller heatmap) som var best på om det var ukes eller månedsvisning?
   Det er vanskelig å svare på.
3. Kan denne funksjonaliteten hjelpe å holde saueflokkene vekke fra områder med rovdyr?

Oppgave (Bonitet og PDF)

1. Sett bonitet på kartet, med synlighet 0,7.
2. (Vi viser en PDF som de liksom har lastet ned)

Svar:
Forstod hvor bonitet var plasser.
Og forstod at slideren var for å justere synlighet.

Spørsmål

1. Hvilke type rapportering gjør du til staten? Hva skal med i denne rapporten?
   Ingen rapportering til staten. Bruker sauekontroll. Søker tilskudd på sauene, og sender inn to ganger i året. Sender hvor mange de har på utmarksbeite. Sauekontroll har et eget søknadsskjema.
2. Hjelper bonitet deg til å forstå hvorfor sauene beveger seg slik de gjør?
   a. Kommer det tydelig frem hva de ulike bonitetsfargene betyr?
      Har aldri tenkt på bonitet. Har ikke tenkt på det før. Det kommer ganske greit frem hva fargene på bonitet betyr.

Samling Spørsmål

1. Hva bruker du nå for å holde kontroll på sauene dine? Bruker du en digital løsning, eller foregår alt på papir?
   Bruker findmy og går på tilsynsturer. Greier ikke telle sauene. Det for mange til fjells og det er flere eiere. Går mest for å se om det noe gale.
2. Har du registrer sau manuelt tidligere?
   a. Registrer du posisjon?
   b. antall sau, slips og øremerke?
   c. Bruker du denne informasjonen til noe senere? Hva?
   d. Dersom du kunne sett alt dette visuelt på en nettside i etterkant (våres applikasjon), hadde det vært til nytte?
3. Tror du denne løsningen med en slik visning hjelpe deg til å bedre forestå hvordan saueflokkene beveger seg på?
   Har findmy så har nok informasjon på det. Har ganske god kontroll når man setter bjelle på de yngste sauene. Er ikke helt den personen som kunne brukt en slik løsning, men tenker det kan være til bruk for andre.
4. Var det noe som manglet for å kunne se på bevegelsene til sauene?
   a. Er det noe tilleggsinformasjon av saueflokkene eller en annen type visning som kunne gjort det bedre?
      Usikker
5. Kan løsningen gjøre jobben din som sauebonde enklere?
   Nei, tror ikke det. Har vært på møte for digitalisering, men er ikke så fan av det selv. Hvis det er en findmy som kommer ned, så vil det være flere komme med. Bruker også varsling fra findmy, hvis det for eksempel har ligget i ro. Har ikke brukt urovarsel og heatmap.
6. Hvor mye beitemark har du?

A.5 Brukertest E

Brukertest av masteroppgave

Bakgrunn

I forprosjektet til masteroppgaven høsten 2021 ble det laget en mobilapplikasjon, som skal være et verktøy for bonden når han/hun er ute på tilsynstur av sau. I applikasjonen kan bonden registrere forskjellig karakteristikk ved saueflokker, som posisjon, antall sauer i forskjellig farger og størrelser, slipsfarger og øremerker. Etter turen er ferdig kan turen lastes opp i en skytjeneste som samler alle turene.
I masteroppgaven har vi laget werbapplikasjonen som samler inn dataene fra tilsynsturer med tilhørende visning i kart. Dataene som ligger inne i applikasjonen fra tilsynstur er generert av oss, og er ikke reell data.

Scenario:
Over en lengre periode har du vært ute på tilsynsturer, som har blitt lastet opp til applikasjonen. Du skal nå gå inn i den nye webapplikasjonen for å se data fra de gjennomførte tilsynsturene.

Før vi starter:
Dette er ikke en test av deg, men en test av applikasjonen.
Foretrekker du å være anonym, eller kan vi nevne navn dersom det skulle være nødvendig?
Kan bruke navnet.

Bakgrunn
Tok over gården i 1995, og har 230 sau. Skal få 210 lam.
Beiteområde rett under 1000 mål.

Oppgave (Velg periode og rovdyr)

1. Velg en ny periode fra 12. juni til 1. august.
2. Filtrer rovdyr, slik at du bare viser jerv i kartet.
3. Kan du finne ekstra informasjon om hendelse av jerv i kartet.
4. Velg i tillegg bjørn og ulv for å vise i kartet.

Svar:
Bruker tekstfeltet på datoen.
Greide å finne riktig periode greit.
Forstod at hen måtte trykke på søk.
Greide fint å filtrere på rovdyr.
Slo av de andre rovdyrene kjapt
Forstod at hen måtte klikke på markøren for rovdyr for mer informasjon
FIltrerte greit på de andre rov

Spørsmål

1. Kommer det tydelig frem hvilke rovdyr som hører til hvert ikon? Noe som kunne vært annerledes?

   For meg gjør det det. Hen bruker rovbase til vanlig. Har tilgang til den interne rovbase, og er ansatt i statens naturvernforbund. Er ikke sikker på om det er like forståelig for alle.
2. Er det tilhørende ikonet inni markøren forståelig?

   Forstod alle ikonene.

Liste opp alternativene av ikoner og hva de betyr

Lage hover med lav opacity visning av en flokk

Oppgave (Oversikt over tilsynsturer)

1. Finn oversikt over alle tilsynsturer du har gått i den valgte perioden
2. Finn ekstra informasjon om en saueflokk

Svar:

Finner ikke helt hvor oversikten av alle saueflokkene ligger.
Prøvde først å gå på årsrapport.
Fikk tips om å ta vekk alle rovdyrene

Spørsmål

1. Kommer det tydelig frem hva tallet på en saueflokk betyr?

   Antok at det var antall sauer i flokken
2. Savnet du noe informasjon når du klikket på en saueflokk?

   Det er viktig å vite hvor mange voksne dyr og lam. Det hadde vært bra med informasjon om det er lam eller voksen dyr. Markerer alle voksne dyr med slips.

Oppgave (Måned- og ukesvisning med heatmap og saueflokk)

1. Prøv å se på bevegelsene på sauene fra uke til uke i den valgte perioden.
2. Gjør det samme igjen, men bruk da heatmap som visning.
3. Prøv å se på bevegelsene på sauene fra måned til måned i den valgte perioden, med å bruke først saueflokker og deretter heatmap.

Svar:
Sliter litt med å finne ut hvordan man går fra tur til tur.

Forstod til slutt at man måtte trykke på neste.
FIltrerte lett av og på heatmap og saueflokk.

Spørsmål

1. Hvilke av disse to visningene syns du var best?
   Saueflokk var greiest å se. Syns saueflokkene er greiest å se på både på månedsvisning og ukesvisning. Der har man med antall på flokkene
2. Var det noe forskjell på hvilken visning (saueflokk eller heatmap) som var best på om det var ukes eller månedsvisning?

3. Kan denne funksjonaliteten hjelpe å holde saueflokkene vekke fra områder med rovdyr?
   Hadde tatt grep. Bruker andre bønder for å vite hvor rovdyrene er. Hen melder også ifra at man har sett noe. Setter inn ekstra tilsyn hvis det er stor pågang på rovdyr. Det er mye rovdyr i området, og det er vært mye tap.

Oppgave (Bonitet og PDF)

1. Sett bonitet på kartet, med synlighet 0,7.
2. (Vi viser en PDF som de liksom har lastet ned)

Svar:
Forstod hvor bonitet var, og hvordan slideren fungerte.

Spørsmål

1. Hvilke type rapportering gjør du til staten? Hva skal med i denne rapporten?
   Hen rapportere antall sauer og lam. Registrerer til statsforvalteren. Fargene er ikke så interessante. Er bare interesserte i sauer og lam. Kan være interessant for eget bruk.
2. Hjelper bonitet deg til å forstå hvorfor sauene beveger seg slik de gjør?
   a. Kommer det tydelig frem hva de ulike bonitetsfargene betyr?
   Bonitet kan være nyttig å bruke. Hen har brukt og bruker bonitet til vanlig.

Samling Spørsmål

1. Hva bruker du nå for å holde kontroll på sauene dine? Bruker du en digital løsning, eller foregår alt på papir?
   Bruker telespor sammen med sauekontrollen. Har mange tilsynsturer. Har krav på å gå en gang i uka. Sjekker telespor for å finne hvor dyrene er, så planlegger hen turen ut fra det. Har en tilsynsbok, som hen registrere på papir.
2. Har du registrer sau manuelt tidligere?
   a. Registrer du posisjon?
   b. antall sau, slips og øremerke?

    c. Bruker du denne informasjonen til noe senere? Hva?
    d. Dersom du kunne sett alt dette visuelt på en nettside i etterkant (våres applikasjon), hadde det vært til nytte?
3. Tror du denne løsningen med en slik visning hjelpe deg til å bedre forestå hvordan saueflokkene beveger seg på?

    Det hadde vært enklere å kunne bruke en applikasjon (mobil) og se det i en webapplikasjon etterkant.
4. Var det noe som manglet for å kunne se på bevegelsene til sauene?
    a. Er det noe tilleggsinformasjon av saueflokkene eller en annen type visning som kunne gjort det bedre?

    Så lenge man kan se på telespor og finne hvor de er. Noterer navnet på plassen der hen finner saueflokkene.
5. Kan løsningen gjøre jobben din som sauebonde enklere?
    a. Hen tror det kan hjelpe.
6. Hvor mye beitemark har du?

Tillegg B

Bruksanvisning for å kjøre koden

B.1 Laste ned kildekoden

Kildekoden til prosjektet ligger på et *repository* på GitHub. Linken til kildekoden er https://github.com/stianaad/master.

Det er flere alternativer til å laste ned koden til en lokal maskin. Den første er å laste ned kildekoden som en ZIP-fil. Dette gjøres ved å klikke på **Code**-knappen inne på GitHub-linken til kildekoden, og deretter klikke på **Download ZIP** i nedtrekksmenyen.

Det andre alternativet er å klone koden ved bruk av git. Først naviger til ønsket sted i mappestrukteren på datamaskinen med et kommandolinjeverktøy. Deretter kjør kommandoen:

```
git clone https://github.com/stianaad/master.git
```

Kjør følgende kommando for å navigere seg til kodemappen:

```
cd master
```

B.2 Kjøre koden

Koden er todelt med serveren og brukergrensesnittet, som ligger i følgende mapper *backend* og *frontend*.

B.2.1 Backend

For å kjøre koden til serveren, må Visual Studio være lastet ned på datamaskinen. Prosjektet åpnes ved å klikke på **backend.snl**-filen og kan kjøres i Visual Studio.

B.2.2 Frontend

For å kjøre koden for brukergrensesnittet, navigere til frontend mappen i et kommandolinjeverktøy med kommandoen `cd frontend`. Prosjektet bruker Node.js med NPM, og dermed krever at dette er installert. For å laste ned nødvendig pakker for prosjektet med NPM, kjør kommandoen:

```
npm install
```

For å kjøre prosjektet lokalt, kjør kommandoen:

```
npm run start
```

Anders Gravdal Steen, Stian Ådnanes

Visualisering av registrerte sauer på utmarksbeite

NTNU
Norwegian University of
Science and Technology