Jørgen Aunet Halvorsen

Sheep Shepherding - Grand Overview

Master's thesis in Computer Science
Supervisor: Svein-Olaf Hvasshovd
June 2019

Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

NTNU
Norwegian University of
Science and Technology

Jørgen Aunet Halvorsen

Sheep Shepherding - Grand Overview

Master's thesis in Computer Science
Supervisor: Svein-Olaf Hvasshovd
June 2019

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

NTNU
Norwegian University of
Science and Technology

Abstract

A farmer which holds sheep, needs to hold track of their animals at a regular basis to see if they are in good shape. Today, this can be done using digital bells, and the conduction of supervision trips. There are different information from these supervision trips which the farmer can have use for. This can be position information for the sheep, the route of the trip, where predators have been observed and such. The main focus for the thesis will be to design and create such a prototype, where we are to find different functionality for the such a system with user interaction design.
Today, lots of data are spread over multiple systems, such as predator data, grazing field information and position data for the sheep. The spread of the information can make it difficult to fetch the information one wants, so the sheep can achieve their best life quality. Sheep lost on pasture, especially from predators, will end up being a loss for the Norwegian state, as they have compensation programs for loss of sheep. In other terms, there are multiple stakeholders which want the sheep to come back to the farm after the grazing season is finished. An overview of the areas and the sheep, are essential when one is to monitor the sheep.
The system to be developed will be based on different design principles fetched from already existing map applications. Other data, such as predator observations and grazing areas, are fetched from external providers, and is loaded into the application. This for testing out if this is something a sheep farmer would like to have.
As this is a prototype, first of it's kind, feedback from testing participants are the main basis for the results of the thesis. All feedback, both functional aspects of the system, and what is working and not, will be included in the results. To achieve a better life quality for the sheep, finding which information and features such a system needs to have, is essential to achieve this.

Sammendrag

En bonde som driver med sau, trenger å ha en oversikt over hvor dyrene beiter på sommeren. Dette blir gjort ved bruk av digitale bjeller og gjennomføring av oppsynsturer. Fra disse oppsynsturene er det forskjellig informasjon bonden kan ha nytte av. Det kan være posisjonsinformasjon for sauene, selve turruten, hvor rovdyr er observert og lignende. Oppgaven går i hovedsak ut på å designe og lage et førsteutkast av et slikt system, der funksjonalitet skal finnes gjennom brukertesting av prototypen.
I og med at masse data i dag er spredd over mange forskjellige systemer, som rovdyrdata, beiteområder og posisjonsdata for sauene, kan det være vanskelig og uoversiktelig for en sauebonde å få tak i informasjonen man trenger for å kunne gi sauene den beste livskvaliteten. Sau tapt på beite, spesielt fra rovdyr, ender opp å bli utgifter for den norske stat, da de har erstatningsordninger for tapt sau. Det er med andre ord flere interessenter for at man bør ha fokus på at så mange sau som mulig skal komme tilbake til gården etter endt beitesesong. Oversikt over områdene og sauene er da essensielt for å monitorere sauene.
Systemet utviklet vil basere seg på forskjellige designprinsipper hentet fra forskjellige eksisterende kartapplikasjoner. Annen data, som rovdyrobservasjoner og beiteområder er hentet eksternt og lastet inn i applikasjonen for å teste om dette er noe en sauebonde vil ha.
Ettersom dette er et førsteutkast av en slik applikasjon, er en stor del av oppgaven å få tilbakemeldinger fra testpersoner. Alle tilbakemeldinger fra testene, både funksjonelle aspekter ved systemet, samt hva som fungerer og ikke er tatt med i resultatet av oppgaven. For at sauene skal få et bedre være og livskvalitet, skal det finnes hvilken informasjon og funksjoner som kan tilfredsstille dette.

Contents

List of Figures

List of Tables

1. Introduction

Each year, sheep are sent on pasture where they spend their time for almost half a year, according to the expert. In the wild nature there can be a lot of dangers for the sheep. For example, there can be wolfs or other predators who hunt them. The terrain can be quite challenging where sheep can get hurt or lost. Both the farmers and the government want to take care of their sheep, either if they get stuck, hurt or lost.
The knowledge the farmers need about the environment and the pasture are crucial when sheep farming. Where should the sheep be let out, so they achieve a good quality life? There are many aspects the farmers need to take into account. Are the area hostile, does the area have enough food for the sheep, or are the pasture area too rough when talking about the natural environment.
As there are many different systems and data which can answer many of these questions, those are spread out on many different systems and can be difficult to acquire. In other terms, there doesn't exist one single application which can be used for this purpose. Our goal for this thesis is to collect different aspects, data and information into one application which can help the sheep farmer to take better care of their sheep.
The thesis will consist of three different phases. The first one is the planning phase, whereas there will be drawn different suggestions of visualisations of the system, as well as looking into different use cases which is relevant for the system. After there have been an agreement between us and the expert, the implementation starts. Different functionalities will be implemented and tested, and showed before the expert. The ending phase will include multiple user tests which the results and conclusion will be based on.

1.1. The Expert

Svein-Olaf Hvasshovd, hereby called the expert, has more than ten years with experience walking supervision trips of sheep. As he has much experience with those trips, and some holding of sheep, he has control on most needs of the sheep farmers. Much of the inputs and assumptions done in this thesis are based on his expertise and knowledge.
As the goal of this thesis is to decide what such a system can look like and what features are interesting, one needs to have a starting point. As the expert has much experience in the field and knows some of the difficulties with holding sheep, he will operate as a product owner for this application. Before the user tests will be conducted, the expert will be the relevant person which will see the system during development, and will give feedback on what implemented and what should be focused on to the next meeting.

2. Theory

2.1. Shneiderman's Eight Golden Rules of Interface Design

Ben Shneiderman is a professor at the University of Maryland where his main research have been in various fields of visualisations [1]. This includes User Interface Design and Human-Computer interaction. During his time of research, he have created eight rules which one can use when creating user interfaces [2]. Some of these design principles originates from 1985 but have been through various changes. The ones listed here, originates from his book from 2016 [3].

1. "*Strive for consistency*" – the user experience should be consistent and hence the design should be the same throughout the application. The usage of colours, menus, buttons should have the same layout and functionality throughout the application.

2. "*Seek universal usability*" – the user interface should be usable for everyone. If a user is new to the system, there should be easy to use the application regardless of this. For users with higher competence with the application, there should exists functionality which can make the user experience better for this user group. This can include shortcuts, more advanced menus and functionality.

3. "*Offer informative feedback*" – if no feedback is given back to the user, the user would have difficulties to know what is happening. It is therefore important to strive for good feedback which makes sense to the user. If there pops up a message containing an error code, this information isn't necessarily making sense for the user. This also includes actions when one is interacting with buttons and other elements in the system – something needs to happen when the user is interacting with such elements.

4. "*Design dialogs to yield closure*" – when the user is starting an action, this should start a journey where there is a clear ending of the action. An example, which is used by the web site [2], is when ordering something from the internet. First, one places the order, then one fills out the contact and payment information, pays for the product, and ends at the confirmation page. When one have reached the confirmation site, one knows the process is finished.

5. "*Prevent errors*" – the system should prevent the users to do any errors. If one action shouldn't be active at a given time, that action should for example be

disabled so the user cannot interact with it. If there are errors in a form for example, the state should not be altered after this validation, but rather show where the error is and what it is, and show how it should be fixed to continue the process.

6. "*Permit easy reversal of actions*" – where it is possible and where it makes sense, most actions should be reversible. The user can therefore experiment with clicking on buttons with the belief that it easy to reverse the action made.

7. "*Keep users in control*" – when people are using the system, they should feel that they have control over the application. They should not be surprised when doing an action, and there should be easy to obtain the information they want.

8. "*Reduce short-term memory load*" – as the human brain aren't the best for short-term memory, the application should be designed in a manner, which prevents the necessity too remember. One shouldn't need to remember the actions done previous in previous views of the application.

2.2. Interaction Design

Interaction Design have the focus on the users' goals when creating a new application [4]. This mindset focuses on user oriented design where the design should be developed with the users' goal in mind. This means, one needs to have focus on what the users' goals are, early in the development. This isn't necessary focusing on what the user needs, but rather the goals the user is to achieve. Can the job and routines be done in a more effectively manner for example. Gould and Lewis (1985) describes three principles which make a "useful and easy to use computer system" [5]. The three principles are the following:

- "*Early focus on users and tasks*"
- "*Empirical measurement*"
- "*Iterative design*"

The three principles leads to the basis of user oriented design. As the driving force behind the early focus on the users' tasks and goals, one can capture what the users actually need, which can make their life more effective. One can get feedback early in the process and see where the problems lie. The most important one, when designing decisions taken into the application, is based on the work, environment and their context of usage. One sees how the users work currently, and can find either new ideas of what to improve, to streamline, or new systems which can achieve their goals in a better way. There are multiple ways to get an understanding of the problem to solve. Drawing sketches for the application, or by writing a description of the problem with normal text, can be a way to go.

When discussing the empirical measurement, this will be to concretize what goals the user is to achieve, in manner of the user experience and usability. This will help the developer finding the correct and valid alternatives when designing the system.
The third one is the iterative design. There should be used multiple iterations when creating the design of the application. This means the system designed, should get feedback from users in each iteration. This will make the development more robust, as the system created will be more of a system which have a more usable design. As the developer and the designer never gets it correct the first time (Gould and Lewis, 1985), feedback during iterative design is valuable.

2.2.1. Basic Activities

Four basic activities are discussed for interaction design [4]:

- "*Establishing Requirements*" – this includes finding out what the goals for the application are. Who is to use the system, why is there a need for such a system, and what goals are to be achieved and completed.

- "*Designing Alternatives*" – the main activity of the process. One will try to find ideas and solutions which will meet the requirements created. The step includes two principles, conceptual and concrete design. The conceptual design focuses on what the system functionality is. How it is used, and what it is used for – what goals is to be achieved when using the system. The concrete design includes how the system is to be visualised. What colours to use, design principles, how buttons are designed etc. In other terms, every design element in the application.

- "*Prototyping*" – is the creation of a prototype of the system or application. This isn't necessarily an full functional application, but can for example be a paper prototype, or a dummy application created for demo and testing purposes. The most important is that the user can interact with it and do actions to get an outcome of the application. To see that the goals are fulfilled.

- "*Evaluating*" – the last step of the basic activities, which are evaluating the prototype and requirements created. This should be done with users which are to use the system. One can get feedback directly, both for functionality and design choices made and to see what worked and what did not. The evaluation phase could be done with all kinds of people so one gets a new view of the ideas used in the prototype. The perfect scenario would be to get people which are from the intended user base, but this isn't always possible to manage.

These are the four main activities which is to be used in interaction design. As the user is in focus, and their opinions and thoughts are more important than what the developer thinks. The overall relations between the activities can be seen in Figure 2.1.

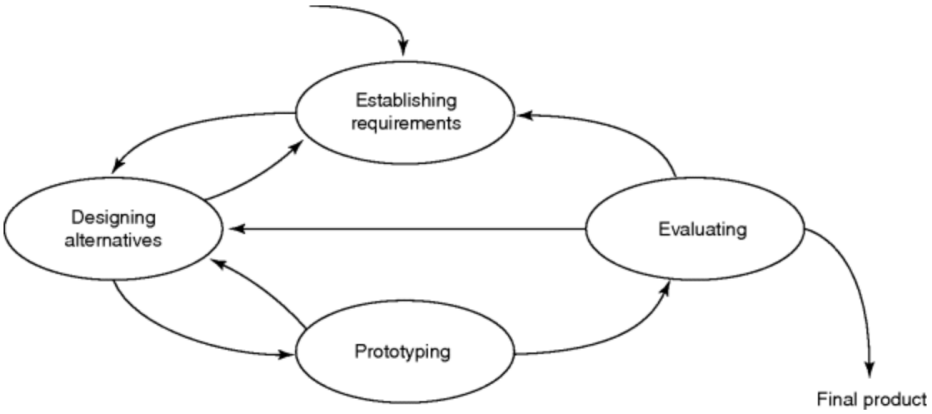

Figure 2.1.: Overall Process of Interaction Design

As one sees from Figure 2.1, the key aspect is the design of the alternatives. One can go back and forwards when using this process. One will go through each step, and each iteration are ended with the evaluating phase. If more is needed, one can go back to the requirements phase, or enter the designing phase.

2.2.2. Practical Issues

There are some practical issues which the book addresses [4]. Which should be taken into account when using interaction design.

"*Who are the users?*" The first aspect to address, is to find out who the users of the system are. As they mention, the question is fairly simple. Finding the optimal users which can be included during the interaction process can be difficult, especially if there are multiple user groups. For example administrators, and the "regular users". Finding the correct user base can make the difference of the evaluation phase much more valuable.

"*What do we mean by "needs"?*" What the user actually needs, is not always as obvious as one thinks. Often the user does not know what is possible to achieve with new technology, and what can be changed. Finding the needs is therefore crucial when one is to find the requirements. An approach is to do some research of what the user does today, what routines they have, what systems they use. Find out how they work, try to address their routines, and find characteristics which can be done in a more effective manner. This will include finding out how they are achieving their goals today, and of course, what goals they are actually achieving. This can include finding new goals which can be interesting for a new system. The goal for the developer is to find a solution which can make the daily life more enjoyable for the end user. The core statement is therefore "the user doesn't always know what they want, before it is shown to them". The main focus should therefore be on what the users' goals are and their usability, instead of focusing on the their needs and demands for a system.

"*How do you generate alternative designs?*" There are multiple ways of generating alternative designs. One can look into related systems which solves some of the goals

today, or one can look into completely different systems which is used in other contexts. The main focus should therefore be to get inspirations from different design choices. One other approach is to discuss with other designers. The process is called cross-fertilization, where one discusses design alternatives with someone who haven't the same pre-knowledge of the goals which are to be achieved.

2.3. Paper Prototyping

In the early stages when one is to create a new system or a new user interface, one technique to use, can be used to get ideas and inspiration is to use paper prototyping. This is a technique which one draws multiple versions of it's proposed user interface [6].

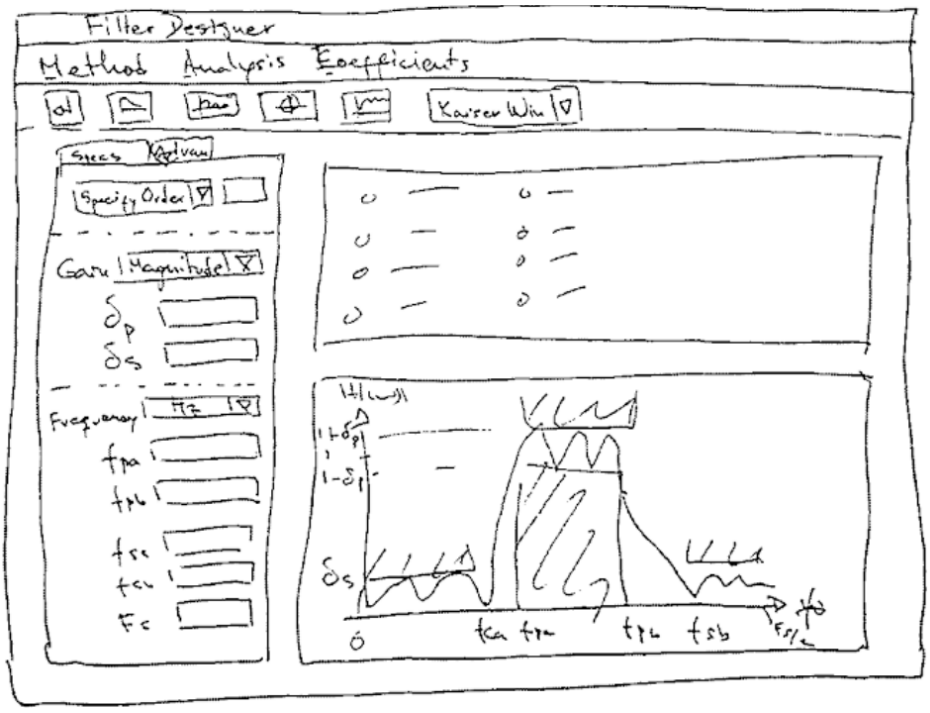

Figure 2.2.: An example of a paper prototype

In Figure 2.2, which is fetched from [6], there is an example of a paper prototype. In this example, the whole screenshot of an application is drawn. This includes headers, buttons, lists and graphs. Interactive elements are also included. After multiple prototypes are created, one can test the sketches with a user which is to be using the system. As paper prototypes are easy to create, one doesn't need technical competence to create and use them. One can create multiple different visualisations and examples, and see which ones work as intended. One can experiment with different ideas, so one doesn't get stuck in one track too early in the process. One other important notice, is that it "encourages creativity in the product development process" [6]. Ideas can pop up when one is creative, which is a state one can achieve when drawing sketches of the system.

2.4. Usability Testing

"Usability.gov is the leading resource for user experience (UX) best practices and guidelines, serving practitioners and students in the government and private sectors." [7]. This is a website which holds different guidelines and principles according to design processes and UX. The website and it's content, is administered by the the *Digital Communications Division*, which is a branch of the *U.S. Department of Health and Human Services*. Why one should use usability testing and how to conduct them, are information they hold.
Usability testing is a way of finding out how well an application is working regarding it's usefulness [8]. Are elements well implemented? Does the system have an intuitive design and functions, for example? The focus of the tests, is the usage of users. In other terms, the test participants meanings and actions are in focus. The main goals for a usability test are to find how satisfying the usage of the application is. To see if there are problems regarding the user experience, and collect qualitative and quantitative data regarding the application. There are three steps one should use when running a usability test:

1. *Develop a test plan.*
2. *Recruit the participants.*
3. *Evaluate the results and report them.*

One of the main benefits of using a usability test as the testing method, is that one can address issues, usability and functionality directly with the users. As the developers can stick to the same path for a long time, fresh eyes from test participants can give a new look on the application. It is recommended to test the application with real users, which can be users from the user base or "ordinary" people. They list up five benefits for using this kind of test [8]:

1. *One can find out wether users can complete different tasks using the system.*
2. *Find out how much time is used completing the tasks.*
3. *Identify the degree of satisfaction for the usage of the system.*
4. *Can find improvements and features which can raise the satisfaction for the user.*
5. *See patterns wether the usability objectives are met or not.*

2.4.1. Planning a Usability Test

The first step is to plan how the test is to be conducted [9]. What are the goals and the purpose for the test. This will include planning for how many participants to test the system on, what metrics to test, generation of tasks, and what the test should include. The following elements should be included in the test plan:

- *Scope* – what will be tested, what part of the software.
- *Purpose* – what is to be achieved by conducting the tests.
- *Schedule and Location* – where the tests are to be conducted, which environment.
- *Sessions* – describe how long each session will last, and how the structure of each session is set up.
- *Equipment* – what equipment to be used, by the tester and the observers.
- *Participants* – which user group is to be tested, how they are recruited.
- *Scenarios* – describe the tasks to be used and how many should be conducted.
- *Metrics* – describe how to measure task completeness, and other aspects to be measured.

2.4.2. Running a Usability Test

After the test plan have been created, one can start running the tests [10]. A test can be conducted in different ways. Two of these follows:

- *Concurrent Think Aloud* – the main key is to make the participant to think out loud when doing the test. The observers will not interrupt during this phase, and will not ask questions, only observe what the participant does and says.
- *Retrospective Probing* – after the testing session is finished, different questions are asked to the participant. Regarding different aspects of the system, the observers can ask different questions about the system, and actions made.

It is important to make the participant comfortable, for example by explaining how the system is to work, it's intentions and purpose. As well as mentioning that it is the system which is to be tested, not the participant. Another aspect which is important to remember, is that one shouldn't help and lead the participant if they get stuck with a task. To see if they can solve it by themselves first, and if they get too stuck, give them hints if necessary. It is important to take good and detailed notes of their actions and thoughts in this phase.

2.4.3. Evaluation and Reporting

After the tests are conducted, the most important part is to summarize the data and show the results of the tests [11]. It is recommended to put the data metrics into tables, and show interesting areas of the application where it fits. In regards of what worked and not – focus on being precise when showing the results. One thing which one needs to identify, are recurring problems which occurs. This can include recurring proposals for new functionality and fixes for the application. There are two data types which are collected and recorded during the tests.

- *Quantitative Data* – how well the participants did it on the tasks, such as time used and success rates. Data, which is collected using the System Usability Scale (SUS), is also put into here. How well the system is based on usability.
- *Qualitative Data* – data which is more about the tangible data. This includes data which describes problems and errors the participant encountered. It can be different suggestions and comments to new functionality and fixes for the application. Others can be observations of how the participant navigated through the application.

Now the time is in for writing the usability test report. There are different things which needs to be included in the report.

- *Background Summary*
- *Methodology*
- *Test Results*
- *Findings and Recommendations*

When reporting the findings, everything described and noted should have it's origin from the data observed. In other terms, things that the participants said and did. After this is done, the recommendations should be implemented, and tested again. Also, it is important to add what worked well, and as intended. Regarding the functionality and choices, what to included in the next iteration.

2.4.4. System Usability Scale

The System Usability Scale (SUS) is a scaling system which is used for measuring the usability of a system [12]. The scaling system has become the industry standard, and it is used widely around the globe. The main benefits of the system are for the first, that it is easy to get feedback from the participants regarding usability. The second is that it can be used on small groups of participants, as a small number of participants can give results which makes sense and are durable. The third one is that it can easy tell if a system is usable or not, it is valid. In other terms the SUS gives a good indication whether the application or system is usable or not. It is important to note that SUS is a tool to find out how usable a system is, it's easy of use, and that is the only purpose of it. The following ten questions are asked in the SUS (fetched directly from [12] without paraphrasing).

1. "I think that I would like to use this system frequently."
2. "I found the system unnecessarily complex."
3. "I thought the system was easy to use."

4. “I think that I would need the support of a technical person to be able to use this system.”
5. “I found the various functions in this system were well integrated.”
6. “I thought there was too much inconsistency in this system.”
7. “I would imagine that most people would learn to use this system very quickly.”
8. “I found the system very cumbersome to use.”
9. “I felt very confident using the system.”
10. “I needed to learn a lot of things before I could get going with this system.”

For each of these questions, there is given a score from zero to four, where zero strongly disagree, and four strongly agree. After all scores have been given for each test, they are summarized and multiplied by 2.5. This will get the total score between zero and one hundred. For all tests conducted, the average total score is computed and if the average score is above 68, the score system is considered usable.

2.5. Single Page Application

Single Page Application (SPA) are web applications which load most of the HTML and JavaScript in one request [13]. One of the clues described is that the application will not reload when the user is doing some kind of action in the web site. With this technology, one can achieve less usage of bandwidth as less data is to be sent during the usage of the application. The first request when loading the application will be larger than a typical single page load, but one only needs to load the application once. Another benefit when using SPAs, is that the navigation is done much faster. As JavaScript can take hold of the action instead of sending a regular Hypertext Transfer Protocol (HTTP)-request to the server, the server processes the request and sends back a new HTML file – all is done in the browser. This can possibly make the application more responsive and hereby a more smooth and better user experience can occur. If one needs a larger set of data, this can be fetched asynchronously from a Application Programming Interface (API) for example. In Figure 2.3 there is a typical SPA-infrastructure which is fetched from the same article [13].

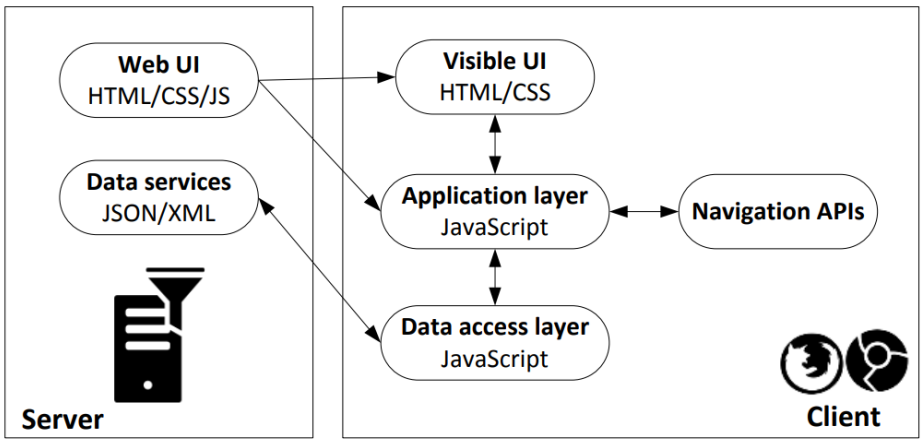

Figure 2.3.: Single Page Application architecture

Here one sees two things which are fetched from the server – the web interface and various data sources using APIs. The application layer and navigation are done in the visual user interface using the browser, which makes the application faster from the user's perspective.

2.6. Coordinate Systems

When one is to visualise the earth using maps, there are multiple ways and coordinate systems to use. Since the earth is displayed as an orbit, one cannot stretch the earth out on a square field. The projection used in this thesis, is the Decimal Degrees format.

2.6.1. Decimal Degrees

The decimal degree system is based on the traditional geographical system [14]. The system is based on how many degrees the point is from the equator, north or south, and similar how many degrees, east or west, the point is from the Royal Observatory in Greenwich, which can be seen in Figure 2.4, fetched from [15].

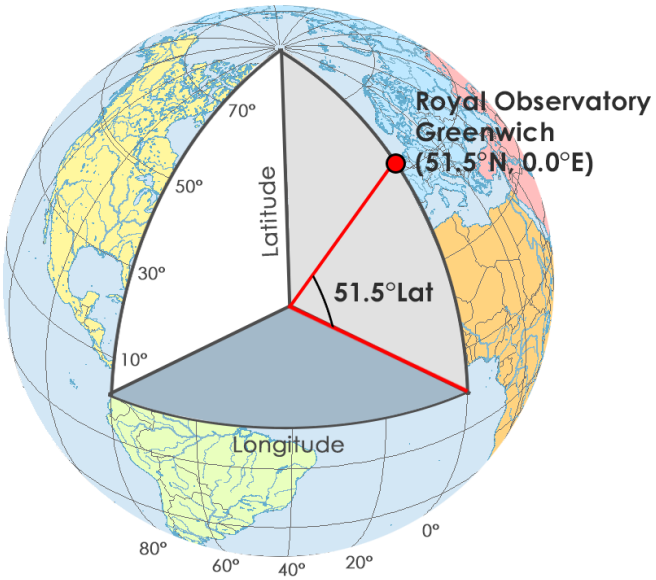

Figure 2.4.: The Decimal Degree Coordinate System

The traditional system would use the integer degree, and add the number of minutes and seconds to the degree. This would then form the point of the coordinate, for example the latitude 61°38' 11" and longitude 8°18' 47", which would be described as 61 degrees, 38 minutes and 11 seconds north and 8 degrees, 18 minutes and 47 seconds east. This is the position for Norway's largest mountain. The more normal and modern way of using this system, is using the Decimal Degrees System. The system uses the same degree system, but instead of using minutes and seconds, it uses the fraction of the degrees [14]. The same coordinate for Galdhøpiggen would in the decimal system be 61.63635°north and 8.31323°east.

2.6.2. GeoJSON

GeoJSON is a simple data format for organising geographical data. It follows the setup of JavaScript Object Notation (JSON) [16] where the format is shown in the following example:

```
{
  "type": "FeatureCollection",
  "features": [
    {
      "type": "Feature",
      "geometry": {
        "type": "Point",
        "coordinates": [11, 60]
      },
      "properties": {
        "name": "null island"
      }
    }
  ]
}
```

This format contains a type which is called “FeatureCollection”. This contains a list of multiple features. Each feature can be a type of the following:

- Point
- LineString
- Polygon

Each feature can also contain a field called “properties” which is an object which can contain statistical data. These data is what is interesting beyond the actual geometry that is drawn on the map. When drawn on a map, the example given will look like this:

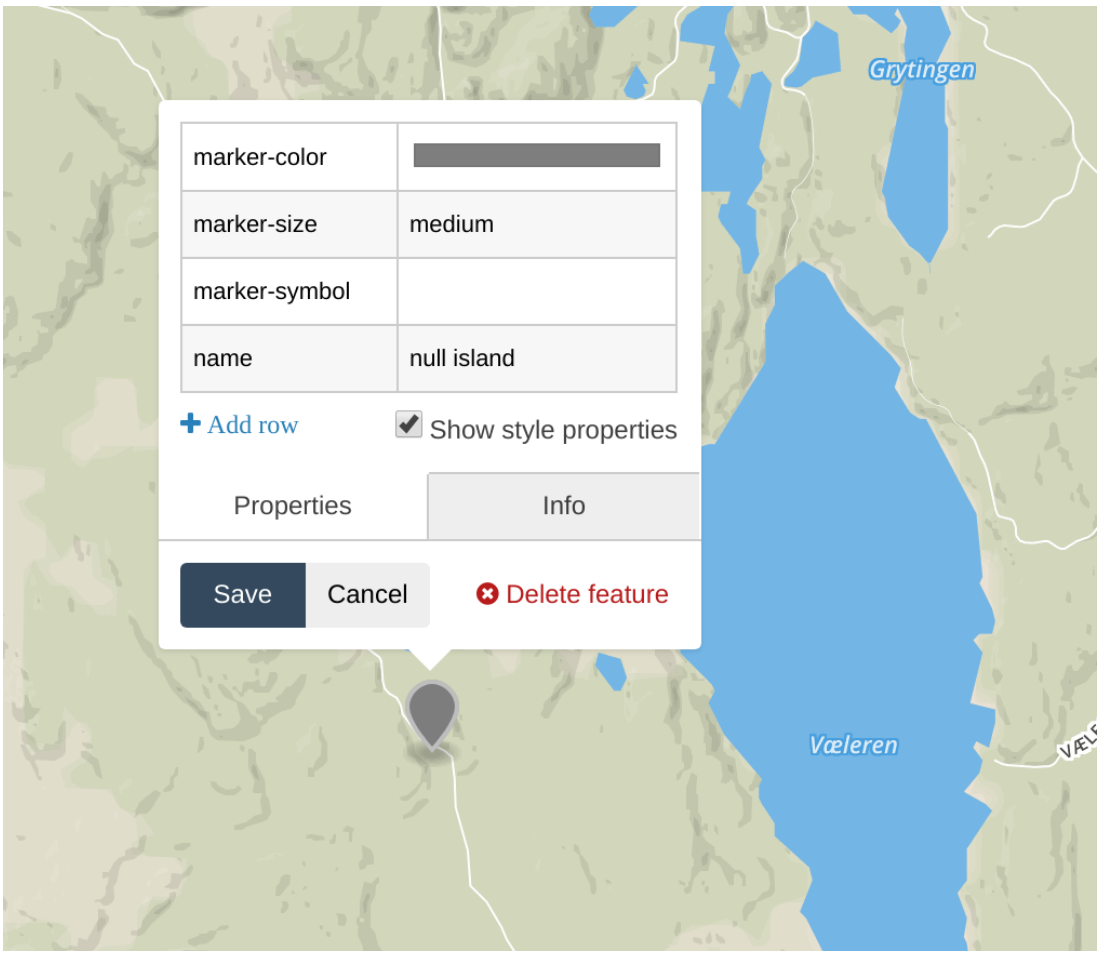

Figure 2.5.: Example of GeoJSON

The image is created using a website called geojson.io [17], which interpreter the GeoJSON and paints it on a map. Here we can see it's property name, which is "null island", which is our custom property. If this for example was a number, and multiple points were made, the software could have created a heatmap for example.

3. The Problem and Process

3.1. Specialisation Project

The system to be developed, will be an integration of the plotting system that were developed in the specialisation project. This system was created as a mobile application, as a single page web application. The goal of the mobile application was to create a tool the farmer could use when doing supervised trips of the pasture. Those supervised trips are today done manually on paper, and one will encounter different problems when using paper in the wild. Such as bad weather, loss of paper, and inaccuracy when plotting observations on the map. The route which the farmer wanders on such trips, are something that could be interesting for analytical and statistical reasons.
As the Norwegian Mattilsynet demands that sheep farmers should report the number of sheep they own, once a year [18], the farmers need to hold track of their animals. If there are deceased, injured or lost sheep, this should be reported. For the farmer to get compensation for deceased sheep taken by predators, they need documentation that the sheep are taken down by predators [19].
The mobile application supports tracking of the farmers position during their supervised trips, so the route can be tracked. The farmer can track observations of sheep, such as how many they are, how many lambs, and the colour of them, which is divided into different trips conducted. If the farmer sees a predator in the area, this can also be plotted in the application. More information about the data format and it's structure, can be read in Chapter 5.

3.2. Process

The process which will be followed during the development of the application, is the Interaction Design approach (described in Section 2.2). As we don't know exactly what the sheep farmer want, the approach will be to find the main goals and needs for the sheep farmer. This will follow the four-step activities in the Interaction Design. The first step is to address the main goals and requirements for the farmer. What are their main problems today, and how we can find a solution which can solve the problems in a more effective manner. We will look into different systems and solutions which are used today, in which we can find inspiration to design the correct system to display for the user, and various functionality. This is the done during the establishment of the requirements.
After some of the requirements are set, there will be a routine where we hop between the prototype activity and the design of different alternatives. This will include paper

prototyping and sketching of different solutions, looking into existing solutions and solutions with relevant visualisations and design for our application. The iterative part will mainly be conducted between us and the expert, where we find out different aspects which can be interesting for the farmer to be using in their daily life. For this thesis there will be conducted one iteration, ending with the evaluation. The evaluation will be done conducting user tests, which the results will be based on. The results will include what functionality that worked, what did not, and proposed changes for next iteration. Various choices of design will be included as well. The main problem is described in the Introduction. The next section will describe the main goals which were found during the requirements findings.
As it can be difficult to know what the user wants, as mentioned in Section 2.2, there have to be used time to enter the farmers mind set. Some of the research done in the specialisation project helps for this purpose, as the same theme was discussed. This will be further mentioned in the State of the Art. We are trying to find a system where the sheep farmer can keep track of their sheep, and finding interesting locations for them. How they can achieve their goals in a more easy manner.
The importance of knowing who is the user base of the system, can not be neglected. For this case, the user base are sheep farmers. This can be farmers which loose a lot of sheep on the grazing field, or farmers which just want to see where their sheep are. The expert will evaluate the system and alternative designs during the alternative design and developing phase. As the user group can have a high age, it can be assumed that the system should be easy to use as technical competence isn't necessary high, according to the expert.
The stage of understanding the user, and getting into their shoes were mainly done during the specialisation project, and by using the expert's knowledge in this field. No field analysis are to be conducted. This is to the lack of time and availability of the sheep farmers as the application is to be used during the season, and the sheep aren't out on pasture before late spring. To get real observations would therefore have been difficult. The knowledge of the sheep farmers are therefore based on the expert's routine and knowledge, similar for the mock data to be used.

3.3. Main Goal

The main goal for this thesis is to find a way to visualise the data described in the specialisation project, which we look into in Chapter 5. How can sheep farmers use this data in their daily life to keep better track of their sheep's care and living. The farmer should get some relevant and interesting information out of this system to it's use. One big issue for farmers are that sheep get lost, get killed or get hurt. How can the usage of this data help the farmer when setting out the sheep on pasture, and holding track of them.
Our goal is to use different visualisation techniques to display the data on a map in a straightforwardly manner. Which we will see in the State of the Art, there exists different solutions which tries to achieve some of the same goals.

Another aspect which will be looked into, are how predator data is to be displayed. As there are multiple research institutions in Norway which keep track of different types of data that could be interesting for our case, some of these data will be combined to create an application which sheep farmers can find usable.
In other terms, our main goal isn't only to find a system which can be used by the sheep farmers to track sheep, but also track what kind of information is relevant and reasonable. Is it possibly to combine statistical and historical data, based on observations done by researchers and the farmers them selves.
It is important to clarify that our system isn't meant to substitute other systems out there, but to be a good addition to the existing ones. For example our goal isn't to track sheep in real-time, but rather focus on the behaviour of the sheep, where there are areas the sheep wander in.

3.4. Loss of Sheep

According to the Norwegian Environment Agency, there were a loss of 16,905 [20] lambs and sheep due to predators in 2018. From a total of around two millions lambs and sheep, this makes for a loss percentage of approximately 0.8%. This number isn't very high, and the curve of loss per year have been declining [20]. This is tracked by the sheep farmers, which are applying for compensation for deceased animals by predators. To get this compensation, one needs different proofs such that the government know for a fact the animals are killed by predators. Valid proof includes cadaver findings and autopsy of the dead animals where one can detect wether predators have been involved. There are five predators which hunt sheep in Norway, which are wolves, bears, wolverines, lynx and golden eagles. From the report [20], there appears to be lynx that kill most sheep in Norway. The government payed out compensation for 43.3 million NOK in 2018. The cost for the government is still quite high, and the farmers use a lot of time finding their sheep and cadavers. Therefore an overview tool can have it's place to make the number of deceased sheep smaller.

3.5. Requirements

One of the main purposes of the application, is to visualise the data for the farmer in an easy and simple manner. As there should not be difficult to extract data with a lots of filters and possibilities, the use of predefined cases were created. This section contains the requirements which follows the second step in the interaction design principle.
"I am a farmer that is holding sheep.":

1. "I want to be able to see my last four trips in the same view. Information about the trips, some analysis on them, and see some kind of movement pattern from a group of sheep. How they are spread in the environment and what sheep have moved where."

2. “I have been in a area where wolverines are observed. The area is traversed regularly, and three of the last trips are in this area.” – here the area is most interesting. See all trips that have been in the given area.

3. “I want to see where there is possible to observe a predator based on the observations conducted”

4. “I want to find a new area for my sheep to stay. Based on observed predators in the area and terrain, plot areas where the animals should not be released” - should be based on trips conducted by the farmer.

4. State of the Art

This chapter will look into different systems which can relate to our discussion in Chapter 3. The State of the Art will mainly look into systems which can help to get inspiration to find relevant design alternatives, according to the Interaction Design in Section 2.2. There will be looked into systems and functionality that exists on the market for tracking sheep, mainly with digital bells. Relevant data and research institutions for holding sheep, grazing areas and predator observations will be studied as well. Other systems which is using map technologies and visualisations are also relevant for this task. The use of Shneiderman's golden rules, will not be taken into account in this chapter as this is more relevant for the development of the prototype.

4.1. Bell Tracking Systems

There are mainly two systems that exists for tracking animals: Telespor AS, Findmy and Shiip. This part will look into these systems, mainly how they have implemented visualisations of the data. As these systems have the option to be used on different animals, some of the functionality can be a bit wide. A more specialised system which the main focus is on sheep, is therefore preferable. All data and information in this section are collected from the home site of Telespor [21], Findmy [22] and Shiip [23]. As all these systems are behind a login wall, all information and screenshots from their applications, are fetched from their websites, guides and examples.

4.1.1. Telespor AS

Telespor is a Norwegian company that offers tracking of animals. They use a digital bell for tracking the animals, which uses the Groupe Spécial Mobile (GSM)-technology. If the animal wanders in an area with bad GSM-connection, their digital bell will have no possibility to update it's position. When the animal reach an area with GSM-connection, the position can be tracked again. The bell itself, uses Global Positioning System (GPS) to track the animal's position.
Telespor offers two kinds of applications to use when tracking the animals - a mobile application, which is supported on iOS and Android, and a web application. As the applications lie behind a login wall, the applications are not tested directly. Images and information are therefore based on their user manual and website. If there is not described how the data is processed, some assumptions are made.
All information and images are obtained from Telespor's user manual [24] and their website [21].

Data Gathered

What data is gathered via the bell.

- As the bell has motion sensors, it can track how much the animal is moving
- The bell has two-way communication, which means one doesn't need physical contact with the bell to program it.
- The bell's position - how often the position is tracked, depends on the settings set by the farmer. Since the bell supports two-way communication, the location update frequency can be changed whenever.

Mobile Application

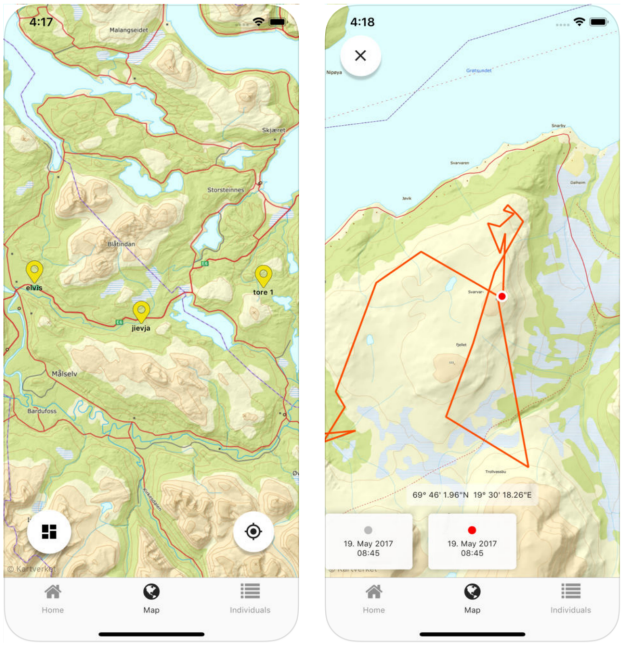

Figure 4.1.: Telespor Mobile Application from iTunes [25]

In Figure 4.1 there are two screenshots from their iOS application. The active tab is the "Map" tab. In the left screenshot there is assumed that each point, in yellow, are the current locations of different sheep, or at least the last tracked location of them. In the right image one can see the movement pattern for a given sheep. The historical points can be clicked at the bottom to see the animal's earlier positions. It seems that the track visualised in the application are just lines between the tracked points. This makes the track a bit angular in it's shape. The map used has contour lines which makes it easier to see where the observations is, according to the expert. In this application this is not the most important as the observations are done via the digital bells with the usage of GPS. When the sheep is to be harvested, this is more important.

As this is done on a mobile screen, large amount of data cannot be placed on the screen. The screen is small and too much information on the screen can make the information disappear in it's context.

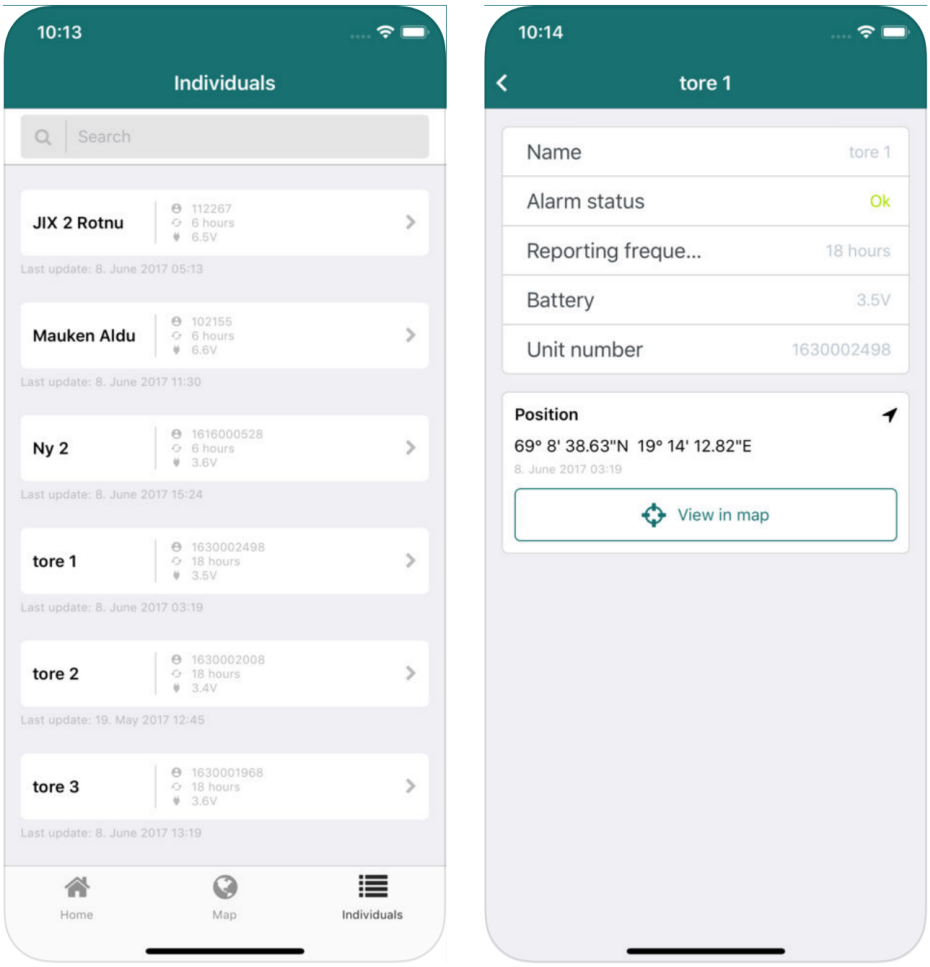

Figure 4.2.: Telespor Mobile Application from iTunes [25]

The left image in Figure 4.2 shows the "individuals" tab in the application. Each animal registered are listed up, where one sees a small arrow at the right of each animal. This is indicating that more information are to be showed if clicked. Each row shows when the position was updated, and update frequency. The others aren't known. The right image in Figure 4.2 shows the view when one is clicking at an animal. A more detailed position is shown in this view.

Web Portal

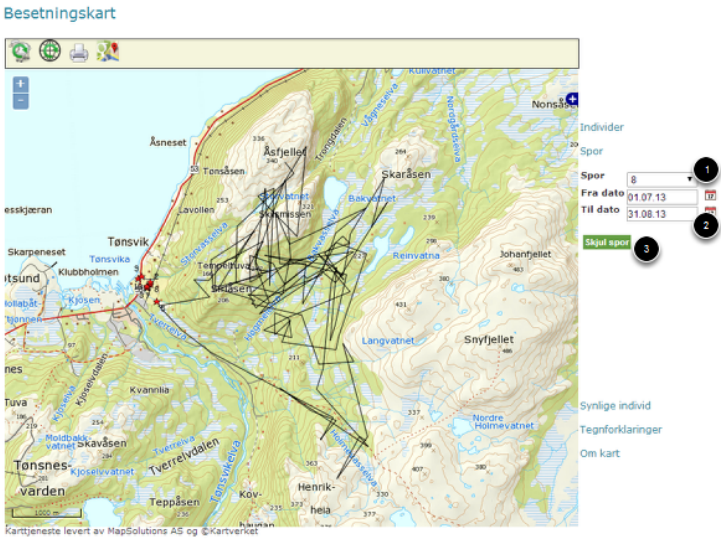

Figure 4.3.: Telespor web application - animal route

From the view which is seen in Figure 4.3, one can choose dates for a time period one wants to see the tracking of a given animal. For example if one wants to see where the animal moved in a given month. The same problem arises here as in the mobile application, the lines get quite angular and the route is a bit messy and complex. On the other hand, as described by the expert in Chapter 1.1, there is interesting to see the behaviour of the sheep over time. This means too see where the sheep wanders and at what speed. For example, if the sheep wanders over large distances in a short period of time, could that be an indication that there are predators in that area? These are some questions that could be useful for the farmers to get an answer on.

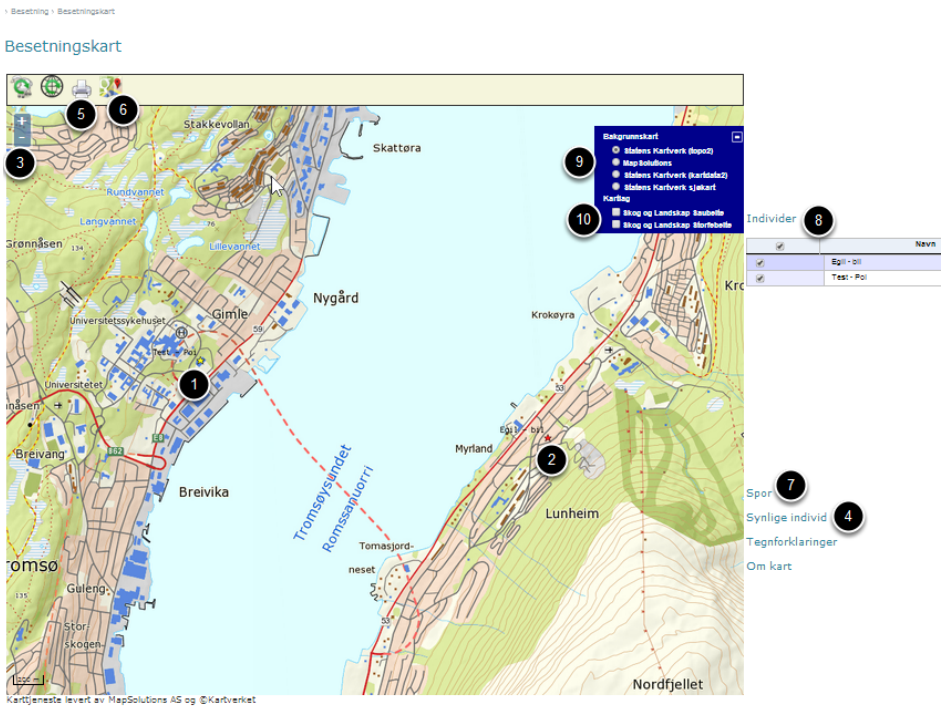

Figure 4.4.: Telespor web application - main view

In the overview page, which is seen in Figure 4.4 one can see the following:

1. A yellow or red star is a representation of one individual on the map.
2. Same as number one.
3. Classic map zooming. When entering the application, all animals will fit into the same view.
4. All animals in a list.
5. Print of the map.
6. Use of Google Maps.
7. A menu for filtering what animals to show in a given time period.
8. What animals to show in the map.
9. Select which map system one want's to use.
10. Choose a grazing map as an extra layer to see if the animals graze on the best areas. (not all municipalities have a grazing map)

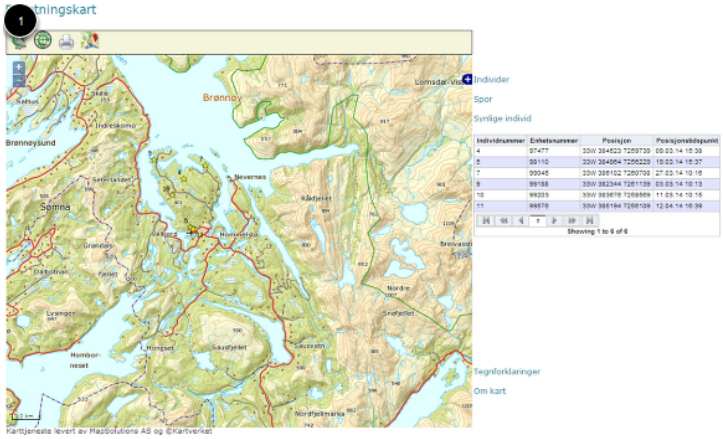

Figure 4.5.: Telespor web application - individuals in an area

In Figure 4.5 there is a view with all individuals that are tracked in a given area. When pressing (1) in Figure 4.5 the application will fetch all animals that are in the area of the map and show them as a star on the map, and put them into the list at the right of the map.

Lacks

One of the main problems with Telespor's solution when discussing the visualisation, is the use of colours and figures in the map. For example the yellow stars that is indicating where an individual is, is quite difficult to locate and also see on the map. In Figure 4.3 the black colour used to show the path of the animal is a bit over-complexed as it is difficult to see where the actual route is in detail. This is better shown in the iOS application in Figure 4.1 where the track is red and is more distinguishable in relation to the background.

4.1.2. Findmy

Findmy is a Norwegian company which offers tracking of animals with the usage of digital bells. The bells use satellite technology to track their position and for communicating. Findmy can be used on multiple platforms. They offer mobile applications for the operating systems iOS and Android, whereas the application is supported on tablets of both operating systems as well. The support on computers are also good, as they have a JAVA application and a Windows application. A lack is that no web application is supported, as one needs to install the application before usage.

Data Gathered

The digital bell tracks the following:

- The animal's position

- Firing of alarms
- The usage of GeoFencing

Mobile Application

The mobile application has a simple representation of the data, as can be seen in Figure 4.6.

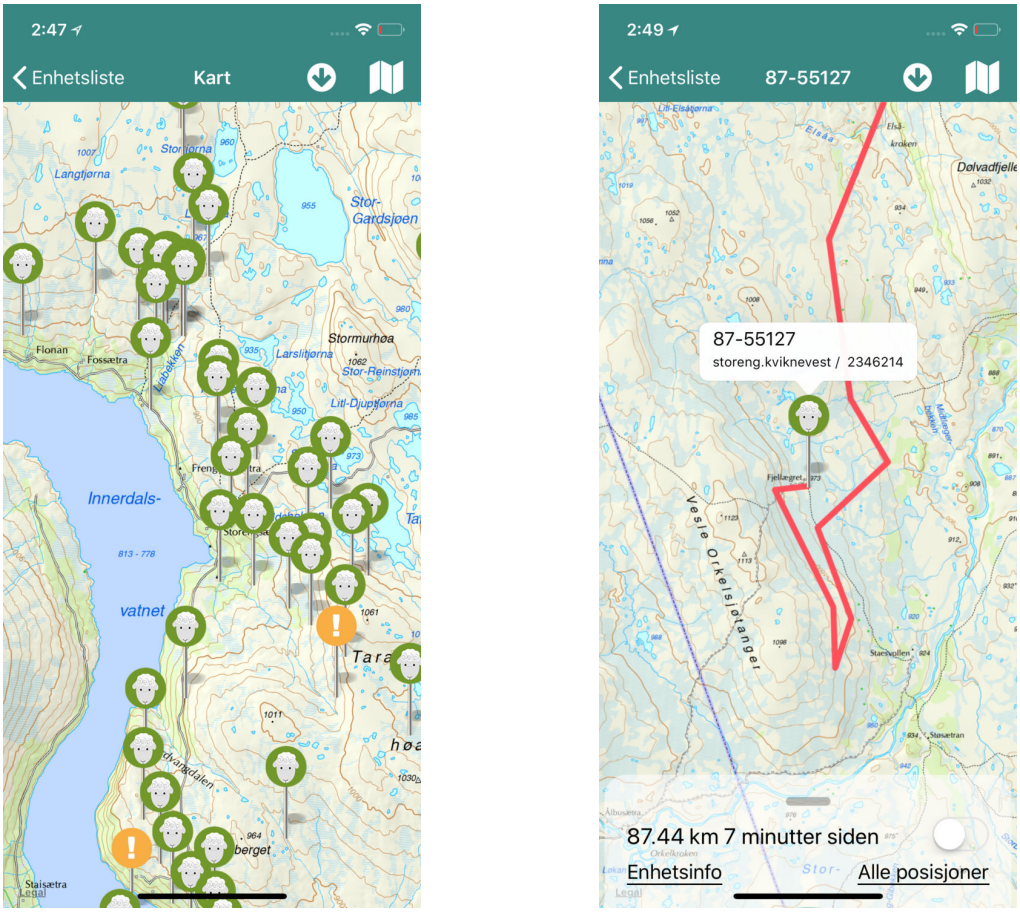

Figure 4.6.: Findmy mobile application – area view

In the left image of Figure 4.6 one sees Findmy's implementation of how to see where the sheep are. The same image in a tablet view, where the screen is quite larger, can be seen in Figure 4.7. Each points corresponds to an animal, and in this case, a sheep. Here they decided to use icons for the representation of the sheep. From the icon, there is a line which connects the icon to it's actual position. On a small device, like a mobile phone, with a lot of sheep in the area, it gets a bit messy and complex showing the data. There is difficult to see where the sheep's actual position is. If the position would have been directly connected to the icon, it would have been more clear where the locations of the sheep are, as described by the expert. It is a bit confusing and complex.

When clicking one of the sheep icons one can see the track of the chosen animal, which can be seen in the right image of Figure 4.6. The track itself is in the colour red and is quite distinguishable from the map background. The same issue here as with Telespor, is that the track is a bit angular, as there are hard lines between the "observed" points. The use of icons as what animal is located at a given point makes it more intuitive what the point actually means. One can see that there are some points with an exclamation mark in the left image of Figure 4.6 – this points are indicating that something is wrong with the given animal. If the animal haven't moved in the last two days, for example. As exclamation marks can indicate that something is wrong, it is quite understandable what it means.

Figure 4.7.: Findmy tablet application

There is easier to see that a larger screen can fit more information in Figure 4.7. In this case it isn't necessary a positive thing. The signs are getting too small, where another solution to show many could have been done in a more elegant way. The icons almost disappear on the map and is not always as easy to see as they are in the mobile view, where they are larger and more distinguishable.

Computer Application

The computer program is a bit more detailed than the mobile application. In Figure 4.8 one can see something more than on the tablet and mobile applications. For sheep that has lambs in it's flock, there are shown a tiny yellow circle with a number that indicates how many lambs there are in the flock or for that given ewe. It isn't the most intuitive way, but it is understandable.

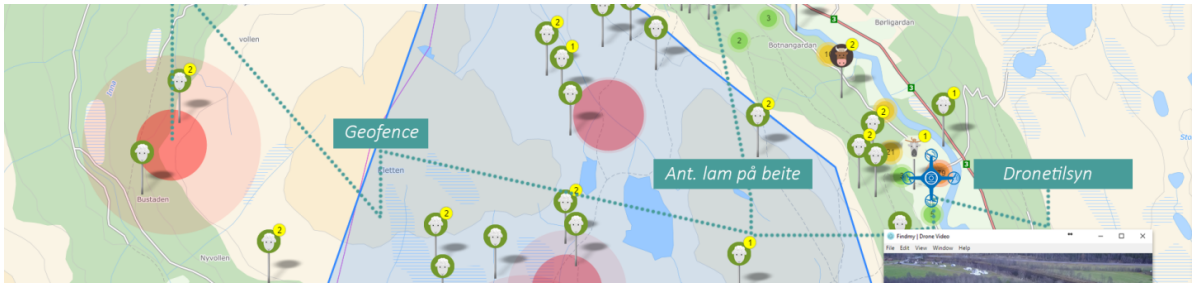

Figure 4.8.: Findmy application

As seen in Figure 4.8, Findmy supports the usage of Geofencing, which is a way of creating a virtual fence in the application. This is a virtual fence which can fire specific alarms. For example if the animal walks outside the virtual fence, the farmer can get a warning from the system that the animal has walked outside. This is not relevant for our application, as the statistical and analytical approach are more interesting. Our system will not support "live feed" of the positions of the sheep, as that is not the intention of the application.

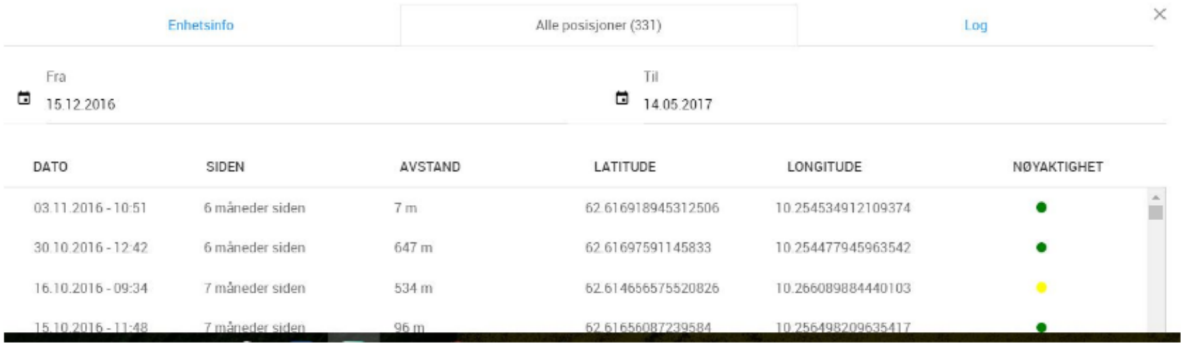

Figure 4.9.: Findmy listing of positions

In Figure 4.9 one can see a historical list of all positions tracked in the application between for one electronic bell. Under the "Nøyaktighet" column, there are two indications on the accuracy of the position. If the colour is green, the bell got contact with enough satellites to get an accurate position, if it is yellow however, the position cannot be hundred percent accurate as the bell couldn't get contact with enough satellites. The position is registered regardless of this, as the movement of the animal is of interest as well.

4.1.3. Shiip

Shiip is a quite new Norwegian company which creates and uses digital bells for tracking of sheep [23]. The bell which they use are a technology which is called Smartbjella [26]. The bell uses the GPS technology to track the bell's position in the field. The bell have support for two-way communication using GSM-connections. The bell can send around 14-15.000 messages with it's location before the power is used up. We didn't manage to find any screenshots of the application they use for managing the bells, but they list up the following which can be done using their web portal:

1. Showing a map with the position of the units in real time.
2. One can register and manage metadata used for the units.
3. The ability to track the position, even indoors.

As the bells aren't set out to the market yet, there will be no more discussion for this solution, but should probably be looked into for the next iteration too get more inspiration for the system.

4.2. Research Institutions

There exists multiple research institutions in Norway which holds track of different aspects of the norwegian society. Two of these are NIBIO and NINA, which both holds track of data that are interesting for our case. As some of the data and statistical approaches that are available for free usage, these data have relevance for us.

4.2.1. Norwegian Institute of Bioeconomy Research

The Norwegian Institute of Bioeconomy Research (NIBIO), is an institute which has it main focus to deliver knowledge and research about different aspects of bioeconomy. As described by NIBIO [27], "The basis of bioeconomy is the utilisation and management of marine and terrestrial biological resources". One of their main purposes is to contribute to a secure and safe production line within the food industry. It also include doing research about the norwegian forests, resource management and different biobased industries.
Some of the areas include statistical information and maps. The most interesting area for our case is their grazing maps. In terms of the grazing data, NIBIO gets this information from the Norwegian Agriculture Agency, which is an agency of the Norwegian Ministry of Agriculture and Food. They collect data, among others, which NIBIO uses for they grazing maps.
The data that lies open on their website can be used for free as these data lies under the Norwegian Licence for public data (NLOD 1.0) [28]. This will therefore be referred to in the application.

The Data

NIBIO offers a service called “the Source”, or “Kilden” in Norwegian. This is a map service which most of their data can be visualised in. They use different visualisation techniques such as polygons for visualising areas of interest and different colours for distinguishing between layers of information. In Figure 4.10 there is an example of this.

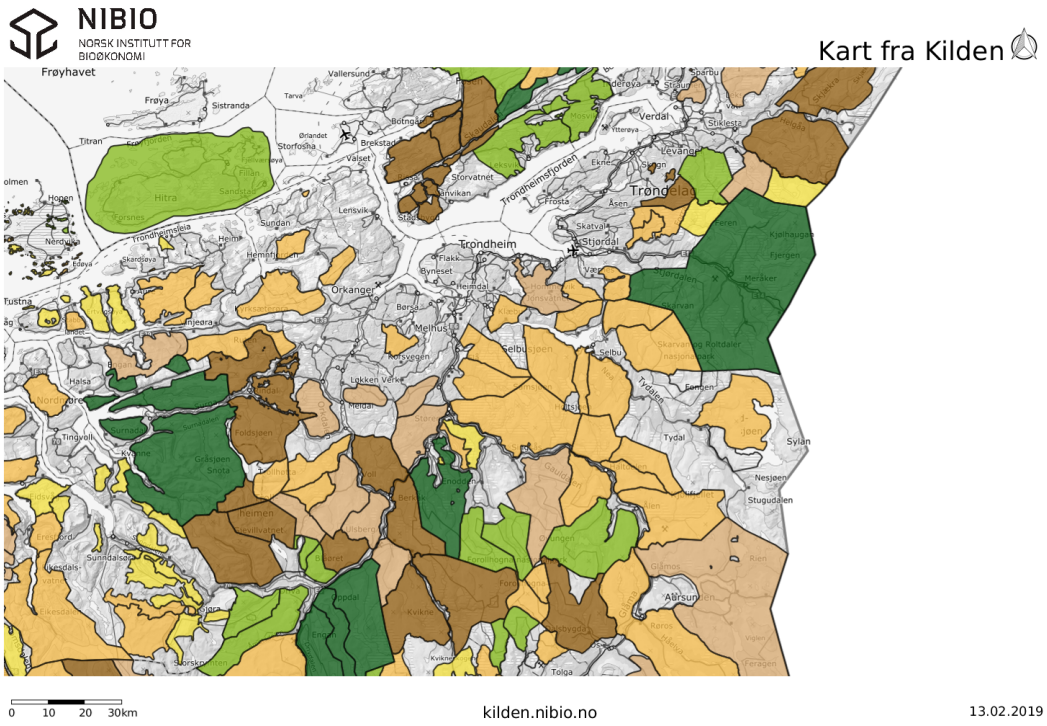

Figure 4.10.: Example of the data that can be fetched from NIBIO

Each polygon shown in Figure 4.10, are representing different grazing areas. For this case, each colour is giving an indication on how many lambs there exists in the area. As the Source have multiple information it can show, this is merely a simple example of the usage. Other examples is to see the loss percent of lambs in each grazing area, which is seen in Figure 4.11.

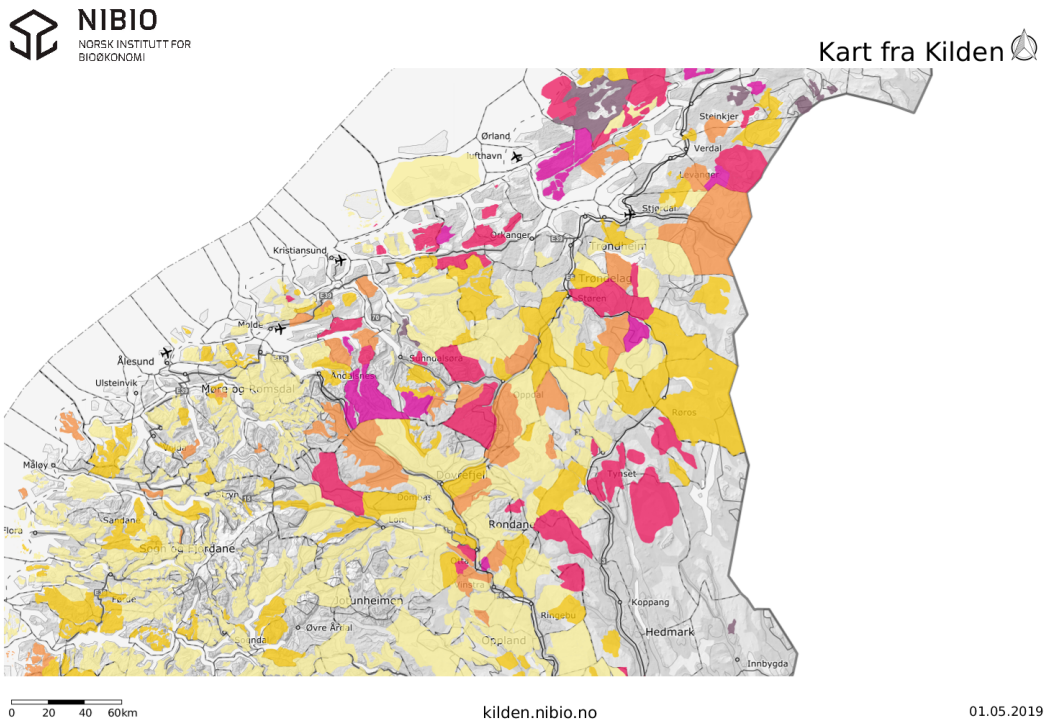

Figure 4.11.: Loss of lambs

The more intense colour used, the higher loss percent of lambs is registered. For example the pink areas, the loss percent of lambs is around thirteen to twenty percent. How much detail or information one wants, can be filtered from a menu. One can decide what map type to use and what information to show. As of everything, if one turns on too many of the information aspects, the map gets quite messy and complex. When multiple information boxes are filtered on, each area gets to much information on it, and it isn't too easy to see what is wanted. An example of this can be seen in Figure 4.12, where five different grazing area visualisations tries to fit onto the map. As one can see, the info gets messy and there isn't easy to get some reasonable information out of it. This is something which will be tried to prevent.

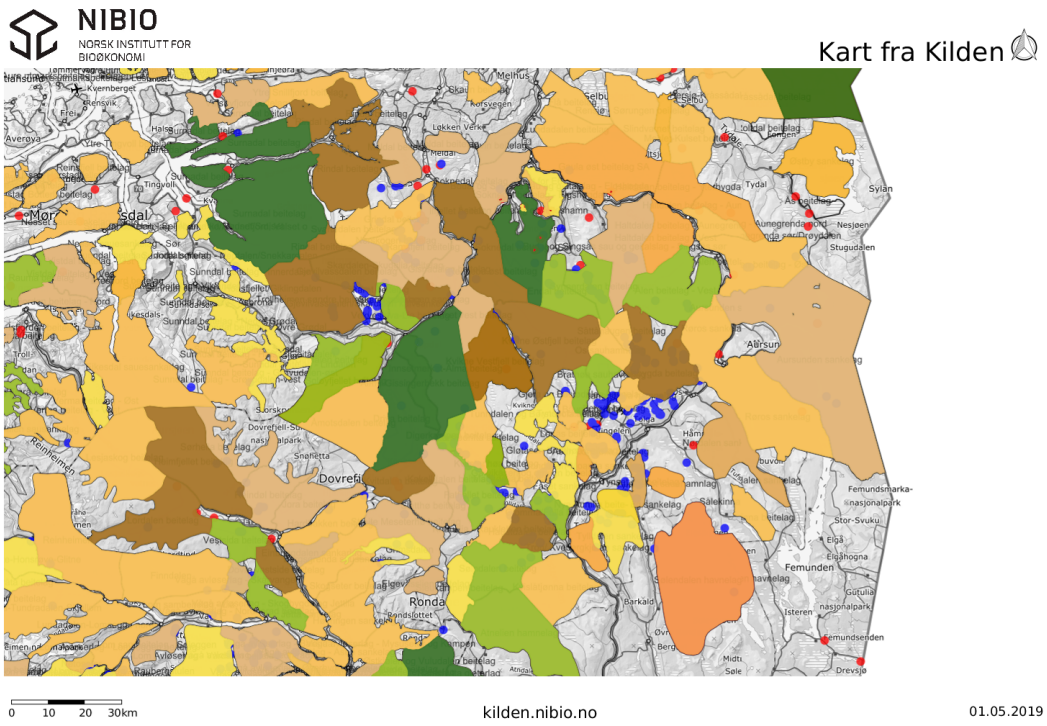

Figure 4.12.: Complex information on the map

Data Type

As already mentioned, the data from NIBIO can be downloaded freely for use as long as the data isn't behind a login page. This means that the grazing maps information are free too download and to be used in our application. The data can be downloaded from their website [29], and one decides which area to download. The map data comes in a format called shapefile, which can be converted into a GeoJSON object.
For each grazing area, NIBIO has data for the following fields which are interesting for our case.

- Grazing area location
- Year the data is collected
- Loss of sheep and lambs
- Number of sheep and lambs in the area
- Density of sheep and lambs in the area

4.2.2. Rovdata

Rovdata is the national supplier of population data of predators. In the year 2000 there was decided that Norway needed a national monitoring program for predators [30]. They

monitor the five animals mentioned in Section 3.4: wolves, wolverines, lynx, bears and golden eagles. The monitoring program's main goal is to fetch monitoring data and generate good population data for these animals. Rovdata publishes yearly reports which contains generated population data based on observations made. The monitoring of the different animals are mainly done by mapping footprints of the animals during winter, DNA-tests from hair and excrements, and the monitoring and counting of newborn predators. It varies how they are tracked from each animal to another. Rovdata cooperates among others with NINA, which we will look into in the next section, where they analyse the DNA-tests harvested.

4.2.3. Norwegian Institute for Nature Research

"Norwegian Institute for Nature Research (NINA) is an independent foundation for nature research and research on the interaction between human society, natural resources and biodiversity" [31].
The institute keeps track of multiple elements that are related to the nature. One of their research fields is on the population of predators, and other animals and species as well. NINA gets their population data on predators from Rovdata. As NINA gets the data, mainly DNA-tests and population data, they can do their analysis on the data so one can get a reasonable picture of how large the populations are for the different predators. After the data is set and validated, NINA offers some of their datasets to the Norwegian Biodiversity Information Center (NBIC) [32].

4.2.4. Norwegian Biodiversity Information Center

The Norwegian Biodiversity Information Center (NBIC) mainly keeps track of many of the species which can be found in Norway [33]. As they know about 44,000 species in Norway today, there are a lot of species to hold track of. For some of the species and animals, there are possible to fairly keep track of their population. NBIC cooperates with a lot of different companies, institutions and research institutions in Norway. The data span from research data delivered by NINA [32] for example, or observations done by nature interested people and volunteering organisations. As they have a portal where everyone can register animal observations done, their data foundation is quite solid. Observations of animals which falls under endangered species, will also be quality assured. Before the year 2005, these data were difficult to fetch as they were spread out on different systems and organisations. The goal of the government was that these data should be easy to fetch for everyone, like the common man and research organisations for example.

Map Solution

As described, NBIC sits on a lot of data, which can be interesting for a lot of purposes. These data can be fetched from a map solution which they have created [34]. The map solution can search for all species which are in their database, and show them

on a map. In Figure 4.13 one can see the five predators which hunts sheep. These are based on observations done in the last five years. This can be direct observations, DNA-observations or based on tracking.

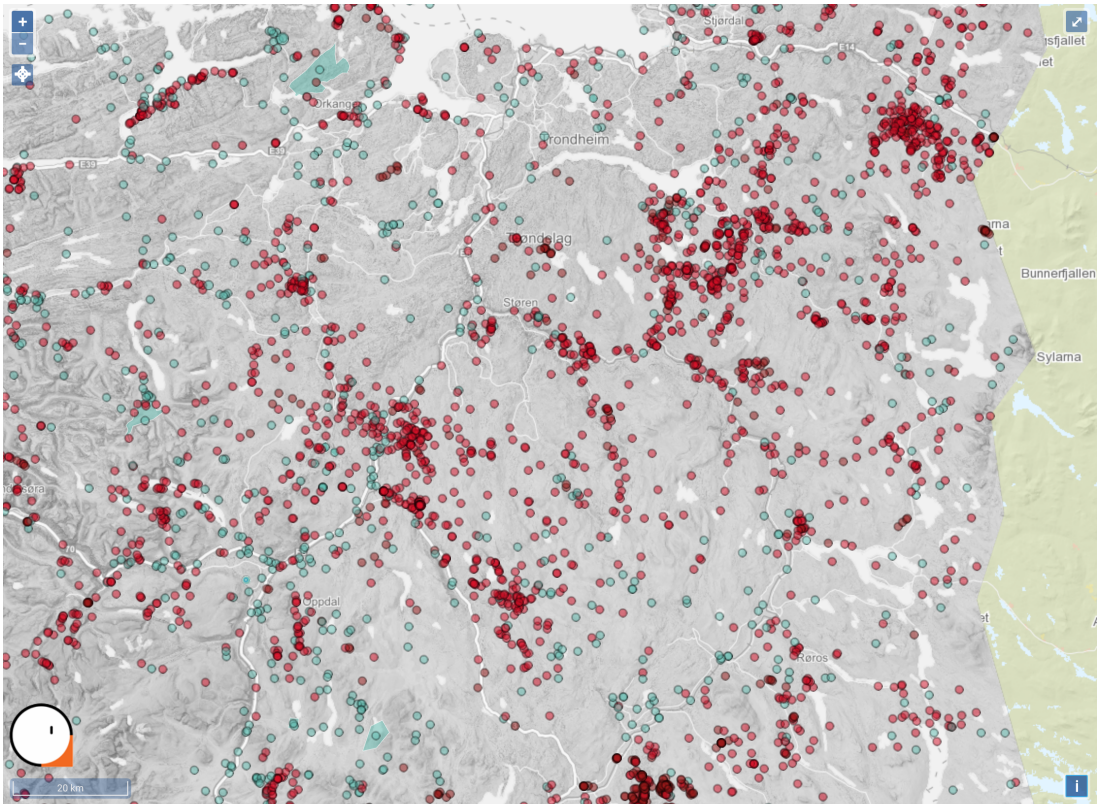

Figure 4.13.: Map of species in NBIC

Each dot on the map represents a finding of one of these five predators. Each colour represents how threatened the species is. For example the cyan points represent the golden eagle as it is not as threatened as the other predators selected. The other ones have shades of red, which means they area threatened in one or the other way. The size of each point represent the number of observations done in the point. In this example, all points have the same size, which entails that there is only one observation done in each point.
When focusing on the visual aspect of the map implementation, they have multiple ways of showing the data on the map. In Figure 4.14 each species has it own distinct colour.

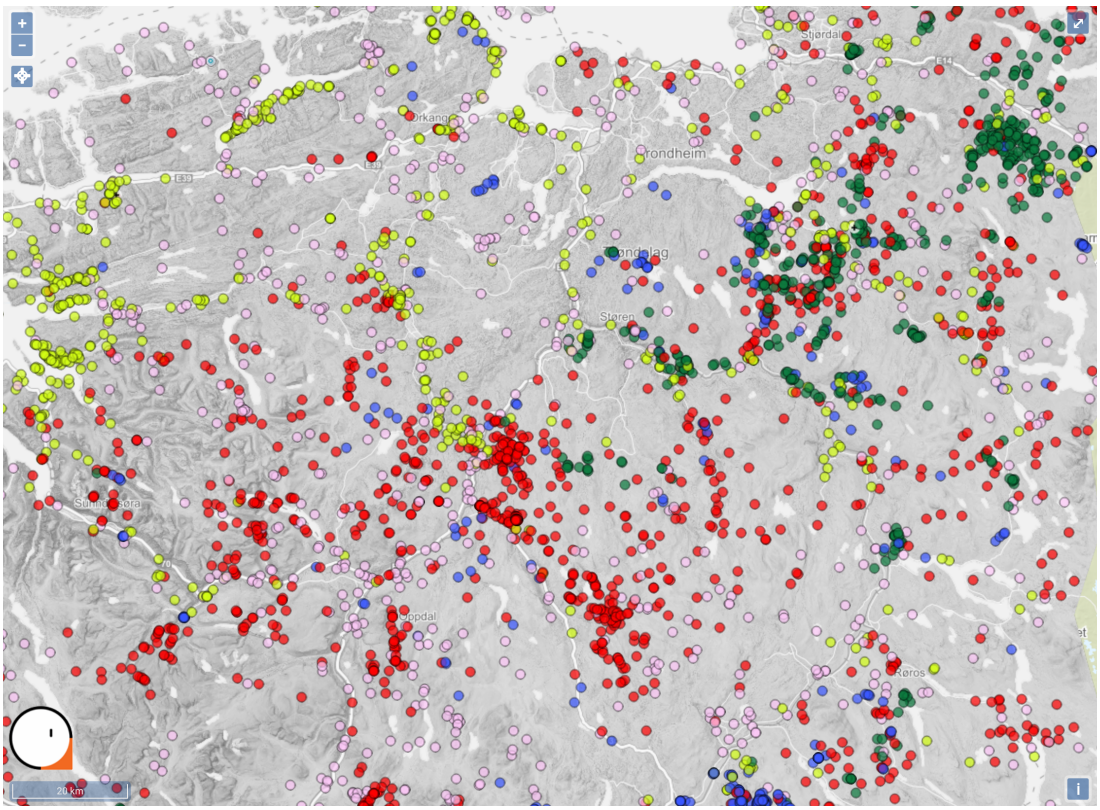

Figure 4.14.: Map of species distinction in NBIC

The application states that up to twenty different species can be shown at the same time. If more are used, the information can be too complex. This is probably due to colour distinction and if more species is to be showed on the same map it would be messy and some of the colours could look like the same colour, which makes the view complex.

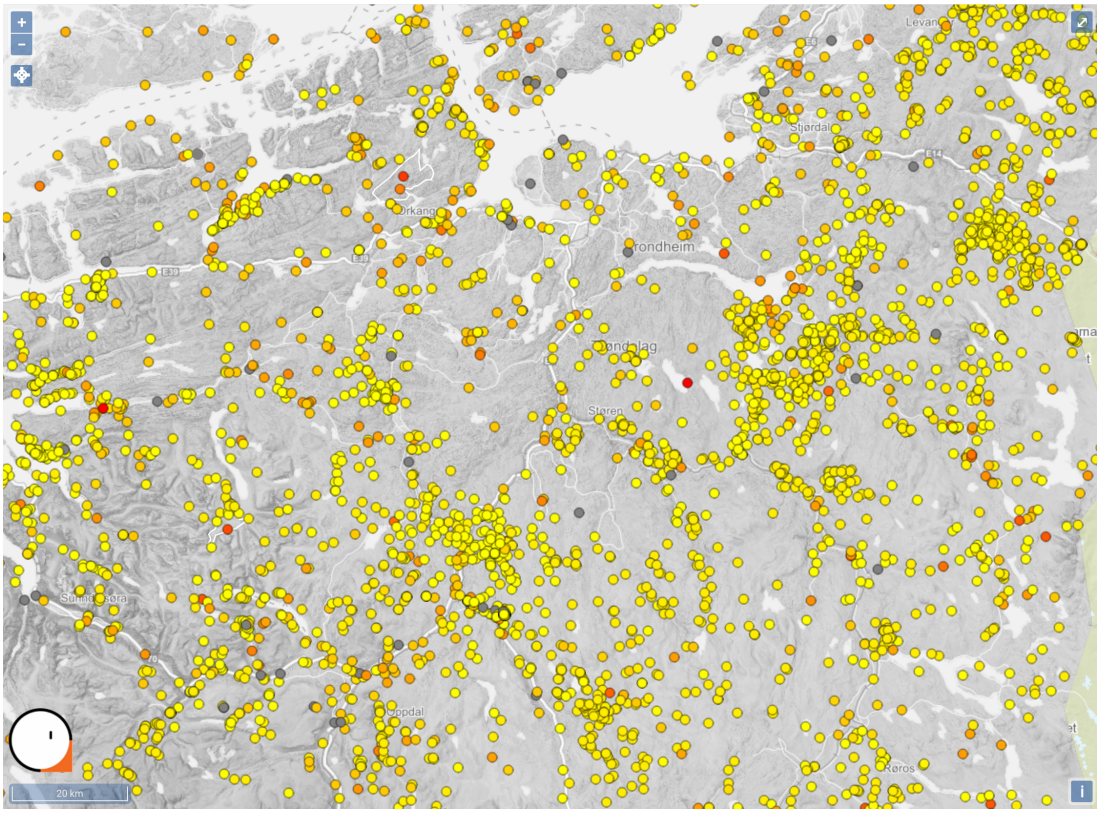

Figure 4.15.: Map of species accuracy in NBIC

The third visualisation technique they use is to show how accurate each observation is, which is seen in Figure 4.15. As the location of an animal may be insecure when plotting it on a map, some of the data points have an accuracy meter. The more yellow the point are, the more precise the observations is. Yellow indicates one meter accuracy, whereas red indicates five thousand meters of precision. The exact position of an golden eagle can be difficult to track for example.

4.3. Geographical Information Systems

Geographical Information Systems (GIS) are systems that map data and information to geographic systems and maps [35]. They use different standards to map the information wanted onto a map. All information can therefore be uploaded for usage as long as there are some geographical information in it. One of their core features are that they are quite general and have many different use cases. They can be used for governmental calculations and analysis, farming, everything which holds data that can be mapped to geographic data. The core feature of these systems is to map shapes, lines and points onto a map, which then can be done analysis on. Each shape, point and line can contain various properties. For example if one have points which have tracked a number of passing cars, each point can be shown large if many cars have passed or equivalent small

point if few cars have passed.

4.3.1. ArcGIS

ArcGIS [36] is an example of a partly free GIS software which can be used to demonstrate how easy one can use data using the GeoJSON format. All information and screenshots of the application are fetched using their web application [36].

Features

One can upload multiple formats as layers to ArcGIS. This include the following formats for upload:

- Shapefile
- CSV or TXT files
- GPX (GPS Exchange Format)
- GeoJSON

Example of Usage

As the web application can be used for free, a more easy version of the application, we could test the application with our data. In Figure 4.16 one can see how the sheep data can be used with a GIS application.

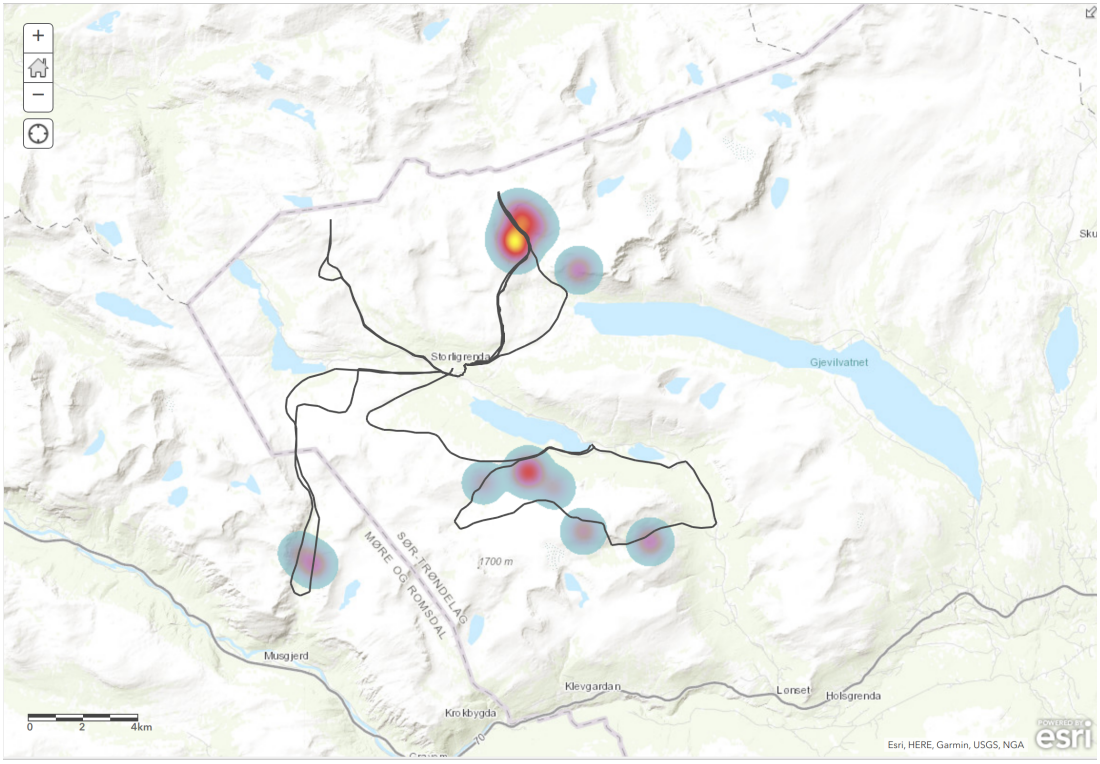

Figure 4.16.: ArcGIS with heatmap

In the example in Figure 4.16, the heatmap option is used. A heatmap is a way of visualising the density of data, whereas the warmer it gets (red and yellow colour), the higher density there are in the area. Instead of showing the actual point as a point, one can see the density of the information. In our case this can be used to see where there are observed most sheep during the season. Duplicates of sheep can arise, as the same flock can be observed at multiple occasions over a given period of time. Anyway, the visualisation gives an indication wether there are a high density of sheep or not in an area. If one could mix in how many observations and trips the data is based on, there could also be some interesting analysis and statistical aspects of this application.

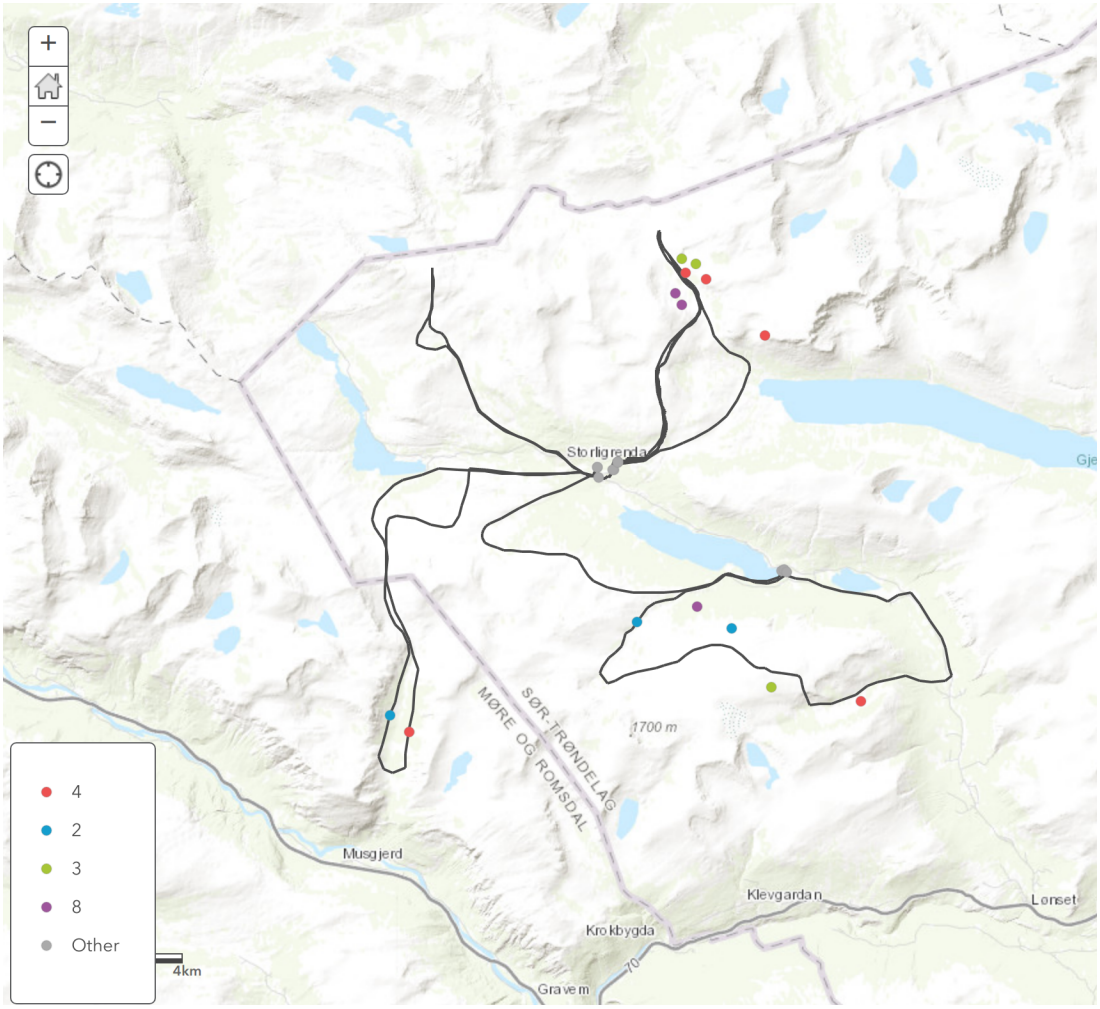

Figure 4.17.: ArcGIS with Types

In Figure 4.17 the visualisation is done another way. Each observation is displayed as a point with a colour. As can be seen in the legend down-left in the figure, each colour represents a number of observed sheep in the point. Green is for example representing observations with three sheep. This view can be interesting if there are a multiple observations that have the same number of sheep in them in the same area. This can be an indication on movement of the same flock, and the farmer can see the movement behaviour of the sheep.

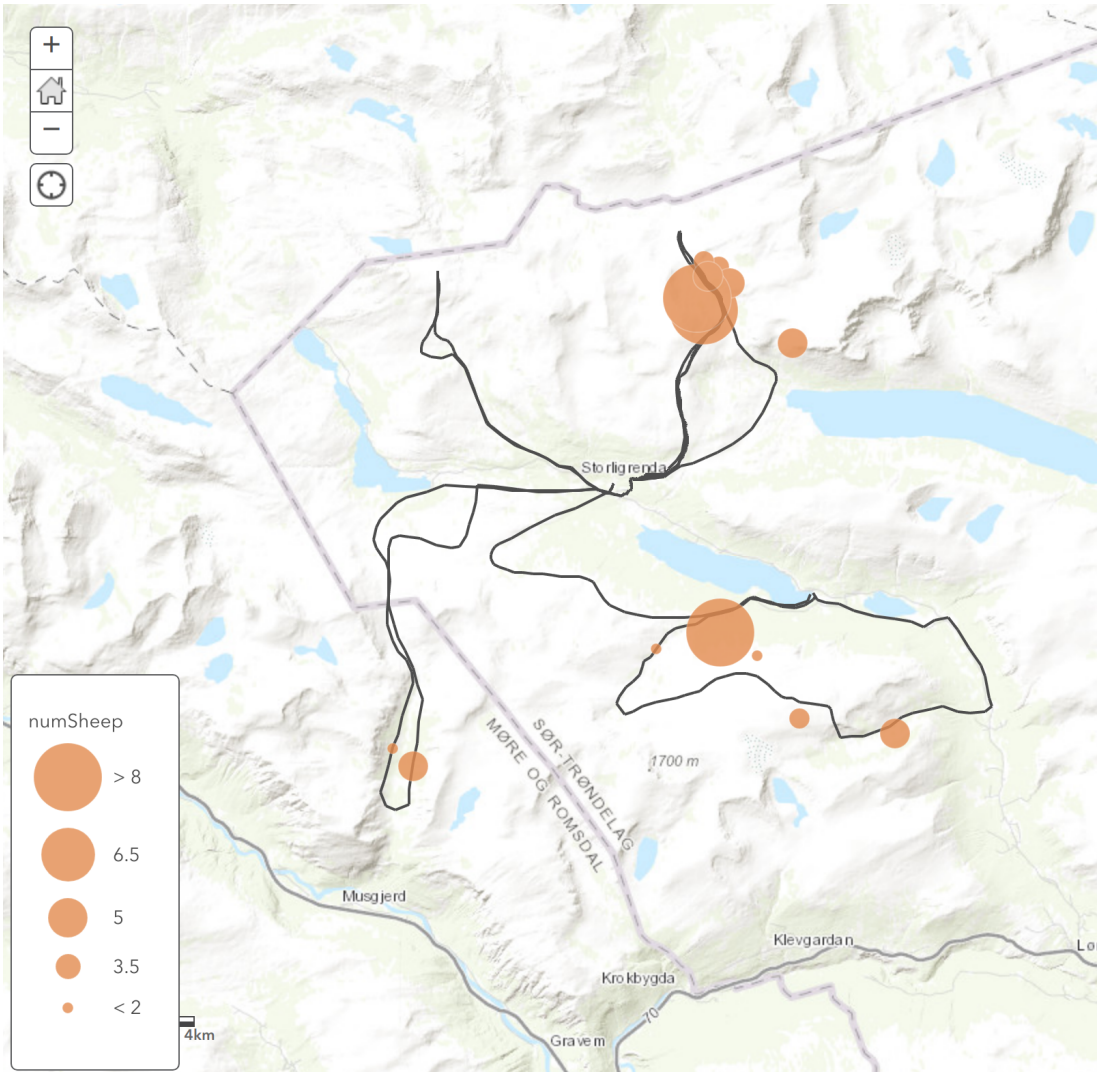

Figure 4.18.: ArcGis with Points

The last example from ArcGIS can be seen in Figure 4.18. Each observation is here represented as a point of which has a radius depending on how many sheep are observed. The larger number of sheep observed, the larger point. As the heatmap figure, this one will also give a reasonable indication on where there are observed sheep, and how many of them which are observed.

4.3.2. Why We don't Use an Existing GIS

Some of the features that could have been used to solve our problem or touch some elements of it, would have been solved using a GIS. One of the main problems with GIS is that they can be expensive, and they require some computer knowledge to use. They can be massive in functionality and will therefore be difficult to use for an end user, especially for our user base. The systems are mainly designed for professional users

which use such a program on a daily basis. This can make it more complicated for the end user as one has to upload the data for them selves.
The strengths of using systems like this, is that they have much of the analysis tools already implemented and supports standards like GeoJson. In the following list, there are mentioned some advantages of using such a system:

- Has much functionality already implemented.
- Can create relative custom applications from them.
- Supports standards for uploading data, such as GeoJson and Shapefiles.

There are some disadvantages as well, which is mentioned in the following list:

- Expensive to own a license.
- Requires some computer skills and knowledge of the system.
- No easy way to implement predefined cases (more of this in Chapter 8).

4.4. Summary

4.4.1. What is Missing

- A more detailed analysis on the data.
- Some of the views uses colours within the map that can be indistinguishable with the background.
- Especially Telespor has some weird usage of signs that isn't intuitive enough. For example the usage of a red and yellow star as the sheep location, wasn't quite understandable.
- The signs used on larger screens are not always as distinguishable from the map background as expected, and can sometimes be to small to be seen at a larger screen. This has to be taken into account as visualising in this research will be based on larger screens. This problem arises from both Findmy and Telespor. In Telespor's web application it is almost impossible to see the stars on the map, and therefore difficult to retrieve any reasonable information from it.
- The use of interactivity of the analysis and data visualisation could have been more interesting.

4.4.2. What to Include from these Systems

- The ability to choose between multiple map types.
- The usage of grazing maps can be quite handy as one can see how many sheep there are supposed to be in an area.
- The ability to fetch all animals in a given area. In this case it will be all observations done in a given area over multiple trips, and predator data.
- The usage of symbols and signs – for example the usage of an exclamation mark to indicate that something is wrong, or the usage of a sheep icon.
- Retrieve information from a given time period.
- From both NIBIO and Findmy, the usage of shapes on the map, is quite useful and understandable. This to mark where things are on the map and a definition on the density of the lambs using colours.
- The use of filtering from NIBIO where one can decide what to show on the map – such as which animals to show. The freedom of choice is something that can be usable.
- The ability to fetch an information box when clicking on the map could be of interest.
- The application which will be developed will be a GIS, with inspiration from the functionality of ArchGIS. The application's main focus will not be wide usability and functionality, but will be more pointed towards sheep farming. The application can therefore be more specialised and use it's focus on usability for farmers.
- Reactive and interactive maps.
- Try not to put too much information on the map as it can get messy and complex. In other terms, if there are too much information visualised in the map, it can be difficult to extract some reasonable information from it.
- As NBIC does it, at most twenty different colours can be showed on the map at the same time. This is probably a good estimate on how much information which can be distinction between different aspects and information.
- Collect different data into on specialised platform.
- Multiple visualisation techniques of observations, such as different sizing of points and heatmap.
- The usage of colours, especially for same animals, whereas white could be sheep.

5. Data

This chapter will mainly focus on what kind of data and it's models to be used in the application. What format is the data on, how it is saved, and how the data is retrieved. The data that are described are the data that mainly will be generated by the farmer when on trips. The data formats are the same as used in the specialisation project. One of the key aspects is to find usable features which have this data as it's base. How should these data be visualised such that the sheep farmer can use it to achieve their goals. For this case, one needs to visualise and find proper design alternatives for the data visualisation. This chapter will look into the concepts used in the application, both from the specialisation project and used in the developed prototype.

5.1. Data Model Concepts

From the specialisation project, the application focused on three data structures which is described in the Table 5.1.

Data type	**Information**
Trip	One of the main concepts of the application. When the farmer is going out to look for sheep, he starts a trip. The trip contains the farmer's walked positions and all observations done in the area. Many trips can be conducted in the same area for example with different routes and observations.
Position	A point which uses the longitude and latitude as it's location. Using the Decimal Degree System described in Section 2.6.
Observation	An observation of either a flock of sheep or predators. Points of interest during a trip. Contains the position of the observed animal and the position of the farmer. If the observation is done with a distance less than 200 meters, the observation is marked as detailed and more information can be filled out.

Table 5.1.: Data Models

5.2. Raw Data Format

The data format is therefore based upon the data format that was designed in the specialisation project. The raw form of the data is based on the concepts described in the last section. The data tracked are on the following format:

```
Trip = {
  "name": "recognition name",
  "start_time": "starting time of the trip",
  "end_time": "ending time of the trip",
  "positions": "tracked locations for the farmer, the route
    walked",
  "observation": "all observations seen during the trip"
}

Sheep_observation = {
  "observed_position": "the location of the farmer when
    observing",
  "position": "the location of the flock"
  "num_sheep": "total number of sheep in the flock",
}

Predator_observation = {
  "observed_position": "the location of the farmer when
    observing",
  "position": "the location of the predators",
  "animal": "what animal observed",
  "num_animals": "total number of animals observed"
}

Position = {
  "lat": "the latitude of the point",
  "lng": "the longitude of the point"
}
```

The position that is tracked, uses the phones inbuilt GPS tracking sensor. If the observation is done from a distance, the position set by the farmer can contain an error as the farmer does this manually. The usage of maps with contour lines, tries to prevent this by making it more accurate, but some error can happen. This is in most cases neglectable, as the observed flock can move it self over large distances.

5.3. Data in the Prototype

As the application were developed to be used on a mobile phone, there should be easy to use when on pasture. The data is tracked for each trip, as only one trip can be active at one time. After the trip is finished, the farmer can decide to synchronise the trip with a server so that the trip later can be viewed on a larger screen for more statistics and analysis.
As this visualising tool will be developed during the thesis, the sheep aren't set out on pasture before May, according to the expert. It will therefore be difficult to actually test the application with a farmer and then use the visualising tools afterwords with real data. The data will therefore be based on trips done by the expert. There will be made a tool in the application where one can mock trip data, such as the trip route and observations seen. The data will not be hundred percent correct and accurate, but they will be representative for the usage. As the purpose of this thesis is to find an application that can be usable for a sheep farmer for analysis purposes, this can be tested without completely accurate and correct data.

5.4. Mock Data

The data which is to be used in the application during the tests, are based on trips conducted by the expert. Visualisations of these trips can be seen in Appendix A. There will be generated nine mock trips within the system – three conducted in 2018 and six conducted in 2019 with various content. All trips are completed in Oppdal, Norway, so all trips are conducted in the same area. Some trips will include multiple observations of sheep and predators, and one will contain only the trip route wandered with none observations done. The trips Langtur, Turen, Høyturen, Tur fire and Tur seks, are all conducted in the same area around Hyttdalen in Oppdal, seen in Table 5.2.

Name	**Year**	**Observations**	**Number of sheep**	**Predators**
Langtur	2018	5	22	1
Turen	2018	4	13	1
Høyturen	2018	5	16	2
Tur èn	2019	4	11	0
Tur to	2019	1	8	0
Tur tre	2019	0	0	0
Tur fire	2019	5	19	1
Tur fem	2019	2	6	0
Tur seks	2019	4	15	1

Table 5.2.: Mock Data

6. Early Sketches

After the research of requirements, goals and alternative solutions, it was time to get creative and draw sketches of different visualisations. This follows the Interaction Design principle where one is to find alternative designs for the prototype. This includes simple paper prototypes, which is described in more detail in Section 2.3. The sketches are mainly based on our imagination, but with inspiration of the applications looked into in the State of the Art, in comparison with the expert. As the sketches can let us find alternative designs, which can be relied on when starting the development of the prototype, time can be saved. As the creativeness when designing a user interface can be lifted when using sketches and paper prototypes, this were a way to go. There are mainly sketches which shows the data displaying page, as this were the most important. Interactive paper prototypes weren't focused on as there was decided not to include user interaction in the start of this process.

6.1. Visualisation of the X Last Trips

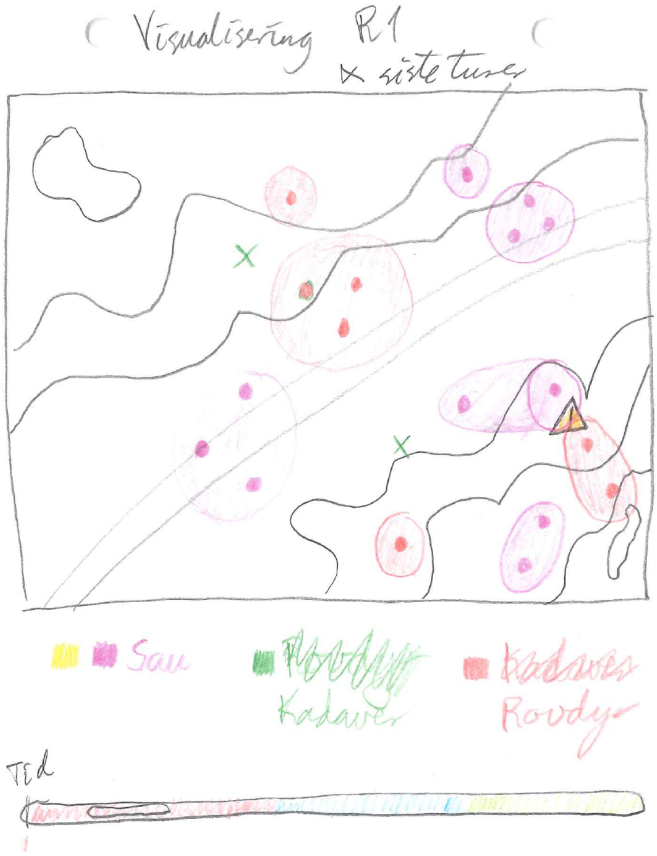

Figure 6.1.: X Last Trips

The sketch in Figure 6.1 is based on the X last trips conducted, where X is a natural number. One of the main ideas of this sketch is that similar observations are mapped together. Each dot is surrounded with a larger circle, which was thought to be the area which was likely to contain that given flock of sheep. The red ones are predators and the pink ones are sheep. The sketch does not take the difference between each trip in consideration, besides the timeline in the bottom of the sketch. The timeline is scrollable and observations will be highlighted when moving it. In Figure 6.2 there is the same visualisation with the wandering route of the farmer. The location of the farmer can also be time tracked using the time bar at the bottom.

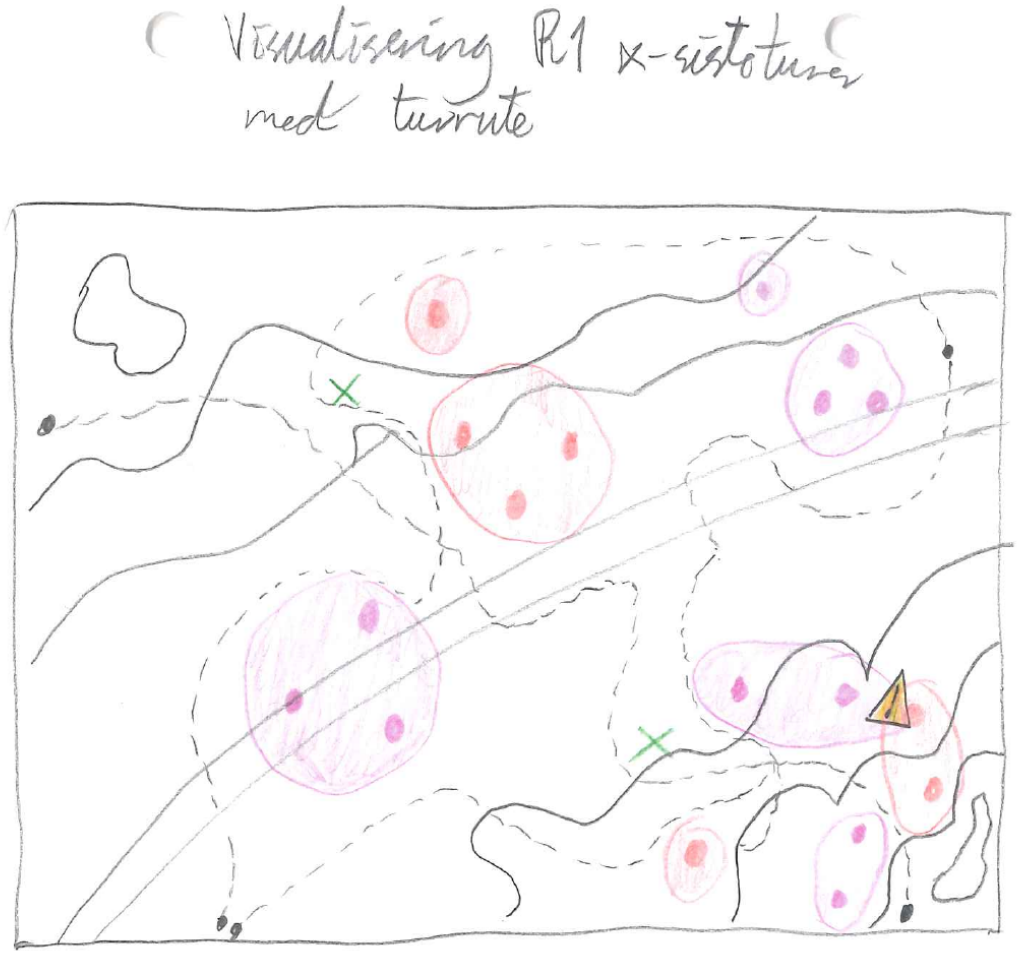

Figure 6.2.: X Last Trips with Route

The sketch seen in Figure 6.2 is the same as the one in Figure 6.1, with the difference that the trip routes are shown. Same functionality is meant for this one as the previous. At the right of the map in Figure 6.2, there are an intersection of observed predators and sheep. One can see that there are a sign with an exclamation mark, which indicates that this is a possible area of interest, as there are both observed sheep and predators here.

6.2. Different Graphs

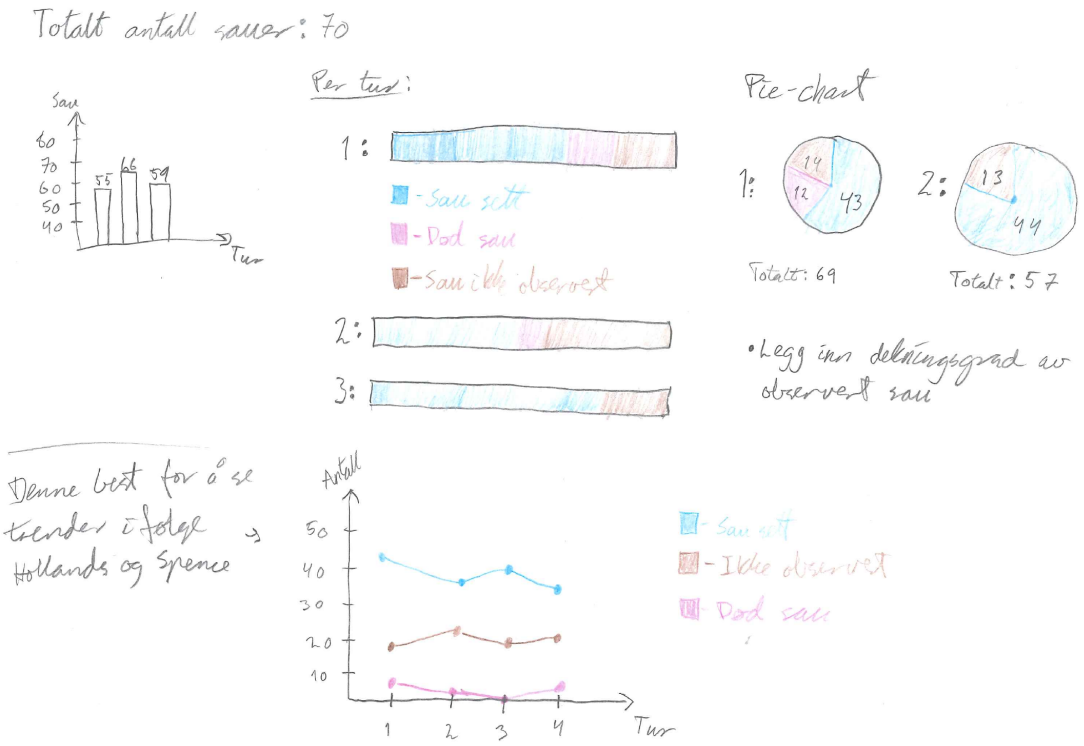

Figure 6.3.: Different Graphs

In Figure 6.3 there are shown different graphs which can show interesting data for the trips conducted. The graphs are based on different graphing techniques, but the most interesting one is the one in the upper left of the figure. This graph shows the total amount of sheep observed for each trip. The other ones take the amount of sheep not observed in that area. Together with the expert, this was seen as not so relevant as the number of sheep not observed can be quite high compared to the ones observed.

6.3. Different Trips with Timeline

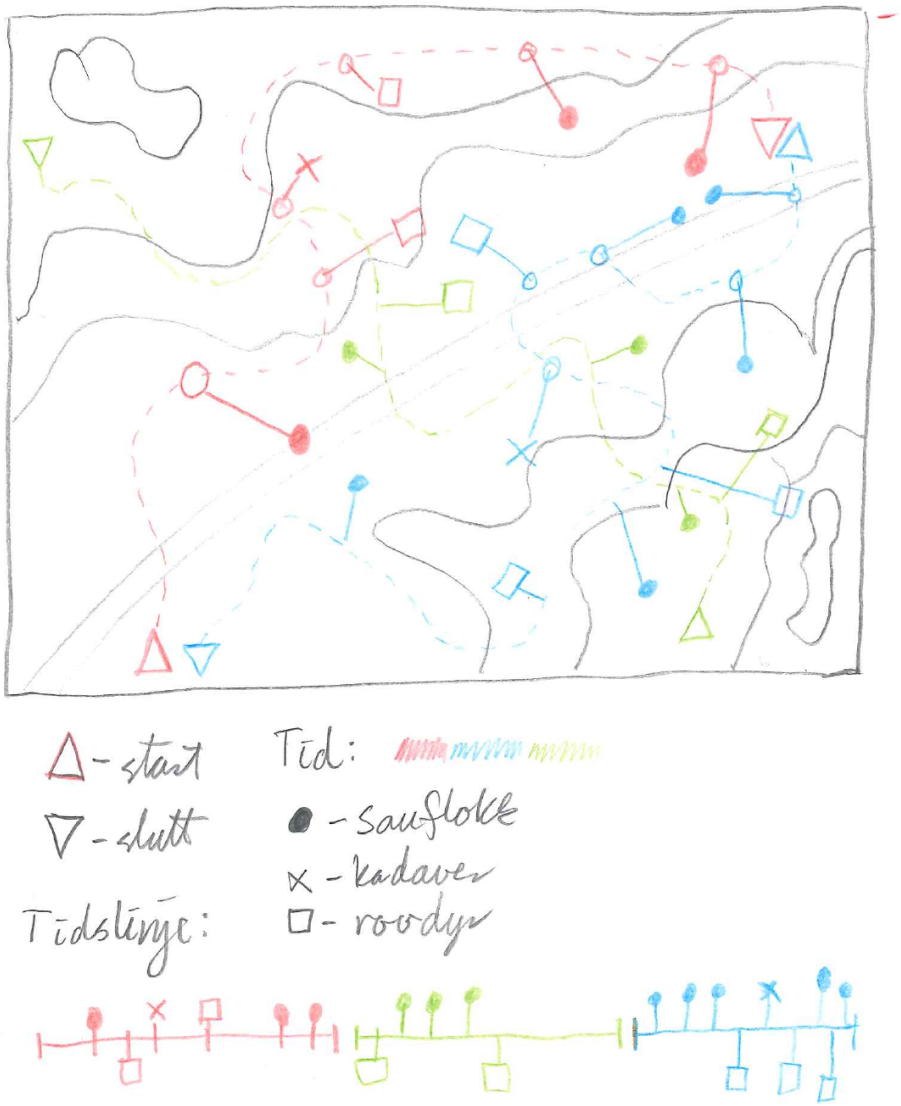

Figure 6.4.: Trips with Timeline

The sketch in Figure 6.4 shows a map where each trip shown on the map have an unique colour. All information which is included too a specific trip, have the same colour, which is observed sheep (circle), predator (square) and cadavers (cross). The trip route which is walked is also shown for each trip using the same colour. At the bottom of the sketch, there is shown a timeline showing when an event during a trip happened. This includes all observations done. The trips are also sorted in chronological order based on when conducted.

6.4. Similar Trips

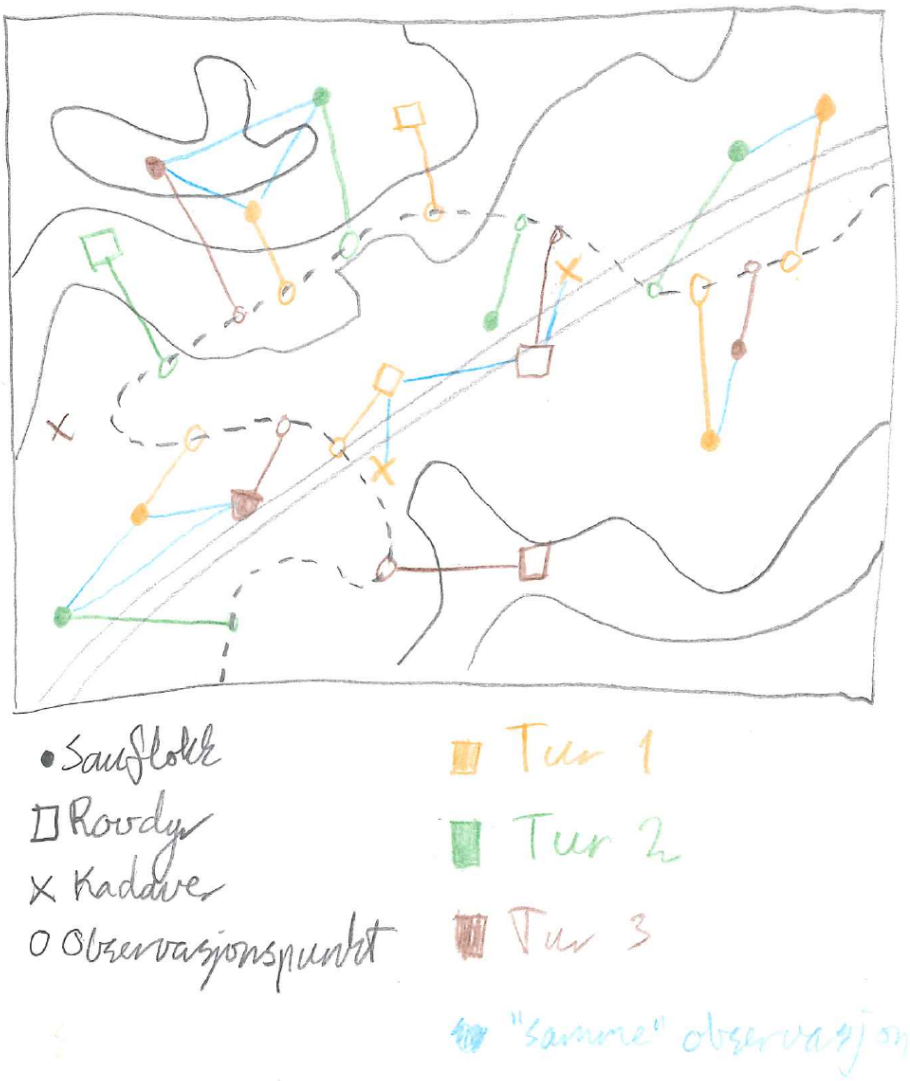

Figure 6.5.: Similar Trips

Figure 6.5 shows a sketch which follows the same principle as the last sketch. The meaning for this view is to show trips which are quite similar, in regards to similar observations and trip route wandered. The observations showed are based on different trips, using different colours. If some of the observations are mapped together to be the same flock of sheep over different trips, one can see that these trips have blue lines between them, indicating that these observations are the same. One can therefore see the movement pattern for a flock of sheep over time.

6.5. Movement Pattern

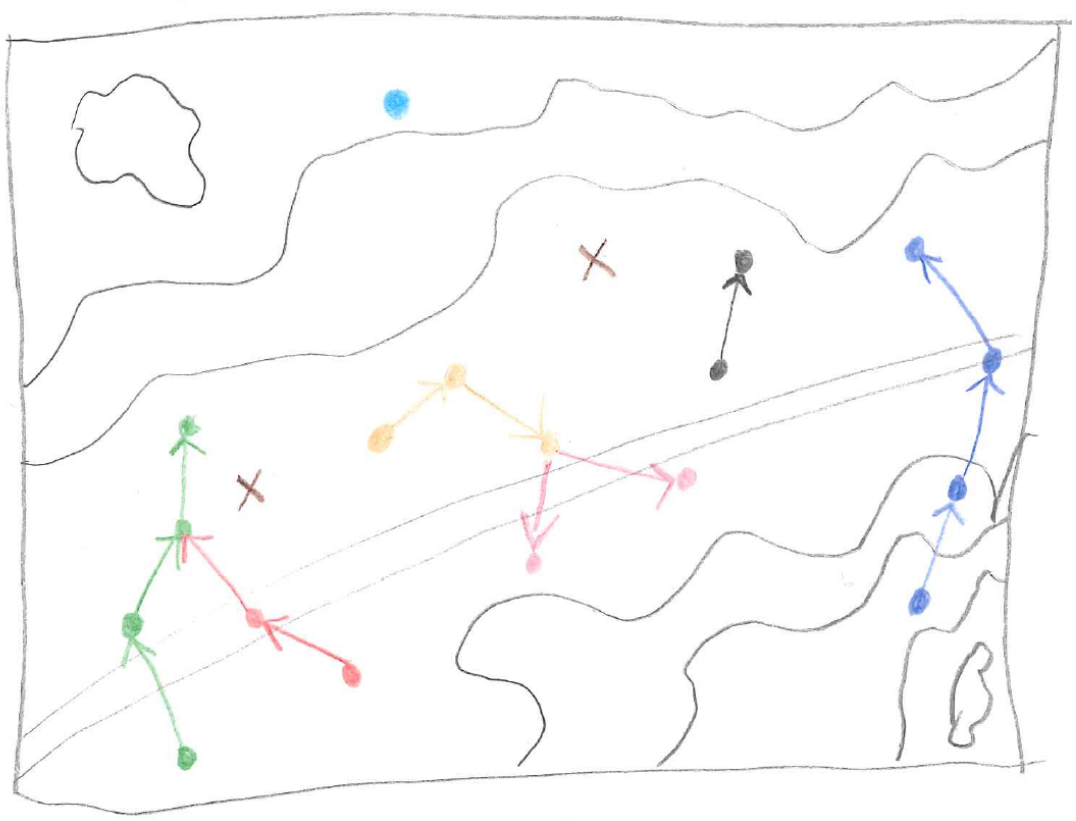

Figure 6.6.: Movement Pattern

Different from the other sketches, each flock has it's unique colour instead for an unique colour for each trip. The system has mapped all observations together in such a way similar or equal observations are mapped as the same flock of sheep over multiple trips. One can therefore see the movement pattern for each flock over time.
In this sketch there are three different scenarios. The first one, to the left, there are two flocks which merge into one flock. One can see that the green flock and the red one merges into one. The second is the one in the middle, where one flock splits into two separate flocks. The third one, to the right with blue colour, is the same flock over all observations.

7. Technologies

The application is split into two different systems, one for holding the information and one for the user experience. Both the frontend and backend application are built on what where developed during the specialisation project. Many of the elements of this thesis and the specialisation project share the same base for models and applications. There were decided to continue the development of this application instead of starting from scratch. Time could therefore be saved using the already set up environment. With minor adjustments on database models, endpoints and software packages, much of the code and functionality could be reused. As the authors of this thesis already are familiar with the technologies used in the specialisation project, less time would be needed to learn new technologies. One could therefore focus on finding a good solution for the problems instead.

7.1. Backend

The backend system's main task, is to save data managed in the frontend application, and hold data from other institutions saved in a database. As some of the data from NIBIO and Rovdata are quite large, with a size of approximately 25 Megabytes, there would have been unpractical to save this into the web application, as well as doing database operations in the frontend. As there was developed an API for this purpose during the specialisation project, one could continue to use that. The web application can take use of saving trip data and fetch data from the research institutions using the API. The backend is built with the framework Spring Boot [37] which is a JAVA framework for building applications such as an API. One of the reasons for using Spring Boot, is that it have a lot of functionality "out of the box". There is easy to setup an API in a fast manner. Such as database models, can be easy built and setup using Spring Boot. As the main purpose for this application is to have a functional web application, there was wished not to use much time on this during development. The choice for using Spring Boot where therefore easy. The database models where therefore built using Spring Boot, which then connects to a MySQL-server instance for consistent data storage [38].
The data are sent over Hypertext Transfer Protocol Secure (HTTPS) protocol, which is a secure version of the most used transportation protocol over the internet, HTTP [39]. The communication between the frontend and the backend are secured, and the API are locked using Basic Authentication [40]. The API asks for a password to get access to the application, so the data is secured and managed in a secure way.

7.2. Frontend

The program developed in the specialisation project, was a single page application, which could be split into modules. The continuation of creating the analysis program as a web application where therefore relevant. One could for example have created a native computer program which for example could have been faster in terms of speed in calculations, but this wasn't seen as a big concern. One problem would have been the lock of operating system, if one haven't used a technology as Java, which works on different operating systems. As a web application can be run on almost any device which have a web browser on it, the availability of the application is much larger. It can then be run on any operating system, such as mobile devices and tablets, as well as computers. Heavy calculations and saving of data can still be done in a fast manner, as the data is uploaded to the server and saved in the backend system, which uses Java and MySQL. The reuse of code and functionality for the frontend application was also a big plus to continue to develop a web application and to continue too use the frameworks.
As there are some information that is to be showed to the user, the main visual focus will be on larger screens. The application is developed using reactive components, and can therefore work on smaller devices, such as mobile phones. The development on small devices is not mainly relevant. Larger screens can include regular computers and tablets. Tablets are easier to carry out in the field rather than a computer, and have larger screens than mobile phones, if one want to use such a system out in the field. As one can fit more information on a larger screen, and the information are meant to be used after a trip is conducted, the usage on larger screens were prioritised.
The front end application works as a SPA described in Section 2.5, all HTML and JavaScript are loaded once accessing the site. The only thing that isn't necessary to load when accessing the application, is the data, which can be retrieved afterwords. As one of the demands of the specialisation project was that the application had to work offline, saving data locally was a demand. Luckily, most browsers have support for local storage, which the application uses to save trips and data connected to it. This data can be synchronized with the backend server for more consistent storage. Some elements of the application could have used lazy loading, but this wasn't prioritised as our main goal is to find functionality and not creating a finished product.

Main Frameworks

As there exists a lot of frameworks for JavaScript, which are contributed by numerous people on the internet, there are a lot of packages, frameworks and libraries that can be used for an easier application development process. The frontend application is developed as a single-page application, which are applications that load one single HTML-file and the content on the page are dynamically updated using JavaScript as the user interacts with it. This means all routing and loading of data are done on the client side instead of loading a new HTML-page for each view (Section 2.5).

VueJS

The application is mainly built on a JavaScript framework called VueJS [41]. VueJS is a framework one can use when creating user interfaces for a web browser. "...Vue is also perfectly capable of powering sophisticated Single-Page Applications when used in combination with modern tooling and supporting libraries." [42]. This framework was used during the specialisation project and worked nice for it's purpose. As it is easy to use with other frameworks and libraries which exists, this was also an important factor. And that one didn't need to rewrite the entire application and use more time on actual developing.

Leaflet

Leaflet is a tiny and simple JavaScript library which is open source and contains easy and interactive maps [43]. Leaflet supports the usage of maps from Kartverket and has easy support for the Decimal Degree system. As this is a widely used library for map usage in web applications, there exists many different libraries and packages which have support for Leaflet. In the application developed, there are used packages which implements heatmaps, clustering and offline maps. There exists good documentation on how the library works, which is also a valid reason why this library was chosen instead of others. As Leaflet has some builtin functionality, such as points, polygons, markers and usage of geoposition from the browser, it makes it easy to display elements on the map. Also used in the specialisation project.

Material Design

For the user to have a good user experience using the application, it is important to have a understandable and consistent design for the application. According to Shneiderman's first golden rule in Section 2.1, one should strive for consistency. This can be achieved by using CSS and reasonable placements of icons, buttons and interactive elements. As the application should fit different screen sizes, such as mobile phones, tablets and computers, the application should be reactive and have different layout for different screen sizes. Instead of creating all of these from scratch, there exists different packages which have implemented these kind of functionality already. One of those is Material Design [44] created by Google, which was used in the specialisation project. They have embedded support for reactivity in applications, such as different layouts for different screen sizes. As well, it has some built in support for predefined layouts and elements, such as buttons, fields, menus, which mainly are reactive – the way they look can alter if the screen size changes. In other terms, the application can look and feel better when using such a library. As Material Design have a large list of icons which can be used, this were also a factor in choosing this framework as the main design framework.
As there exists predefined buttons, menus, and other UI-elements, it is simple to reuse this for a consistent user experience. The design package contains many different icons, which are created and used by Google, and therefore can be familiar for many users. For example a walking person icon will in our system indicate that this is a trip, and

is used throughout the system. This also follows Shneiderman's second rule. The strive for usability, which can hold for many user groups.

Highcharts

An easy implementation of charts were something that were needed. Highcharts [45] is a framework which have support for an easy implementation of charts in a website. Highcharts have multiple products, such as maps, graphs and stock graphs. We are only using the graph section for the application, such as regular column graphs and the timeline package. Their graphs are quite interactive and clickable which fits to the demand of an interactive and easy solution. Less time could therefore be used when implementing such graph solutions. The reason that we wanted to use this solution for the graphs, is that we are familiar with the technology, as well as it's wide support, demos and documentation [46]. A lot of time could therefore be saved instead of implementing such functionality from the bottom.
There could also have been used their map solution instead of Leaflet, but this was not wanted as the implementation from the specialisation project were done using Leaflet. The time was wanted to be used on development of new features, and not re-write the implementation from the project, there were decided not to use their map solution.
The technology can be used for free, according to their Non-Commercial license [47], as it is used for demonstration purposes on a university.

8. Prototype

The main focus were on designing functionality with respect of a simple and easy design. As most components used are standard components from Material Design [44], many of the stylings are done using it, with some custom CSS. As the visualisations on the map were the most important ones, these were prioritized. Most of the visualising aspects designed and implemented, are based on the scenarios described in Section 3.5 and the sketches drawn, which is seen in Chapter 6. Other visual elements are based on both the Bell Tracking Systems and Geographical Information Systems, such as heatmapping, usage of polygons, points and lines, and others, which is described in more detail in Chapter 4. This is the prototyping phase, which is the third phase of Interaction Design discussed in Section 2.2.

8.1. Language

The application's main and only language is at the moment Norwegian. As the prototype is to be tested by Norwegians, and that the program is, for now, meant for Norwegian sheep farmers, there were decided to only include the Norwegian language for this prototype. On a later iteration for this application, there could be included the support of other languages as well. All screenshots of the application and it's information will therefore be in Norwegian.

8.2. Interactivity

As the seventh law of Shneiderman (Section 2.1) describes it, one should strive for the user control of the application. We want to achieve this by having a interactive application, where much can be altered and changed. This will as well follow the third and sixth rule of Shneiderman, where the user should get good feedback when using different elements, as well as one easy can reverse actions done in the application.
During the development, this was therefore one focus that were taken into account. Keep the user in control of the application. As of GIS, discussed in Section 4.3, the ability to decide for one self how to display the data, makes the solution more robust. As the application is developed as a single page application, time used on fetching HTML, CSS and JavaScript are only done once, hence less network bandwidth is used and less time on loading. All routing are done inside the application, which also helps with the interactivity of the application. There will therefore be a larger focus on the application, and not using time on loading different features and functionality. When data is to be

fetched from the server, this is well indicated with a loading bar appearing at the bottom of the map. This is discussed later on.

8.3. Overview

For this prototype, there are one module which will be tested. This is the part of the prototype which contains the functionality which we want input on. The module contains two views, which are the predefined cases overview, and the map view. As the application is a further development of the application built in the specialisation project, there exists different modules in the prototype. As of now, it is built into different modules. These modules can be seen in Figure 8.1. The whole application contains the top bar at any time. To the left is the name of the application, at the right there are three symbols. The house symbol routes the user back to the home page, which is seen in Figure 8.1. The cog is sent to the settings page, and the red symbol in the middle indicates whether geolocation is turned on in the application. This one is only relevant for the harvesting module of the application which uses the clients position to track the walked route. The position of the user isn't relevant for the analysis module.

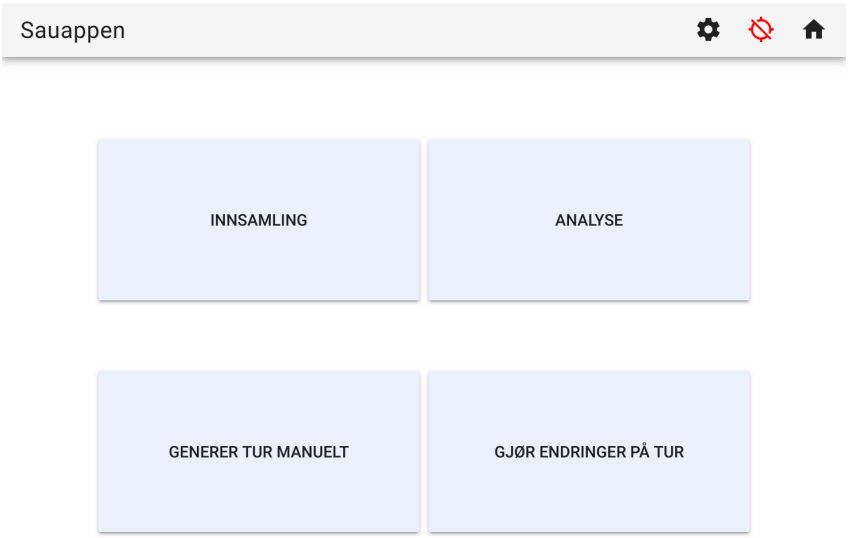

Figure 8.1.: Home View of the Application

When accessing the main page, one gets four choices where one can access:

- the data harvesting module – upper left in the home view
- the analysis module – upper right in the home view
- the manual generation of trips – down left in the home view
- the edit page for trips – down right in the home view

The data harvesting module, which was developed in the specialisation project, is where the farmer can go to start tracking a trip and register observations done in the field. The module consists of a list of all trips conducted, a map for downloading offline maps and the new trip generation part. As this module is not further relevant for this thesis, it will not be discussed more.
The manual generation of trips is a module which one can generate trips outside of the harvesting module. One sets the name, start and end time manually. After this is set, one can manually enter observations and the trip route to build a trip. Mainly, this part were developed so generation of mock trips could be done easily. The data used in the prototype is, as mentioned in Section 5.4, based on trips conducted by the expert. This module will not be tested, as it isn't a part of the core functionality of the prototype. Other aspects for the application are only for the purpose of maintaining the data which is to be used during the tests, and to get relevant information to show.
The edit page for trips is a module which one can edit an existing trip, such as date, name, observations and the walked route. This was developed for the ability to alter some of the data already created in the prototype. As well as the ability to change errors made when generating a trip. Same as the manual generation of trips, this module isn't relevant for the core system and will not be tested.
The last module is the *analysis module*. This the core module of this thesis and contains all functionality which is to be tested. The module is split into two different view, the predefined cases and the map view. The predefined cases holds multiple predefined statements which the farmer can click on. The map view holds the actual showing of the data, with different filters. The predefined cases will set the different filters according to what kind of information it will show. This will be discussed later on.

8.4. Map View

The applications main view, is the map view. One example of how it looks is visualised in Figure 8.2.

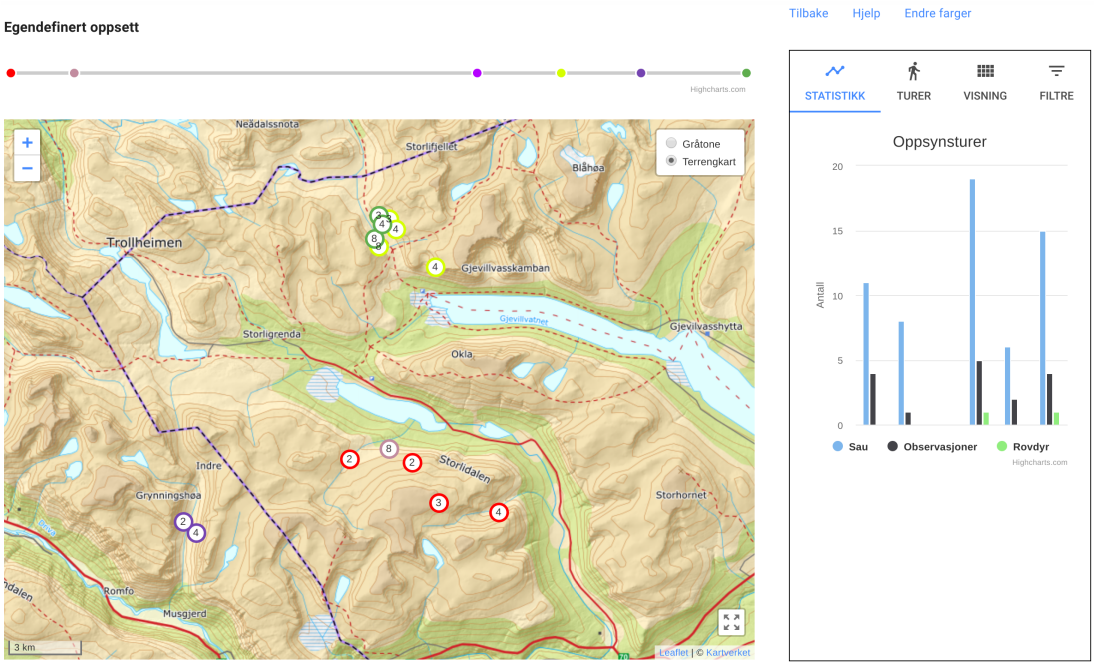

Figure 8.2.: Application's Main View

The map view consists of three different elements, which are the map where data is visualised, a timeline at the top which have dots for each trip conducted, and side view including the interacting part which can be seen at the right of the figure.
The timeline contains all the selected trips in chronological order based on their date of conduction.

Figure 8.3.: Timeline with Hover

Hovering over the point, which is done in Figure 8.3, a popup appears. The popup shows the time and name of the current trip, as well as how many sheep that were observed in it. If one clicks the trip point, the map will zoom onto that trip's bounds, which are in the decimal degree format. These are the largest and smallest point of the trip route walked, and observations done in the trip. The timeline is created using Highcharts. The timeline is based on the sketch which can be seen in Figure 6.4. We decided not to include the events which happened in each trip, as these data didn't have a timestamp on it, and it made the timeline a bit messy and complex.

Figure 8.4.: Timeline Compressed with Hover

Figure 8.4 shows the same timeline as Figure 8.3, but as a compressed variant. This timeline was implemented for cases where the trips showing having a large time gap between conduction. For example if trips from two different years are showing, the points in the timeline in Figure 8.3 will be too difficult to distinguish each point from another, and one gets a large space between them. This is seen in Figure 8.5, where it is difficult to see the proportion of time done between each trip. In such cases, the compressed timeline will be used.

Figure 8.5.: Timeline with Complex Data

8.4.1. Map Implementation

Leaflet [43] as the main map engine used for displaying the maps and information in the prototype. As Leaflet is only an intermediary of the information put into it, it doesn't contain predefined maps. The maps that are used in the application, are delivered from Kartverket [48] which have multiple maps available. The maps from Kartverket are free to use as long as they are referred to in the application [49]. This can be seen down right in Figure 8.6 where it states "©Kartverket" which also links to their web site [50]. The ability to switch between different maps are something that were a feature in both of the bell systems, in addition to the GIS software.
The application includes two different maps: a gray tone map and a terrain map. In Figure 8.2 one can see the terrain map in action. This map type contains contour lines, which where argued in the specialisation project to be important to distinguish where the sheep were observed. This is therefore included. The map also includes road information, locations of houses etc, so one can get much detail if one wants. As there are a lot of different colours on the map, and that it can get a bit messy with many different colours of trips, observations, and other data, there were decided to include a more simple map as the gray scale map. This map representation can be seen in Figure 8.6.

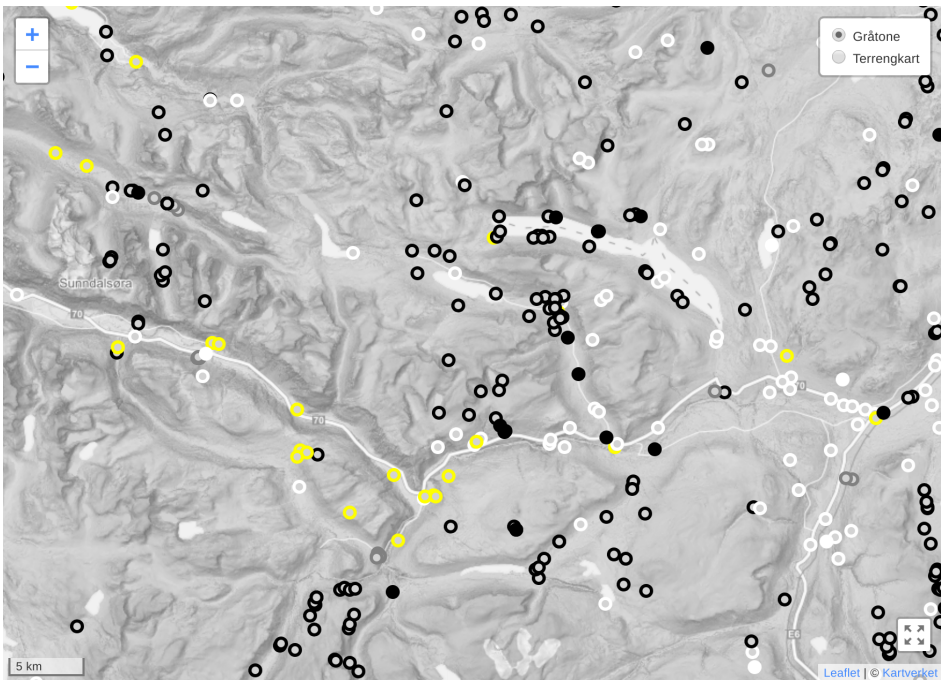

Figure 8.6.: Predator Data from NINA with Gray Map

As one can see there can be easier to distinguish between the actual map and the points drawn on it, as there are less disturbing elements in the background map. As points can be put onto areas with different colours, there can be a nice feature to switch between different map types. If one finds out that one needs more map types available, this can be added in a easy manner. Down right in the map, in Figure 8.6, there is a button with four arrows pointing outwards. When clicking this button, the map will zoom onto the whole area of selected trips. If one for example have zoomed far out from the area, this is an easy way back to the data.

8.4.2. Side View

At the right in Figure 8.2, one can see an interacting element which consists of three tabs. Each tab has a name, and an icon to symbolise it's meaning to make the tab more distinguishable. In the same way a tab in a regular internet browser, an internet page usually have a "favicon.icon" image to distinguish each tab from each other. The same principle is used here. The statistic tab, which is seen at the right in Figure 8.2, shows a simple column graph for all trips. Each "group" of the columns is a trip, which shows the number of observed sheep, the number of observations, and the number of predators observed. All these are based on the trips shown on the map at that time. How many trips that are showed on the map, varies, and depends on the scenario chosen, which is discussed later on. The icon used for the statistic tab is illustrating a graph. All icons used in the application are fetched from Material Design (Section 7.2). The graph used is based on one of the sketches made, in Figure 6.3, where graphs are included.
The second tab, contains a list of all trips which is filtered for this view.

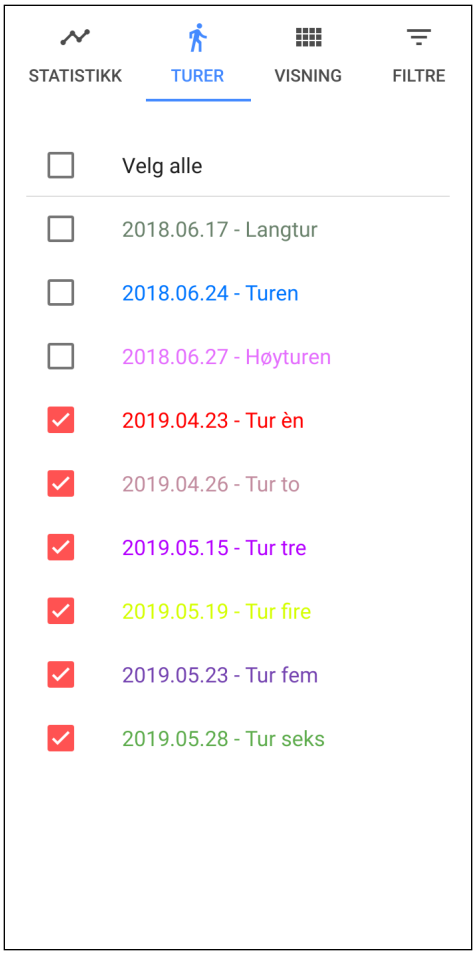

Figure 8.7.: Trip List

In Figure 8.7 one can filter out trips which are selected for this case. One can see each trip having a unique colour. One can therefore more easy distinguish between the different trips when the data is showed on the map. If the colours chosen for the trips are not as distinguishable as expected or wanted, this can be edited manually, which can be seen in Figure 8.18. The same colour is used everywhere in the analysis module. The colour represented in the trip list is the same colour on the map and on the timeline. There should therefore be easier to see where a trip is staying at on the map for example, and when it is conducted compared to the other trips.
For example one predefined case states that one only is to show trips from the last week. The trip list can therefore be at a lower amount than this one. Trips can be toggled, and if they are not selected, the trips' information will not be showed on the map nor in the statistics. It will also be hidden from the timeline. When clicking on the trip, not the checkbox, the map will zoom onto that trip's bounds – same as when clicking the trip point in the timeline. The information shown on each entry is the date the trip where conducted, as well as it's name. The icon used for the tab, is a person walking, which is to illustrate a person out walking a trip. The trip list is sorted in chronological

order based on the time of conduction.

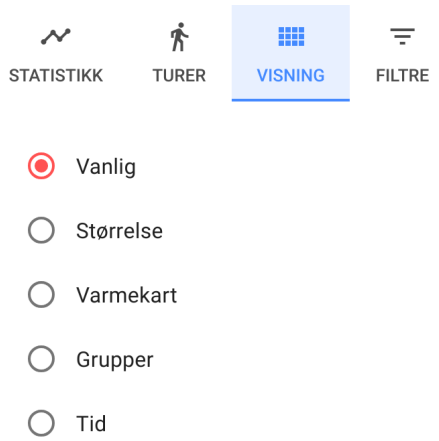

Figure 8.8.: Visualisation Choices

The application have two types of visualisation for observations, the main visualisation (herby called main views), which decides how the observation points is to be shown on the map, and the filters. Filters will be discussed in Section 8.4.3. The application consists of five different main views, which is seen in Figure 8.8. The different main views mainly decides how the observations are visualised. There are five main views which are implemented in the application:

- *The regular view*
- *The sizing view*
- *The heatmap view*
- *The grouping view*
- *The time view*

The background map is selectable independent of which main view is selected. The images which shows the main views, uses a random map background.

Regular View

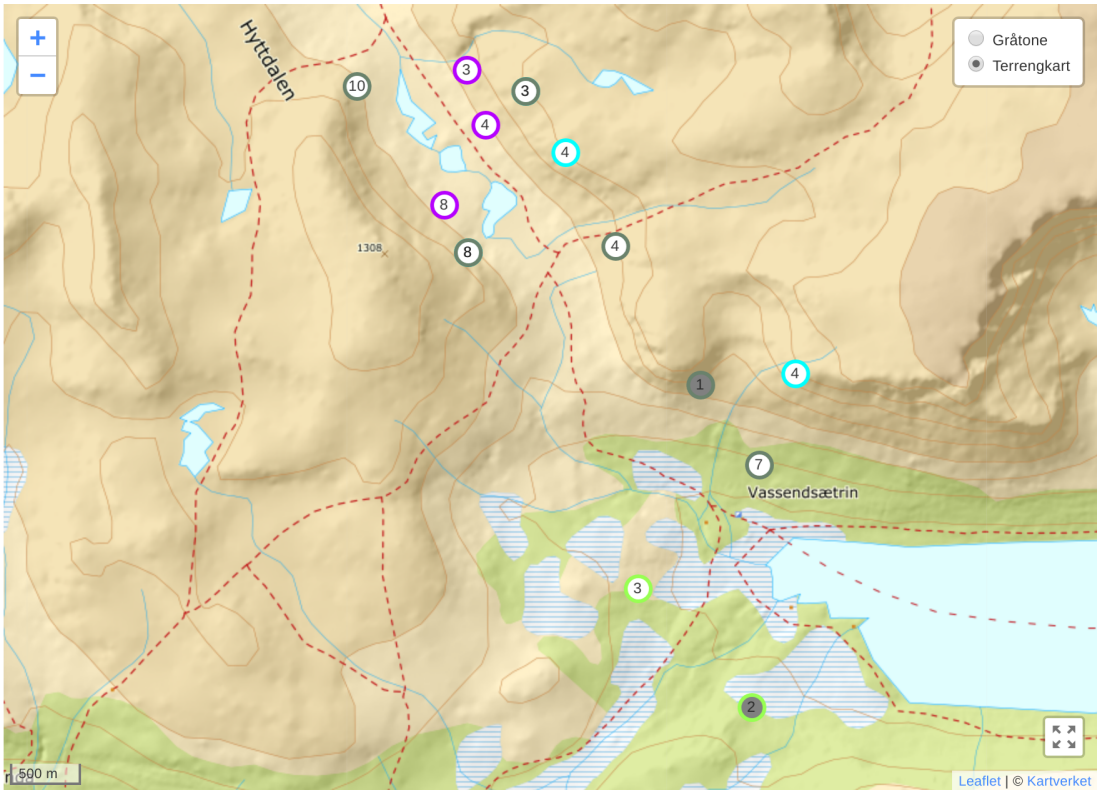

Figure 8.9.: Application Regular View

The regular view, seen in Figure 8.9, consists of all observations shown as a single sized point. Disregarding the number of animals observed, all points have the same size. Each trip has it's own colour, which is the same for the observations done on the map. All observations from the same trip have the same colour. Instead of using icons, like Findmy, we decided to use just points with different colours. As it could be difficult to actually see where the sheep's position was, there were decided to just use a regular point, in cohesion with the expert. When clicking on the point, there comes up a popup with what animal is observed and how many of it. For example if there are observed a number of lambs and a number of sheep, this will be shown separately when clicking the point. For predator points, which are filled with gray colour, one has to click the point to see which predator it is. Sheep observations are filled with white, as the wool of a sheep is mainly white.

Sizing View

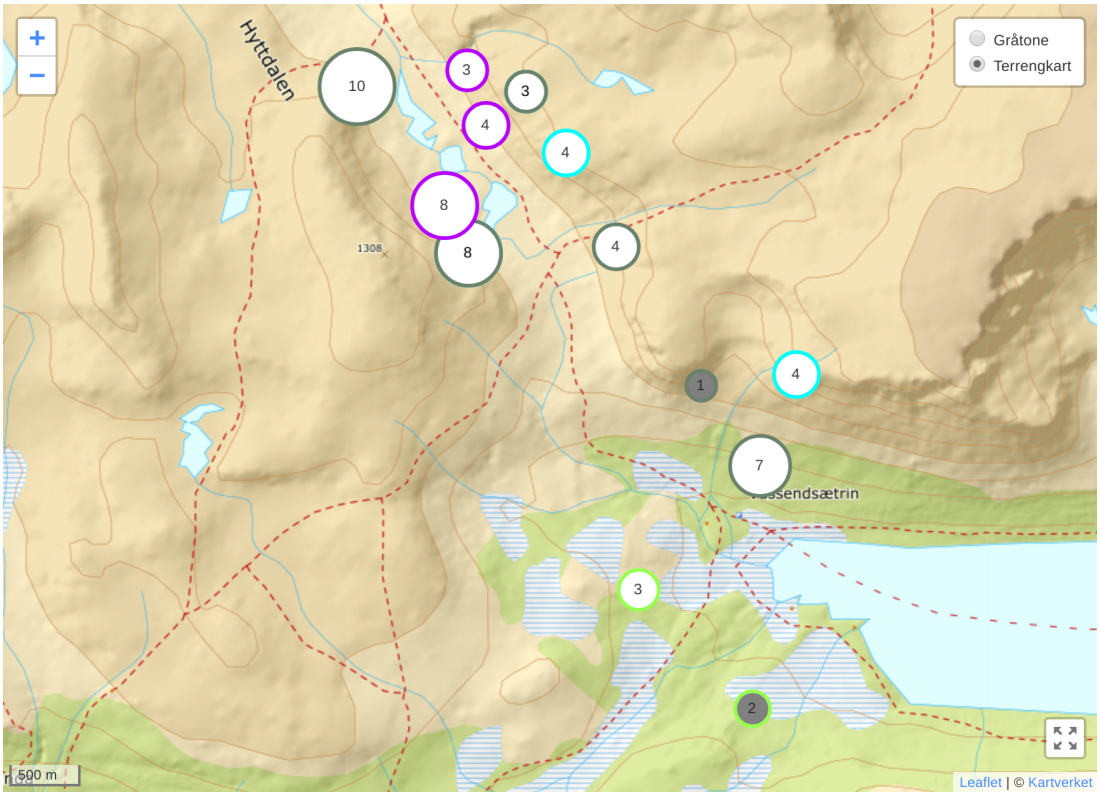

Figure 8.10.: Application Sizing View

Figure 8.10 shows the sizing view, which is quite similar to the regular view. The only difference is that each point's radius is proportional with the number of observed animals in that observation. This is done in the same way as the GIS software did it, with the main difference that each point still has it's trip colour surrounded. Regardless of this, the sizing point share the same functionality.

Heatmap View

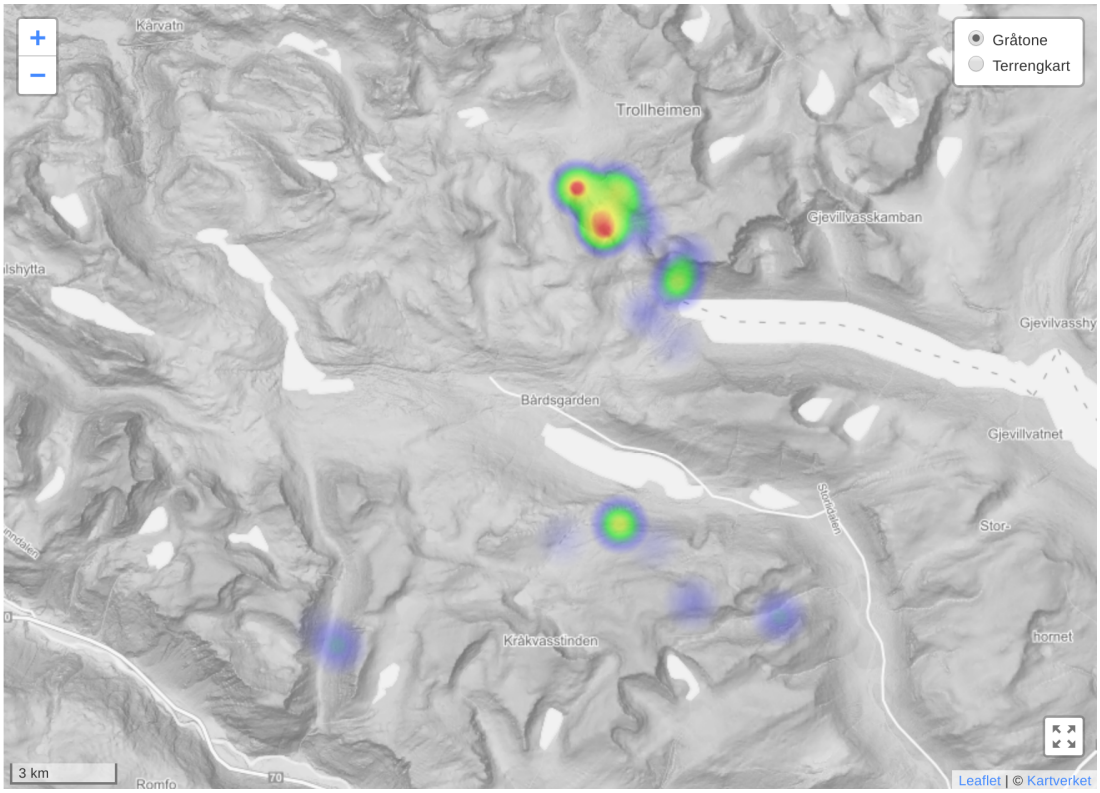

Figure 8.11.: Application Heatmap View

The heatmap view is quite different from the two other views, shown in Figure 8.11. This one will show the density of where there are observed most sheep. The closer to red the colour gets, the higher density of animals are in that area. If one zooms in, the points will spread out and one get less concentration of the colours. Blue indicates a smaller amount and green and yellow is in the middle. This is based on the highest amount of animals observed in one point and is hence a relative value. This one was used in the GIS software and is a great way to see where the density of sheep are. The points are not clickable, so no similarities with the other views on this point. In this view, one cannot differentiate between the observations from the different trips and the colours used are independent of the trip colours.

Grouping View

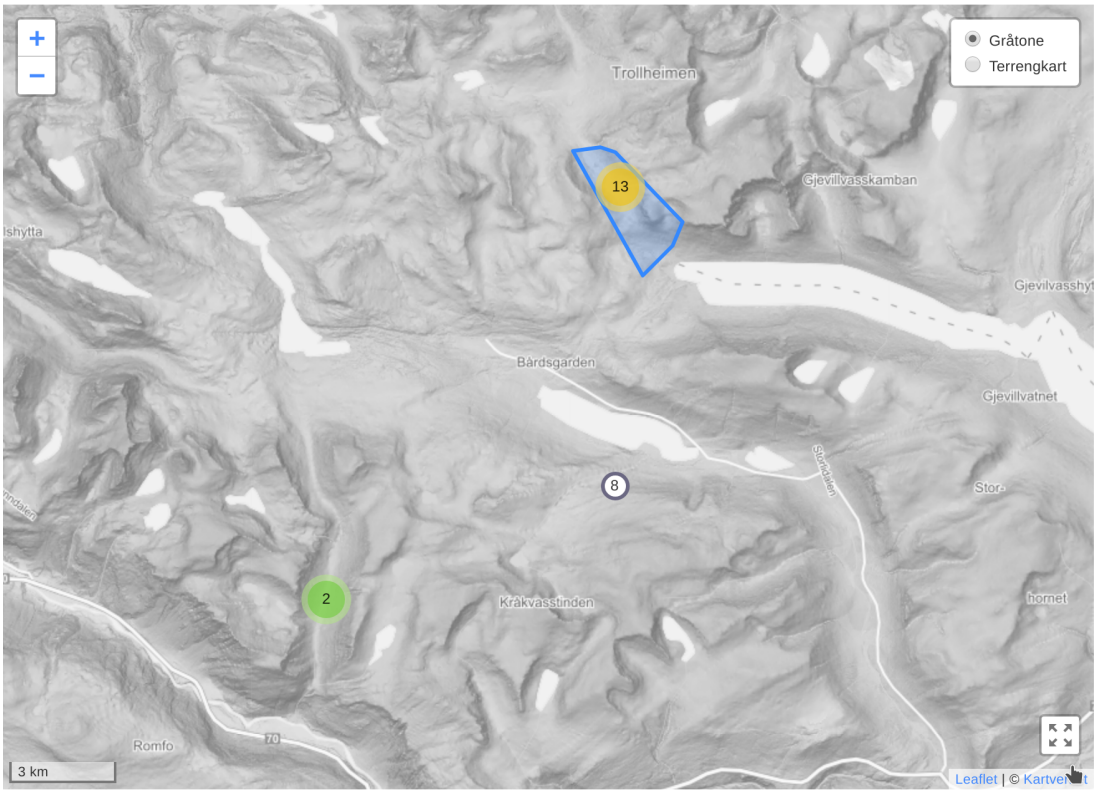

Figure 8.12.: Application Grouping View

The group view is a combination of the heatmap view and the regular view. When the zoom level is low (far out), as shown in Figure 8.12, all observations are grouped with each other, where the closest ones groups into one point. The number in the point, in this example 13, indicates that there are 13 observations which are grouped together. Similar one can see the colour of it are yellow, the higher density, the warmer colour, similar too the heatmap view. Similar, the other point, which is green and contains two observations, indicates that there are less observations done here with the usage of colours. When hovering over the points, a tiny polygon showing the bounds of it's grouped observations are showed. When clicking the point, the map will zoom into that area and reveal the points it had grouped, which can be seen in Figure 8.13. If one gets close enough, the regular point is shown, same as in the regular view. When the regular point is shown, it has the same functionality as in the regular view.

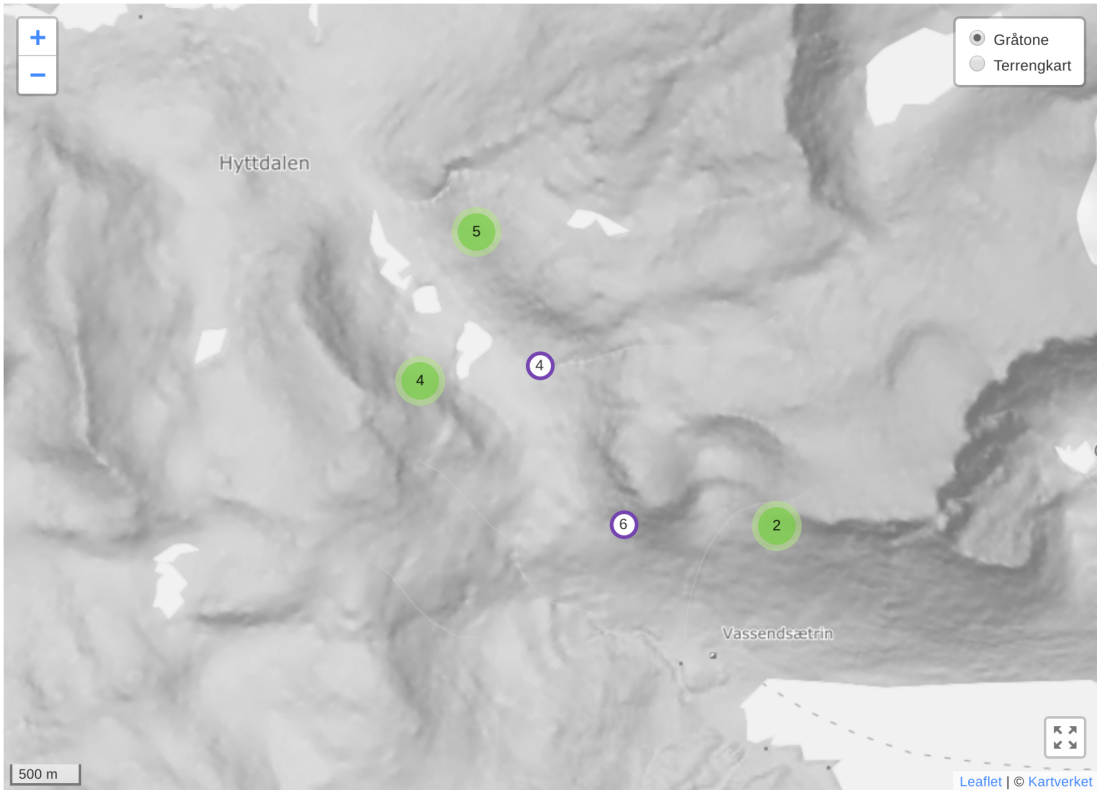

Figure 8.13.: Application Grouping View Clicked

Time View

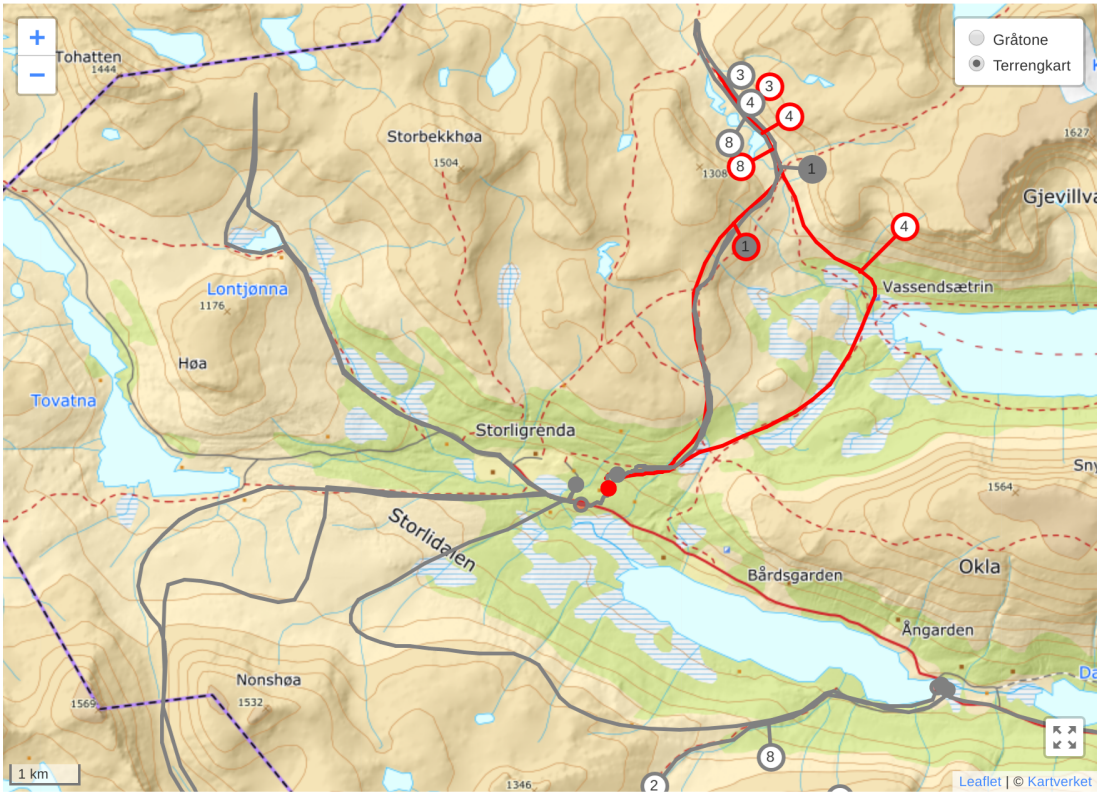

Figure 8.14.: Application Time View

Figure 8.14 shows the time view, and it has the same functionality as the regular point view. The difference are the colours used. Here there are one trip which is selected. That trip gets the colour red, both on the map and in the trip list. All other trips get the colour gray. The timeline also disappears and a dragger pops up. Clicking the arrows will select the next trip in line, which follows the chronological order of the trips based on when they are conducted. One can therefore easy see where one trip is, and where it is on the timeline.

8.4.3. Filters

How observations are depicted, are up to the main views of the application. As an addition to the main views, there are implemented filters. The different filter's job is what kind of information to show on the map. The filter list view are shown in Figure 8.15.

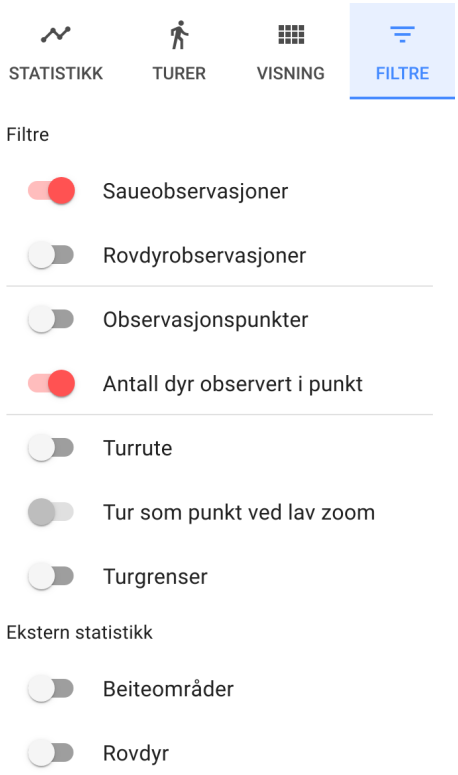

Figure 8.15.: Filter List View

Mainly each filter will toggle a specific type of information on the map, and possibly in the graph as well.

- *Sheep observations* – will show a dot on the map where there are observed sheep.
- *Predator observations* – similar to sheep observations, but will only toggle predator observations.
- *Observation point* – will toggle a line from the observed point to the place where the observation were done by the farmer.
- *Number of observed animals in point* – will toggle the number showing in each observation point, showing how many animals observed.
- *Trip route* – show the route where the farmer went while on the trip.
- *Trip point* – if the zoom is far out, the trip will hide it's observations and only show the trip route which makes the image more visual good. Similar too the grouping view with observations, only with trips instead.
- *Trip borders* – Show the area the trip was conducted in as a rectangle. Will show it's extremes.

- *Grazing areas* – depending on the area decided, show data from NIBIO based on actual pasture data.
- *Predator data* – predator observations fetched from NINA.

The two last filters, grazing areas and predator data are discussed in Section 8.4.5. In Figure 8.16 are all filters turned on (except of grazing areas and external predator data). Mainly, the filters are used to make the application more usable for expert users, which follows Shneiderman's second rule. In combination with the predefined cases, which will be discussed later on, one have a part of the systems which is to be used by novices, and the filters (and maybe the main views), will operate as an expert part of the system. As we also want the user to be in control of the application (Shneiderman's seventh rule), the filters work as a section of the application where the user can make the map more or less complex. In other terms, the users are in control.

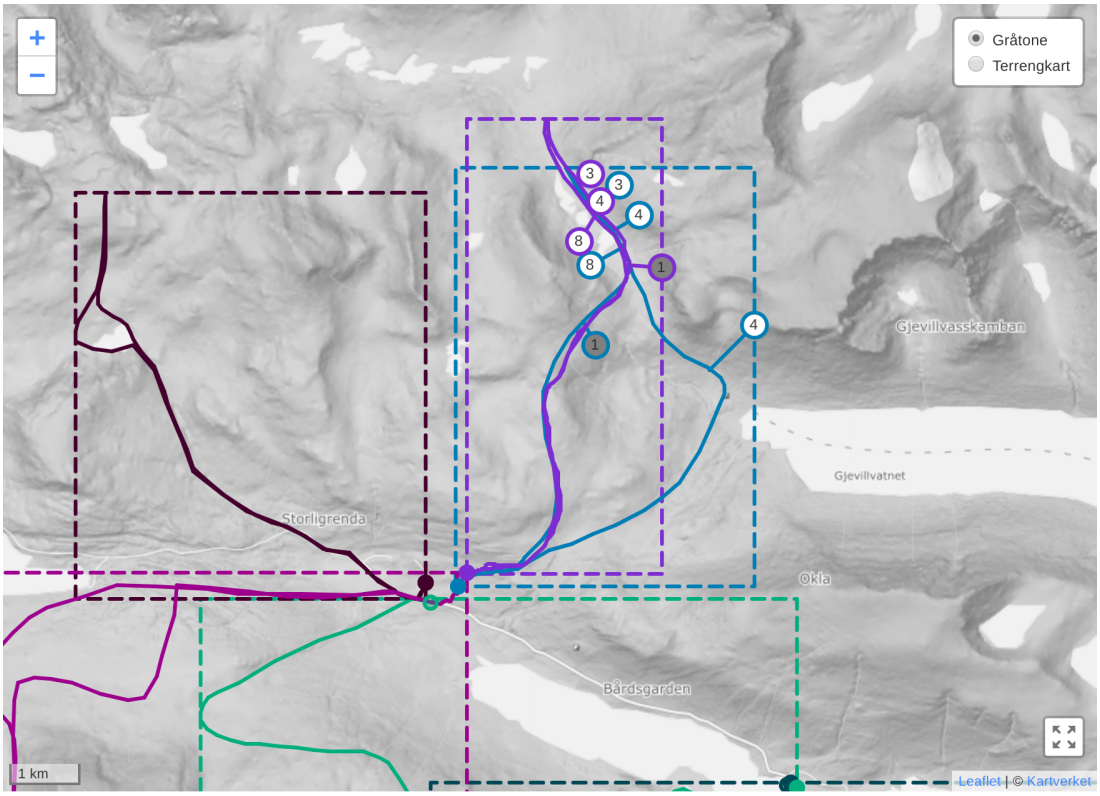

Figure 8.16.: Application All Filters On

The dotted lines around each trip are the extremes showing their bounds. In Figure 8.17 one sees how the trip point filter work when the map is zoomed out. The trips' observations aren't showing, as well as their routes are made clickable. When clicking on one of these trips, the map will zoom into that trip, and the observations will show again, similar to the grouping view. This will make the view less complex and messy.

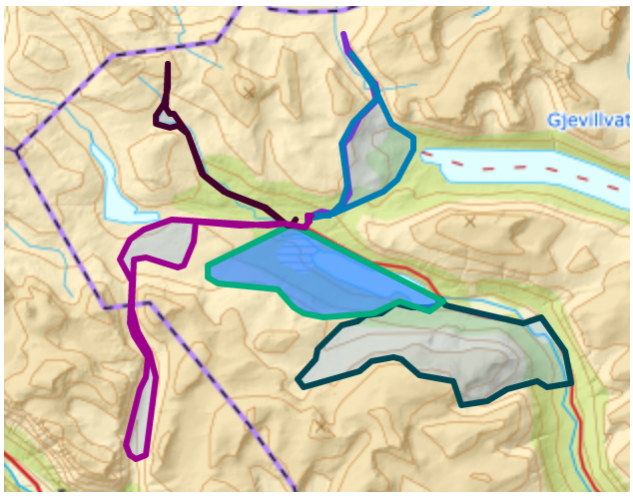

Figure 8.17.: Application Grouping of Trips

8.4.4. Edit Colours of the Trips

To keep the users in the control of the application (Shneiderman's seventh rule), there have been implemented a view in the application where one can edit the colours of the trips manually. During the development, there was seen that some of the trips could get colours which could easily be indistinguishable from each other. In terms of users with colour blindness for example, this option was a wanted feature to include as well, such that one easy could change the colours. The colour picker can be seen in Figure 8.18.

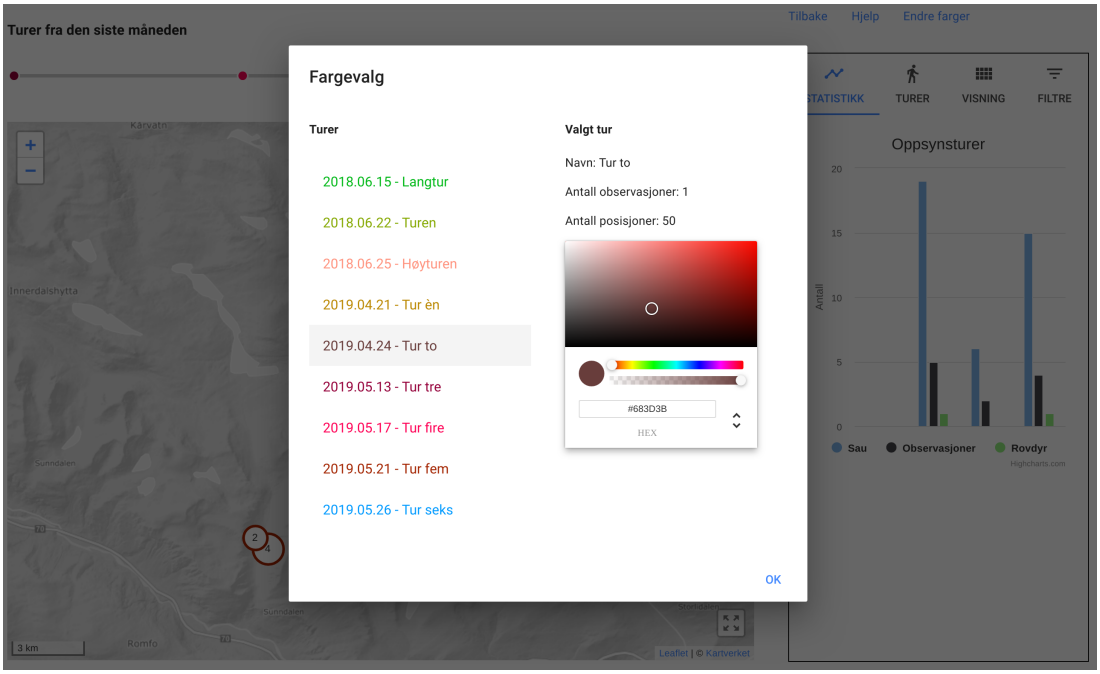

Figure 8.18.: Colour picker for trips

In an earlier stage of the application, the colour picker view was set as an own view which one could navigate to and from. This was changed as it didn't follow Shneiderman's eight rule, as one could forget what to do when not in the same view as the map view. It was therefore moved to be an overlaying section in the same page such that it was easier to remember what one actually was to do, as well as one can see the map behind of it.

8.4.5. External Data

The application have implemented two different sources of external data beyond the usage of mock data. This includes grazing area information from NIBIO (Section 4.2.1) and predator data from NINA (Section 4.2.3). As the size of the frontend application should be as small as possible, these data are saved in the database instead of in the frontend. The strength of the backend could therefore be used instead. When the data is loaded from the API, if done manually using the filters (described later on), the toggle buttons are disabled when the data is loading, such that the program doesn't allow multiple loadings and errors to happen, according to Shneiderman's fifth rule.

NIBIO

Nibio have much data in their databases, and most of them can be downloaded freely from their website [29]. In our case, the data that was downloaded and converted into using the Decimal Degree coordinate system (Theory Section 2.6.1) as the Leaflet package uses this to project data on the map.
As the size of the data is around 18MB after conversion, the data is a bit too large to be saved into the web application. The data is therefore saved in the database which the API has control over. When a case is clicked, the user can decide whether to show this data or not. If the user want to show the data, the application will fetch it, based on the bounds set by the map view shown. As each case is bounded to a given location, the application will only load the data that is necessary for that area.
The fields are then showed as polygons on the application map and can be seen in Figure 8.19. Each polygon is clickable and a popup with the following fields will be shown on the map:

1. Number of sheep in the area.
2. Number of lambs in the area.
3. Number of sheep lost in the area.
4. Number of lambs lost in the area.
5. The year which the data were tracked.
6. Density of sheep per square kilometer.

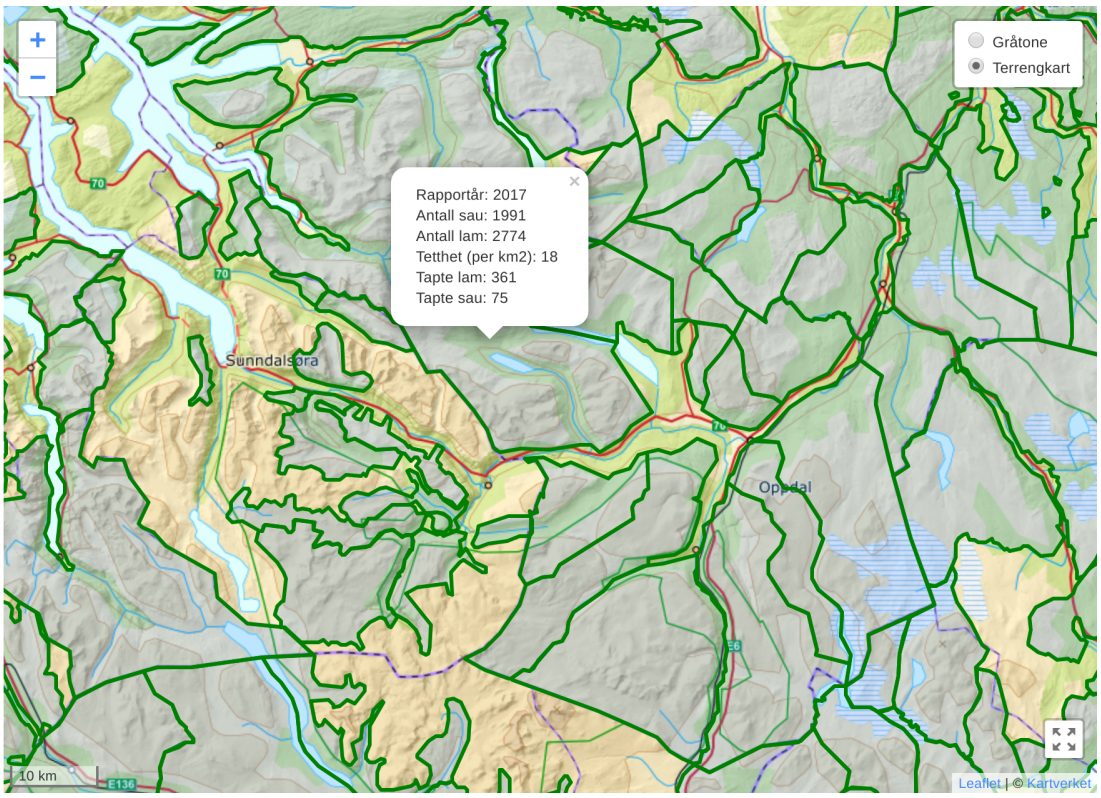

Figure 8.19.: Grazing Areas

NINA

As NINA collects different data and does analysis regarding animals and species in the Norwegian flora, they have much data that can be used in our application. The predator data were downloaded using their exporting tool in Excel format. There were therefore created a tiny Python script which read the relevant columns in the file, and wrote the data to the MySQL database. There were decided to only use data from five years back, from 2013 and forwards. This reduced the size of the dataset, causing it to show more relevant data. As there isn't necessary relevant that there were observed a wolverine in an area for fifty years ago, this data were neglected. The data file contained approximately 33 thousand entries with observations of predators, such as cadaver examinations, trace findings and real observations. As described in Section 3.4, there are five predators which hunt and kill sheep in Norway, which are wolves, wolverines, bears, lynx and golden eagles. In the data downloaded from NINA, all these five are represented. The data were downloaded from Artskart (Artskart.artsdatabanken.no) on 30.04.2019. The data originates from the following datasets: ARC, GBIF-noder utenfor Norge, Norsk Ornitologisk Forening, Miljølære.no, NTNU-Vitenskapsmuseet, Norsk institutt for naturforskning, JBJordal and Miljødirektoratet. The columns which are relevant from the exported data, are the following:

- Dataset – what dataset the data have it's origin from.

- Find date – the date the observation were done.
- Name of the animal.
- Sex of the animal, if collected.
- Location – the location of the observation in the Decimal Degree format.

As the size of the data is approximately 6.5 MB, there were decided to only save this in the backend database, so that the frontend could query only relevant areas and get predator data from it. The size of the frontend application is therefore kept as small as possible. The frontend sends it's current bounds to the API, and retrieves the observations done in that bounds. This can be seen in Figure 8.20.

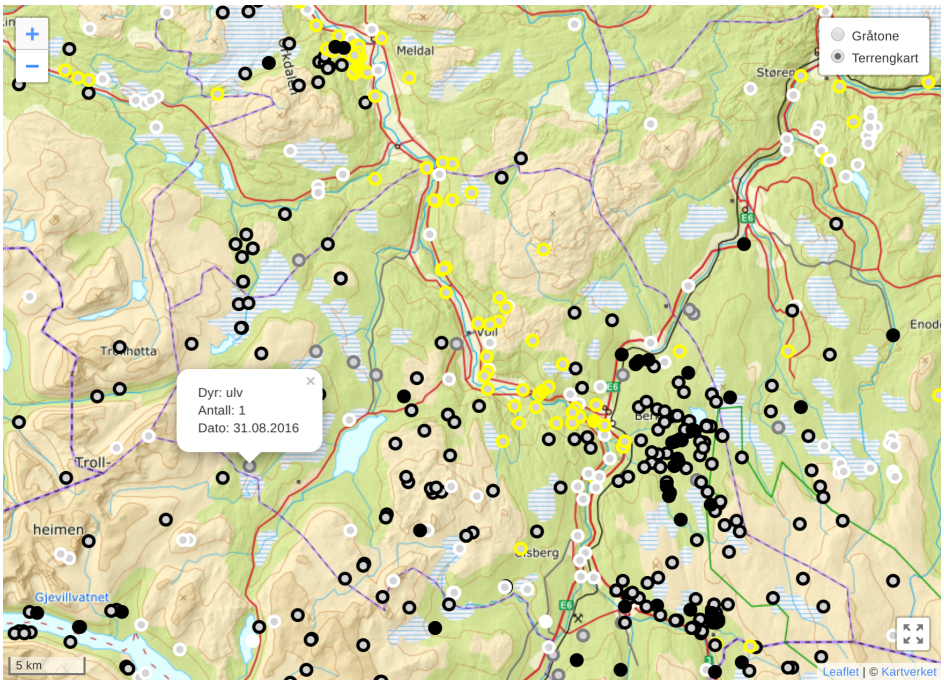

Figure 8.20.: Predator Data from NINA

Each observation which is done during the last year, in our case 2019, are filled dots on the map. Observations done before this, are filled with gray colour. Each dot is clickable, which is seen in the figure. The information that pops up, are what kind of animal it is, how many which are observed and the date for the observation. Each predator have it's own colour which partly represents their actual colour so it should be easier to distinguish between them.

- Bear – brown
- Wolverine – black
- Lynx – yellow

- Wolf – gray
- Golden eagle – white

8.4.6. Help Page

As the map view has a lot of options and much functionality, there was decided to implement a legend and a help section for the prototype. The points on the map can get a bit complex, and maybe not as intuitive wanted. A brief description of how the main views for observations are showing are also included, as well as where the external statistics comes from.

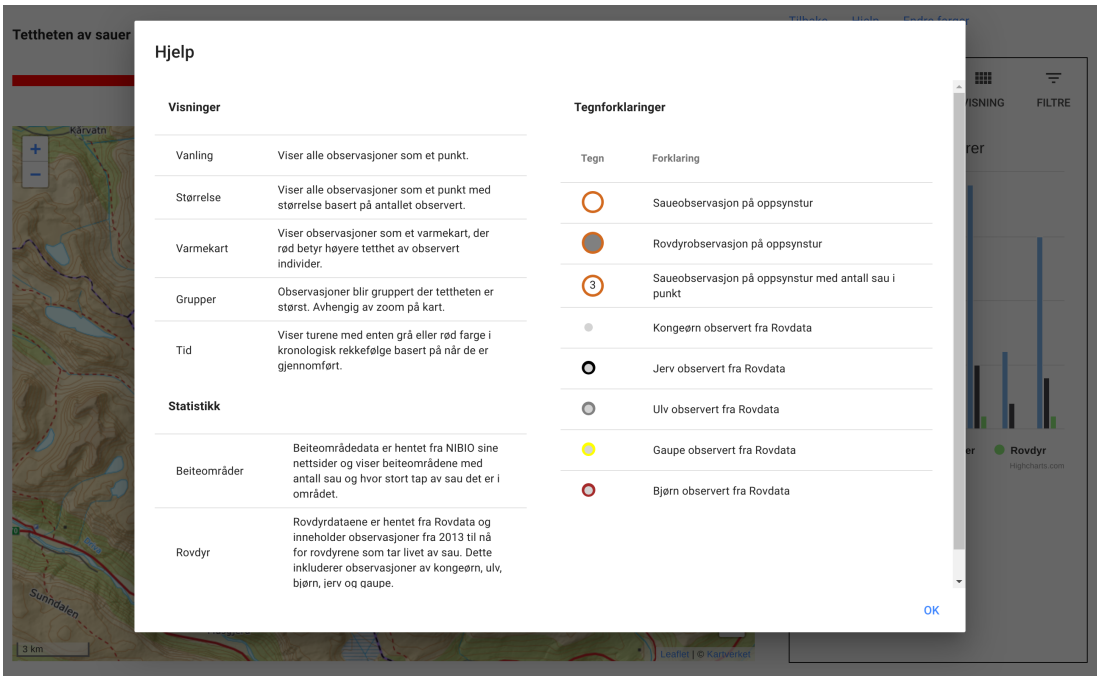

Figure 8.21.: Help view with legend

To the right in Figure 8.21 there is shown a legend for the signs used on the map. This includes descriptions of what a sheep and predator observation is. As well as showing all points which is used in the predator data from NINA. To the left there is a simple description of what each of the main views is showing. Down left, the external statistics are shown and where they are fetched from. Both the grazing area and the predator data. This page were created so that it should be easy to understand what the meaning of the symbols are, if they weren't as intuitive as intended. This is also the case for the main views. They can easy be tested out, but maybe some of them doesn't make as much sense as intended.

8.4.7. Feedback From the System

When errors happen in the application, or some different aspects cannot give the users what they want, the system should provide useful feedback (Shneiderman's third law). This is the case for all of the interactive elements in the application. For example if one toggles the filters or edits the main views, new elements should be shown on the map instantly when interacting with the buttons. Some relevant action should be shown. Another example which is for error handling, is if one clicks on a predefined case whereas there are no trips to be find, for example in trips for last week.

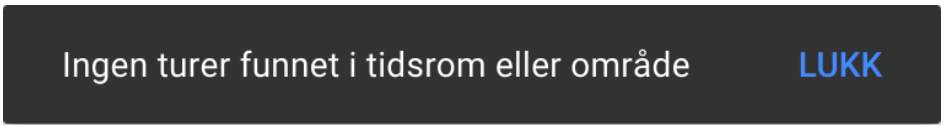

Figure 8.22.: No trips found in predefined case

Figure 8.22 shows the feedback the user gets when accessing either a predefined case with no trips included, or searches for an area with no recorded trips. It says "No trips were found in that period of time or in the area". This feedback box pops up down left in the application, and can also pop up if there are any error regarding loading of API data. Another example can be if the user clicks the trip in the trip list, and the map zooms to that given trip. The popup will either hide itself after five seconds or by pressing the "Lukk (close)" button.

8.5. Predefined Cases

As there are multiple filters which the user can choose from when obtaining the data visualisations, it can become a bit complex. To remedy the complexity of the filters and options, we wanted to drill down the ease of use of the application. In other terms, create some predefined cases which will set the optimal filters for the given situation. The predefined cases will have different main views and filters toggled for each of them, depending on how many trips which are expected to be visualised on the map. For example if one wants to see the last trip conducted, the map view will only show one trip on the map, wether if ten trips are shown on the map, other options should be used to get a less complicated map. Therefore the usage of predefined cases were developed, which toggle different filters and main views. After one have entered a predefined case, and entered the map view, some set of filters are turned on and some off. These filters can be toggled later if the user want to hide or show more information, in the same map view for the selected predefined case.
As a farmer has been in the pasture and conducted a trip, the application was used for tracking. Each predefined case have a different combination of the filters described above, which will give the farmer a different view. Each predefined case will filter away trips which aren't relevant for the given query. The main idea is that each analysis view have a list of all the relevant trips of the given predefined case. If a predefined case

cannot find any trips within it's bounds and restrictions, the user will get feedback that none trips were found. If the case is valid, the map will zoom into the bounds of all the trips that were chosen based on the case.
After accessing the analysis module of the application, one gets access to the predefined cases, which is the view in Figure 8.23. This is the first view the user will see. The map view will reveal itself after one have chosen a predefined case.

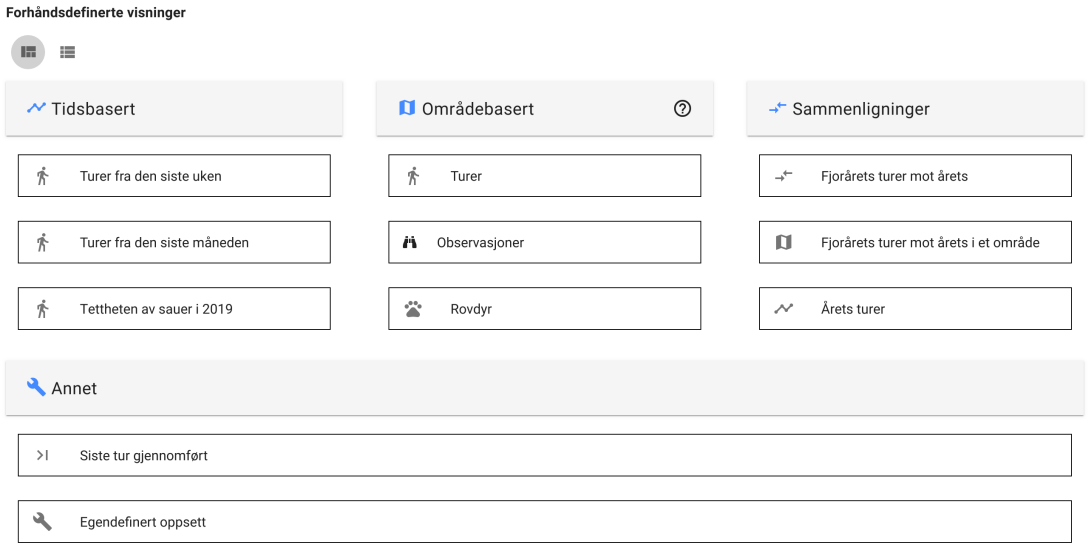

Figure 8.23.: Predefined Cases of the Application Grouped by Type

Predefined cases are grouped by four different metrics:

- Time based – upper left
- Area based – upper middle
- Comparison based – upper right
- Others – down

To make the view of the cases less complex and more understandable, they were grouped this way. This will possibly make the overview less complicated, and the user can more easily find which predefined case suits a given goal to achieve. Early in the development process, these cases were shown as a list, which is seen in Figure 8.24. It seemed like the predefined cases became more hidden using the large list, instead of grouping them, therefore the grouping was set as the standard view.

Figure 8.24.: List of the Scenarios of the Application

The icons used are an indication on what kind of filters or data that is showed to the user. As not all of them could make sense to the user, this could have been solved by implementing a legend for the signs. In the same manner as the grouping of the cases, the icon usage were added so each case was more easily distinguishable. As one doesn't only need to read the text to understand what the case does, but get an indication of the functionality only by looking at it's icon.
As the application is to prevent errors, Shneiderman's fifth rule, predefined cases which don't contain any trips, will be prevented to access. The reason why the case aren't hidden in the first place, is for consistency, Shneiderman's first law. If removing it, the user may have wondered why there were missing some cases. The case could have been disabled, but for better performance, there was decided not to do this calculation before the user clicks the case. For example if no trips are conducted in the last week, one shouldn't be able to access this case, as no trips have been conducted.
When the users are in the map view, the view's header is the same as the case selected. This is so that the users easily can remember what case they actually clicked on (Shneiderman's eight law). The same can be said for the map view, where there were chosen that all information were to be shown in the same view. This is the reason why there have been implemented a side view, were all actions can be done, such as filters, choosing of trips, and main view choosing.
Note, in the next sections, which will discuss each of the groupings, they will mention which main view is used. This main view is referred to how the observations are visualised (Section 8.4.2). The five listed were regular, sizing, heatmap, grouping and time main view. The predefined cases were made for more easily to achieve the requirements

described in Section 3.5.

8.5.1. Time Based Cases

This section includes three predefined cases, which is mainly based on time. All cases in this group depends on when the trips were conducted. The icon used is symbolising plots on a graph, which is done over time. All cases focuses on trips which have been conducted this given year, and since they are focusing on the trips in the time grouping, each case use a walking person as the icon. All trips shown will show it's trip colour when visualised on the map.

Trips from Last Week

Will show all trips which were conducted the last week – the last seven days. Will have predator and sheep observations turned on with the regular main view enabled for observations. The trip route is turned on as well. As not too many trips are to be shown in this view, more filters can be turned on so that more information about this trip can be shown. This case can be seen in Appendix B in Figure B.1. Uses the regular timeline.

Trips from Last Month

Quite similar as the "last week" case, but will not have the trip routes turned on, as more trips are expected to be shown. As well, the sizing main view is chosen for the observations. As the routes can become overlapping with many trips showing, there was decided to have this turned off, so the map got less complicated. All trips from the last month, or from the last 31 days will be included in this view. This case can be seen in Appendix B in Figure B.2. Uses the regular timeline as the span of conducted trips is expected to be not too large.

Density of Sheep from this Year

Will contain all trips that have been conducted this year. As this view probably will contain many trips, the heatmap main view is turned on for this one. It is easily seen where there have been observed sheep as the heatmap view shows "hot areas" where there are lots of observed sheep. The trip route is turned off and predator view is turned off. This case can be seen in Appendix B in Figure B.3. As there can be a large span on when trips are conducted, the compressed timeline is used for this view.

8.5.2. Area Based Cases

The area based cases have a another focus than the time based ones. Instead of focusing when each trip have been conducted, one focuses on where they have been done. The icon used is representing a map, as should be an indication of area based focus. Beyond this, similar functionality is used as the time view. Figure 8.25 shows the searching area view which one will access after pressing one of the three predefined cases in this group.

Figure 8.25.: Area Searching View

Before one can enter the map view, one have to find the area which one is to find trips for. Zooming into an area and going further, will search for all trips which have been conducted in that area and select those for the map view. Here, one can set dates as well, such that one can find trips for a given time period. This is optional. The toggle button in the figure will, when turned on, include all trips which have been into that area. If off, it will only include trips which solely have been in that area. If no trips are found in that area, and in the given time period, the system will give feedback to the user that no trips were found. All trips shown will show it's trip colour when visualised on the map. For the predefined cases "Trips" and "Observations", the regular timeline is showed. For the "predator" case, no timeline are showed as it isn't relevant for this case.

Trips in a Given Area

Will find and locate all trips which have been conducted in a given area. The icon used for this case indicates it's focus on trips, as in the ones in the time grouping. The regular view is turned on, as well as the trip route and the sheep observations. This case can be seen in Appendix B in Figure B.4.

Observations in a Given Area

Quite similar as "Trips in a Given Area", with the only difference that the trip route is turned off and the sizing view is turned on instead of the regular view. This focuses more on the observations made. Predator observations are not turned on for this view and only trips with sheep observations will be included. The visualisation of the icon is showing a set of binoculars, which is to indicate something one have seen. This case can be seen in Appendix B in Figure B.5.

Predators in a Given Area

If the farmer is about to set out sheep on pasture, for example in an area the sheep have been onto multiple times, the farmer may be interested to know which area to put out the sheep. There could therefore be interesting to see areas where there have been observed predators. As described in Section 3.4, there are many sheep that get killed by predators during the grazing season. The farmer could therefore have interest to see where there have been seen and observed predators, which could be registered directly into the system, or by data from NINA. The application will show predator data from NINA and the ones observed from the trips conducted. This case can be seen in Appendix B in Figure B.6. The trip route is turned off, and the search for trips will only include ones with predator observations. Sheep observations are turned off. The icon used is a paw.

8.5.3. Comparison Based Cases

The focus for this section of predefined cases, are that one can compare data with what have been done the previous year. Looking at areas of interest, and data which have been harvested from last year compared too this one. Other usage of the timeline and colours are used in this section. The icon used are two arrows pointing at each other, indicating comparison.

Last Years Trips Compared to this Year

Will search for all trips conducted this and the last year. Trips from this year will have the colour green, and trips from last year will have the colour blue. This applies to the map, timeline and the trip list. One can therefor easy distinguish between the trips conducted for the different years. The regular main view is turned on, as well as sheep observations. Uses the compressed timeline, which will show the trips as usual. The graph is a bit different from the others, as it will show two groups of information. The total data will show sheep, predators, number of observations, and number of sheep observed per trip, for each of the two years. This case can be seen in Appendix B in Figure B.8. Uses the same icon as the comparison grouping.

Last Years Trips Compared to this Year in a Given Area

This one is quite similar too the last one, as the only difference is that one is to decide what area to search in first, with the same procedure as the area based groupings. One can therefore get more representative data from a given area, where there have been conducted similar trips from the two last years. The colour mechanic is the same as the last one, as well as filters and main views. This case can be seen in Appendix B in Figure B.7. As this is an area based case, as well as a comparison case, the map icon is used.

Trips from this Year

This will show all trips from this year, with both sheep and predator observations turned on. The trip route is turned on as well. The main view which is turned on is the time view. The user can therefore easy see where the trips have went in chronological order based on when they were conducted. The trip selected will have the colour red, and the other trips will have the colour gray, which means the user can easy see which trip is selected. This case can be seen in Appendix B in Figure B.9. As this is some way similar to the time views, the icon used is the plotting icon. No timeline is showed here as the stepper for the time main view is showing. The grazing area data from NIBIO is used in this case.

8.5.4. Other Cases

Last Trip Conducted

The last trip conducted will be showed when clicking this. The trip list will only contain this trip and it cannot be toggled. The trip will be showed on the map, and one cannot zoom out of the bounds of the trip. The trip route is showed along with sheep and predator observations, and their observed location. As this case only contains one trip, no grouping is necessary. This case can be seen in Appendix B in Figure B.10. As only one trip is to be showed here, no timeline is shown in this case.

Custom Setup

As not all of the predefined cases can support all wanted use cases, there is created a custom setup case. One can toggle all trips if one wants, and show everything on the map. All trips will be shown initially, with the regular view, as well as sheep observations turned on. All trips are filtered on, as well as the regular timeline is showing. As trips from the last year and this year are showing, there will be a lot of empty space between those two years. This case can be seen in Appendix B in Figure B.11.

8.5.5. Yield of Closure

The application contains mainly of two parts, the choosing of a predefined case to start from, and ending in the map view. This was mainly due to Shneiderman's fourth rule, which says one should have a distinctive end for the goals to achieve. Our ending is therefore the map view where one yields the wanted information. For the predefined cases which are area based, one need to go through another step as well, choosing what area one wants to search in. Then one would achieve the wanted information when achieving the map view.

8.5.6. Reversal of Actions

According to the sixth law of Shneiderman, an action should be easy to revert. When one is in analysis module, either the map view or the predefined cases view, the data state of the application is stable. In other terms, one cannot edit or temper the actual data in this mode, only see different visualisations of it. The application is made easy to navigate between the predefined cases and the map view. In the map view, every action done can be easy reverted, either by toggling filters on and off, switch main views, and filter different trips. If one for example wants to show all trips on the map, there is created a "toggle all trips on" switch, which makes this easy. Every action which tempers the map information can therefore be easily reverted. The same can be said for the map, if one for example zooms to a location which is far away from the location of the data, one can easily press the zoom button in the down right corner, which will zoom to the area containing the data. One can as well click on a trip to zoom into it, which prevents the user of getting lost within the map, and reverse the zooming action done. The auto zooming on the map does follow the seventh rule of Shneiderman as well, as it keeps the user in control of the application. If not, they could have been annoyed when trying to find back to the desired area.

8.5.7. Cases Not Implemented

Not all cases which was written down during the alternative design and requirement phase, where implemented, for various reasons. These cases will be listed here.

Show Similar Trips in a Given Area

Trips which have the same "parameters" could have been interesting to show for the farmer. This was seen in Figure 6.5 in the Early Sketches Chapter. For example where different trip routes are quite similar in a given area. This can as well include trips which have similar observations, mainly sheep observations.

Show Potential Areas of Interest

This case can be quite ambiguous. It can show areas where there have been observed predators and cadavers of all types of animals – especially cadavers of lambs and sheep, and show these areas which will have points of interest. For example one can show where one should not put out sheep, or areas where it can be perfect for putout of sheep. This can also be combined with usage of geographical and rural data which isn't implemented in this iteration.

Movement Pattern for Sheep

One interesting aspect is to track and see the movement pattern for the sheep. One way it can be done is the same way that Findmy and Telespor have done it. As one can see where they have walked at different points of time. As they uses digital bells to

track their position, it is more easy for them to see where the actual position is. As the data tracked in our application, aren't "live data", this is more difficult. A observation needs to be mapped to another observation from another trip for this to work. Another different aspect, is that the observations done in the application, are flocked based. This means one has to see the movement pattern in addition to flocks and not each individual. This link should be generated automatically, where the farmer can edit the links if they contains errors. Based on the early sketch in Figure 6.6.

9. User Tests

9.1. Method

There will be conducted six tests with six participants. No personal information will be recorded during the tests. The questionnaire will be answered using our computer, and all notes taken during the tests, will be taken with pen and paper. No personal information are treated electronically. This is therefore not reported to the *Norwegian Center for Research Data*, according to the guidelines set by NTNU for following the *General Data Protection Regulation* [51]. Each test will be done using the same data for all of them. As the tests will be conducted on different days, the dates of the data will be shifted so that the last trip conducted in the system, is done in the same day as the test. All cases will therefore behave in the same way for each test participant. The tests will be split into three different phases:

1. Testing of the prototype where one is to use the application and solve tasks.
2. Fill out a questionnaire with relevant question based on the SUS (Section 2.4.4), and questions about the whole of the application, in regards to features, usability and design.
3. Interview where we discuss with the participants different questions according to the application

The tests to be accomplished, are following the principles of a standard usability test.

9.2. Test Plan

When conducting a usability test, which is described in Section 2.4, before starting with the testing, one needs to create a test plan.

9.2.1. Scope

As the system created for this thesis is based on the specialisation project, there are only one part of the prototype which is to be tested. As of this, all navigation out of the analysis module will be turned off during the tests, such that users cannot navigate outside the given module of the application. The analysis module is what to be tested, which includes the predefined cases overview, area choosing view, and the map view.

9.2.2. Purpose

According to the last phase of interaction design, this will be the first part of the evaluating phase. The main goal for the user tests, is to find which of the functionality implemented, worked as intended and not. The other aspect is to map missing functionality, which will be based on the participants suggestions and thoughts. Are there features which is interesting for a sheep farmer to use. The same will be looked for in regards of the prototype's design.
The data collected during the tests will be the basis for the results of this thesis and the conclusion. It will suggest changes to the application, new features, design choices and what worked as intended. What should be included in the next iteration of the system. As this is the first iteration of such a system, there is expected that the system will not work a hundred percent as intended, and that there will be missing functionality.
As described in the Theory chapter, in Section 2.4, the goal for using this kind of test, is to find functionality of the application, in addition to finding out how well the user experience is.

9.2.3. Schedule and Location

The tests will be conducted ad hoc, when the participants have time available. All tests are to be conducted during the same week, after the application is finished developed. When the first test have been conducted, no more implementation and fixing will be done to the prototype. The only thing which is different from each test, is the time of the data in the system is set. All tests will have a trip conducted that day, two the last week, and four for the last month. The data starting point for each participant will therefore be the same. A deeper description of the mock data is described in Section 5.4.

9.2.4. Sessions

There will be conducted six different testing sessions with one participant in each of them. Each testing session will last for approximately 45 to 60 minutes, depending on how fast the participants are solving the tasks, and how much input they have. The sessions will contain three different parts. The first part is testing the application and solving tasks. Each participant will be offered to test the application before solving the tasks, so they can get familiar with the system. As the main goal is to see whether features are interesting, we want the participants to get familiar with the system before they start to solve the tasks. They can get the feeling of how the system is to work, and what to expect when using the predefined cases. This part is estimated to last for approximately twenty to thirty minutes. The second part will contain a questionnaire with different questions regarding usability, features and design. Time estimated to be used on the questionnaire are approximately five to ten minutes. The third part will be an interview, as the retrospective probing (described in Usability Testing). Will mainly focus on the functionality and design choices for the questions asked, and to get their meaning of different aspects of these two. If the participant did something in the

prototype testing that were interesting or weird, this will also be asked during this phase. To get most feedback from the sessions, we will allow the subjects to show suggested changes directly in the prototype, for both design choices and features. This section is estimated to last approximately fifteen to twenty minutes.

9.2.5. Equipment

The system will be run locally on one of our computers in a local environment. The computer have support for navigation and interacting within the system, using either the touch pad or the touch screen. The computer will be used for testing the application and for fulfilling the survey. The observers will take notes using pen and paper. As mentioned in the introduction for this chapter, no personal data will be treated electronically during the tests.

9.2.6. Participants

There are six participants which will test the application, whereas the last participant will be the expert.
As the sheep farmers are in a busy period, with having lambs and putting the sheep on the grazing field, it was difficult to get hold of a sheep farmer to perform a demonstration of the prototype. In a later iteration of the system, it is recommended to perform such a demonstration for a sheep farmer. Instead, each testing participant will be "regular people", and one of the participants will be the expert, such that some of the inputs can have a "sheep farmer's head". As the expert have been involved in the development process and decisions taken, this test can be a bit biased. Independent of this, the expert have never used the application for himself, only got demonstrations of the functionality and design of the prototype. Therefore, in cooperation with the expert, there was decided to run the test for him as well, as he has some knowledge of what a sheep farmer wants and not. The tasks, questionnaire and interview are created and developed with no interaction with the expert, such that the tests' goals and purposes are developed independent of him. Except of the expert, none of the other participants have pre-knowledge of sheep farming.

9.3. Implementation of the Tests

9.3.1. Introduction

Before the test participants are to use the application, they get an introduction to the problem to be solved. The following text is mainly the information the participants get before the tests start.

"You are a sheep farmer which is holding sheep in Oppdal, Norway. As a sheep farmer you have used a tool tracking where you have seen sheep and predators in the grazing areas where you hold sheep. As well as registering observations in the application, the

route walked is tracked and sent to this prototype. A total of nine trips are completed. Last year, you completed three trips and six trips for this year, whereas one is conducted today. These trips are then sent to the application you are to test. The main purpose of the application is to show data from these trips in a reasonable manner, so that you can collect some reasonable information about resent trips, and earlier completed trips."

9.3.2. Tasks

The participants are recommended to test the application before solving the tasks. Test some of the core functionality, and features just to get familiar with the system. When the participants have become familiar with the prototype, they can start solving the tasks. The most important goal aren't for us that they actually solve the task perfectly, as some of them maybe even don't make too much sense. The most important is to see how they use the tool, if they use the predefined filters in a high degree, or if they used the filters instead, to solve the tasks. The participants are asked to "think out load" while solving the tasks – concurrent think aloud. The observers will not talk or give hints during this phase, unless necessary.
After the participants are finished testing the functionality and aspects of the prototype, they will run through the task. The tasks are made like use cases, where one asks for an outcome one wants, instead of actually telling what to do with the application. For example we will not ask the participants to turn on the trip route as a task. This will be baked into the tasks' goals.
There are created seven different tasks which the participants are to solve. The point of departure, there isn't always one correct way to achieve the goal for the task. Some of the tasks can be solved easily using some of the predefined cases, but as there are multiple ways to achieve the goals, this aren't necessarily the way they use. For example, if some of the predefined cases are too difficult to understand, there could be easier to just set some of the filters manually. The same case can be used when filtering trips.

1. *You have just been on a supervision trip. Find out how many sheep that were observed in this trip.*

2. *The grazing season have ended and all sheep have been collected. You want to see where there have been observed most sheep and the density of these.*

3. *You know there are missing some sheep from a given area. Find out if there have been observed any predators there.*

4. *In the area around Hyttdalen, there have been observed 34 sheep during this year, but you cannot remember how many which were observed last year. Find this number.*

5. *On trip the "Tur tre", there wasn't observed any sheep. You want to walk this trip once again to see if there are some sheep there this time. Find this route.*

6. *The government wants information on the data collected last year. Show data from 2018.*

7. *You want to see the trips done this year and see the data from them stepwise.*

As the main goal for this thesis is to find functionality and design choices which are interesting for such a system, time will not be recorded. Similar for the completion of the tasks, this is not important and will not be further discussed. The important aspect is to see how the participants use the system and it's core functionality, as well as the discussion of the design. The tests will have much more focus on suggestions from the participants, both of what worked and not. To find which features were useful, the participants will get a questionnaire with eight questions from SUS, which will tell how usable these features and designings was.

9.3.3. Questionnaire

After the tasks are completed, the participants are finished using the application. The participants will respond to a questionnaire, which is divided into five different topics:

1. *Usability* – contains eight questions which is based on SUS. These are standard questions used in many usability tests, to ensure that the application have the requested usability.

2. *Functionality* – asks twelve different questions wether the features in the application was good enough or not. Asks different questions based on functionality as a whole in the application.

3. *Predefined cases* – wether how well the predefined cases worked or not. Which of the groupings did what intended etc. This step includes five questions.

4. *Design* – focuses on questions regarding the actual design of the application. Can be the use of colours, visualisations and the map. Includes eight questions.

5. *Other* – adds a text field where the subject can type in anything regarding the application.

A complete list of the questions asked, and the results of them, can be seen in Appendix C.7.

9.3.4. Interview

The interviews' main goals are to get the participants talking about the prototype. This phase is the retrospective probing where different actions made by the participants can be asked for. The following questions can be used to probe for feedback from the participants:

1. What did you like about the application?

2. What did you not like about the application?
3. How well did the predefined cases work?
4. What could have been done better with the application?
5. Do you have any suggestions for improving the design of the application?
6. Do you have any suggestions for other features for the application?

The interview can work like a conversation, where the observers can be a part of it. The questions are used if the participants don't have any special thoughts. If the participants want, they can use the prototype to illustrate problems, or other aspects of it. The observers can therefore get a better understanding of what they mean if it can be visualised directly in the application. This is not a demand, but only if it is appropriate. Other questions or things the observers are wondering about can be asked for during this phase. For example decisions made during the prototype testing.

9.3.5. Evaluating

There are different aspects which are to be evaluated. The focus on evaluating the qualitative data is the most important, as this is the first iteration of developing such a system. As this is far from a complete product, the main goal isn't to see how good the system works, in manner of usability, but to have this in mind. Key aspects of the evaluation:

- Find key features which will be suggested for further development.
- Map which features did not work as intended.
- The degree of complexness one wants and needs in such an application – how user friendly it should be and how much functionality it should contain.
- How much freedom and customisation such a system should have.
- Proposals for new design elements and interactions.

The results of these tests will be the foundation of the conclusion of this thesis. The findings fill be described in the next chapter, and the raw data and notes from the six tests conducted, can be read in Appendix C.

10. Results

The results of this thesis is based upon the user tests conducted, in regards of the questionnaires and the hand notes taken. Each user test have been noted in detail in Appendix C, whereas the most important key notes from each test are written in the last section for each test session. The results in this chapter are based on these notes and questionnaire.

10.1. System Usability Scale

There are two questions which are neglected from the original SUS (Section 2.4.4). The reason for removing those, is that that the questions "*I think that I would like to use this system frequently.*" and "*I would imagine that most people would learn to use this system very quickly.*" wasn't too relevant for this purpose. As a demonstration before a sheep farmer wasn't possible to achieve at this point, the first question didn't make sense to ask. The second one was merged with "*how easy the system was to use.*".
The total score gained for all questions was **185** with an average of **23.1** per question asked (the detail of each SUS question, can be seen in Appendix C.7.1). As there is used one to five for all questions in the questionnaire, another formula is needed to convert the score to a number between zero and a hundred, than described in Section 2.4.4. The lowest score one can achieve is eight, and highest is forty in total for the eight questions. Removing eight from both the highest and lowest possible score for each participant, the gap will be 0-32. Finding the score between 0-100 is done by finding the factor which one need to multiply the result with to achieve a number between zero and a hundred. The difference of the highest and lowest score divided by a hundred will give us the scaling factor: 32 / 100 = 3.125. If the score is above 68, the system is considered usable and is above the average. For each participant:

1. (19-8) * 3.125 = 34.4, which is below the wanted amount.
2. (35-8) * 3.125 = 84.4, which is above the wanted amount.
3. (33-8) * 3.125 = 78.1, which is above the wanted amount.
4. (27-8) * 3.125 = 59.4, which is below the wanted amount.
5. (33-8) * 3.125 = 78.1, which is above the wanted amount.
6. (38-8) * 3.125 = 93.8, which is above the wanted amount.

In other terms, there were two who struggled to use the application as their SUS scores are below the wanted limit, which is 68. Four participants are above the limit, which qualifies it as a usable product. The average SUS score is the sum of the above scores, divided by the six participants. The average score is therefore 71.4, which is above the average for a usable system. A summary of the questionnaire can be read in Appendix C.7.5.

10.2. What Worked as Intended

Graphic representation of the data on a map

Displaying geographical data is well suited to show on the map. This was seen during the user tests and from the questionnaire (Appendix C). The average score for how well the system showed where there have been observed sheep was 4.3 of 5. This indicates that there was easy to see where the sheep were observed on the map. The same can be said for the system as an overview tool for sheep and for predators, with an average of 4.3 of 5 and 4.5 of 5 respectively. The overall user experience with the map was smooth, and one should continue to visualise the data on a map. Multiple ways of zooming on the map and general usage of the map worked well.

Displaying of predator data

It became clearly from the user tests that the displaying of the predator data was an interesting aspect. Especially the data which was fetched from NINA, which contains real observations made. As well there was liked that observations of predators done in trips was present as well. Using this kind of information, one could for example exclude some areas which there have been set out sheep before.

Use of filters to hinder the complexity of the map

Most of the participants used the filters actively when the information on the map became too complex. It was mainly easy to see and use them.

The regular, heatmap and grouping view

These three got either five or six votes in the questionnaire and was liked by most of the participants. The option to visualise the observations in different ways was preferable. The heatmap view was maybe the most liked one, but should have some more metadata attached to it, link an exact number to each colour for example. The grouping view was much liked as it made the information on the map less complex when it was turned on. It should therefore always be turned on when there are regular observations showing on the map.

Simple, intuitive and easy design with icon usage

The usage of a simple design using Material Design made the application quite easy to use. Buttons and interacting elements made sense and was pretty much consistent throughout the application, which got an average score of 4.3 in the questionnaire. This made the system more attractive to use, in combination of the icon usage, which got an average score of 4.2 in the questionnaire. The same score, 4.2, were on average given to how well the interactive elements worked in the application. How the elements was updated when clicking the filter buttons, main views and auto zooming, worked well. One participant mentioned that all icons made sense, except the one used for "the last trip conducted". For example, a walking person indicates a person on a trip, and maps for the area predefined cases made sense for the participants.

The ability to choose between different maps

Especially the terrain map, which includes contour lines, makes it easier to see different aspects of the area. For example to see roads, rivers and other natural signs, directly on the map. If the background map became too difficult to distinguish from the data, one could easy change to another map. The list should possibly contain more maps than this prototype did.

Grouping of the predefined cases

Makes it more simple and easy to distinguish between the different predefined cases and what kind of functionality they represent. As some of the predefined cases were too equal or didn't live up to the expectations, some of the cases should be recreated, but the main groupings was intuitive. The groupings made an intention of the usage and meaning, for example the compare grouping is the comparing of different data.

The overall interactivity

The connection and use of colours between the timeline, the map and the trip list made sense, and was intuitive. The colours used, worked as a clear connection between those three elements, and what they represented. For example if one clicked on a trip in the trip list, the map zoomed to that trip. Similar, as many things in the system was connected, and much could be interacted with, the user felt control of the application. Many elements on the map were clickable, and having an interactive map was likeable. Easy to switch between different main views, and the map updated fast when clicking the buttons. As almost nothing were locked in the application, the freedom of choice were liked.

The option to have detailed level in the application

Some of the participants liked that one could look at the data in a detailed manner, and got high granularity of the data. This can be combined with an expert mode, such that users who want this kind of functionality can have it.

Easy to see where sheep and predators have been observed

In combination with the number in each observation point, and the different views one could use to show the observations, it is quite easy to see where there have been seen sheep. This was the same for the predators, both the statistic points and the ones observed on trip.

Interacting using a touch screen

The system wasn't mainly designed to work with a touch screen, but with the usage of large elements, Material Design, buttons and other interactive elements worked well using a touch screen. The system is developed for larger screens, but this can as well include tablets for example. Zooming and navigating on the map, worked quite well using touch mechanisms.

The use of colours for distinguishable elements

Mainly, the use of colours, especially for trips, made it easy to distinguish between the trips and what data they represented. At times, the colours could be too equal, but the idea of having a representing colour works for distinguishing between different elements. Similar for the colours used when visualising the predator observations from NINA, as well as using gray for predators and white for sheep. How the trips are connected throughout the map view works nice, for example in the timeline, on the map and in the trip list.

Filtering of trips

The way one could filter the trips in the map view worked well. Both in regards to showing on the map, and how fast it went. One could easy remove trips which wasn't too interesting, if this was the case.

The usage of real observation data

As the predator data that are showed, are real observations which have been observed by ordinary citizens, made the prototype more usable for different use cases.

10.3. What did not Work as Intended

Too complex information on the map

At times, most of the participants got to a state of the application where the information on the map became too complicated. Some of the participants used filters to hinder this, but there should be even more focus on getting a less complex view. For example always show the grouping view, such that there is a way of seeing an area with a lot of observations. This could also be considered to be done with similar trip routes, as these could get a bit messy as well. From the questionnaire the one which gave most votes for complexity on the map, was too many overlapping elements showing. This can be solved by using the grouping main view. Too much information on the map, and too many colours which had a tendency to become too equal, made it complex at times. This is a difficult aspect to balance, as one wants to distinguish between the trips, but too many different colours can make it difficult to fetch the information one wants. Especially if one takes the background map into account, as there can be many colours on this as well.

Too complicated to search for trips in an area

The functionality was used much, but was to tricky too manage. As one of the participants said, "When I am holding sheep, my sheep are only in a given area, not the whole of Norway". The system should therefore show the area where there have been conducted trips, whereas one can search for trips there. Some information should be shown on the map when searching, so it will be easier to find the trips one wants. The dates should be removed from this area, or at least have been preset, as the searching focus disappeared a bit. This can be combined with a stepwise mode (discussed later on) if this is to be implemented. It wasn't as intuitive as one should expect as well. One participant mentioned that it could be easier to draw a rectangle on the map which would indicate the bounds which one had to search into, and not just the zooming in area. Another annoying aspect noted here was that it didn't show what predefined case was selected when in the area choosing view. One should also consider to show the trip routes when searching for the trips, but this can get quite messy and complex if there are lot of trips. These trips can maybe be grouped in the same manner as the grouping main view.

The use of grazing areas

None of the testing participants actually used this in a reasonable way. The information wasn't so interesting as wanted, and the use of colours also made it a bit difficult to see in combination with the trips. Should possibly be neglected the later on, but should be tested for a sheep farmer before removing it. There should be created a better predefined case for this where it is more clear what meaning it has. It can also be a combination in

that the participants didn't know what was showing, and how to use the data showed from it in a better manner.

Predefined cases could have been more specified and unique

Some of the predefined cases was too equal in their functionality, and some of them were a bit confusing. What their main intention was, didn't always live up to the expectations. For example the area based cases, there was tricky to find the area wanted. This wasn't mainly because of the zooming, but made the overall impression less good for the area based. And what the different between the predefined cases "observations" and "trips" wasn't understandable. Should either have been merged, or had a larger distance between those two. The time based ones made more sense, but should have had the option to select dates and not just the three predefined as setup here. The comparison views need to get more difference between them as well, and maybe implement other aspects to compare. This can for example be to compare only predators, only sheep, compare trip routes walked etc. This could also fit well with the stepwise approach (discussed later on), where one can decide area if one wants.

The use of colours for trips and other elements

Mainly, there were liked that each trip represented a colour, as it was easier to distinguish between them. The lack was that they had a tendency to become too similar to each other or too similar to other colours used in the application. For example the heatmap colours could use same colours as the trips. As it was too difficult to change the trips' colours, the colour picker was too detailed, and should possibly only contain a list with available colours instead. A regular user may not be comfortable to use hex colours when choosing a colour.

The intention of the timeline is a bit unclear

The meaning of the timeline is a bit unclear, for example what data it is showing. It could be confused with a zooming tool, or a draggable element. That it only showed the trips in chronological order wasn't always clear and one should look into alternative designs for the timeline. Another way of looking at it is to combine it with the trip list view, so that one doesn't have both. One could add the dates on the timeline which makes it more understandable. For example the colours of the timeline, when it was in it's compressed state, was confused with the colours of the heatmap and the meaning of them. There should be easier to see when the trips are conducted, therefore there should be some focus on the timeline and making it more functional. Maybe it could be visualised more like in the sketch in Figure 6.4 with event information on it.

The use of custom setup

The predefined case "custom setup" wasn't used by any of the participants during the tests. All of them used some of the other predefined cases. Maybe this is because they didn't want to use something custom, or that it meant more work to achieve the wanted task. Nevertheless, it could be replaced with the trip list, and one could remove the others grouping. As it didn't lie underneath one of the three main groupings in the predefined cases overview, but lied under "Other", it became a bit hidden.

More visible auto zoom button

None of the participants did use the auto zoom button in the down right of the map. This button could have helped some of them when navigating in the map, to get to the overview of the selected trips. The icon chosen and that it was a bit hidden could maybe have solved this, as it didn't shout "click me".

No data showing at some states

When the participants were to find the trip with none observations, there were different approaches for this. Many of them entered a state where the trip was present, but the trip route wasn't turned on. As well, in the graph, there was a spot for that trip, but no data was showing as none observations were done on that trip. This wasn't always understandable. When the participants filtered away all other trips, except the one with none observations, nothing showed on the map and the map zoomed out.

10.4. Proposals for Next Iteration

Have the system be more stepwise where necessary

Multiple suggestions from the participants wanted a more stepwise system, where one could go through multiple steps for achieving different goals. For example, where one could set dates, area, and what to show on the map, before one enters the map view. A similar way as the internet order example described in fourth rule of Shneiderman's rules, where one have a process which will have a clear ending with the map view. The user should always know where they are in the process, so a timeline at the top showing where they are in the process can also be considered.

A more visible legend showing what the elements on the map are showing

Much of the information on the map wasn't always easy to understand. Mainly the observations done, were understood to be sheep and predators, but it wasn't always given. Having a legend showing at the bottom right of the map for example, could have

solved this. In the similar way which was done in the GIS, which is seen in Figure 4.18. The same case should have been used for the heatmap, where one should show an indication on "how hot" a colour is at a given time and zoom. If the zoom is far out, the hotness is larger than when the zoom is low, whereas the hotness indicates a lower amount. Same for the sizing view, there should be an indication of how many sheep there are observed in a point of it's size. The number of animals are showed in the point when using the regular view, so it isn't too necessary for the sizing views. This was also an indication from the questionnaire, as only one of the participant used the help section during their session, two did not know it existed, and three did not use it. The same can also be said for the different styling of the predator observations done on trip and the external ones from NINA.

Better use of distinguishable colours throughout the system

A common problem during the tests, was that some of the colours used, for both trips and embedded colours in the system, were too equal. They should be more distinguishable. If a trip got the colour red for example, this colour could be mistaken to represent something else in the system, as the heatmap colours for example, or the colours in the graph. Use a fixed number of colours for trips which is distinguishable from other trip colours and the colours used in the system, is recommended. For this to work, a max limit of trips should be implemented to show on the map, similar as described in Section 4. Using heatmap, there isn't necessary to show the trip list, whereas showing the route and observations can get messy when there are a lot of data showing with different colours.

The complexity of the system, both for information and options

Have a deeper focus making the map less complicated, which can be fixed if one combines some of the main views. The grouping view should always be turned on where one could skip the regular view and only use the sizing view. The time view have it's functionality, but is a bit difficult to understand, both the meaning of it and the usage. Some of the filters as well should be removed, so one only have a couple of filters to show, which will make the system less complicated.

Implement a standard mode and expert mode

The expert mode can include functionality where one can get access to the filters and main views for example. The standard mode will only include the predefined cases, and some easy filters. Mainly, the participants thought there were the right amount of functionality, but could possibly have been a bit less. This can be solved with the expert mode where one can have the same amount of functionality for expert users, and less options for the standard view, which makes the core functionality of the system easier to use and understand. In combination with removing some of the filters and the quite equal main views, the system would have been more lucid.

Have the opportunity to select which trips to show in the map view before entering it

Many of the attendants of the tests had some problems finding the information they wanted, especially when one distinct trip was to be showed. As they could have used the "custom setup" case, this wasn't intuitive enough as one had to enter the map view before filtering the trips. There should be a good indication when one have completed a task or a goal using the application, as the fourth rule of Shneiderman states. The map view is such a ending phase, and maybe the choice of trips to include, should have been moved to an earlier stage than the map view. Therefore this kind of filtering should have been done before entering the map view. There could for example have been created a list where one can filter the trips wanted before entering the map view. The problem came clear when the participants was to show the trip with none observations.

Have more background maps available, and make the system remember which map the user prefers

There were two background maps available in this prototype. This number can be expanded, as there isn't too much job to include more of these. Kartverket have support for more maps than used, which will work with the technologies used in the prototype. Some of the participants preferred the terrain map instead of the gray tone map, but the system didn't remember that the user changed this. The system should remember the choice and show the user's preferred map.

Have the ability to save custom setup of filters and dates

This should probably be used in the expert mode if it is necessary. Another view one can have for this, is that one creates better predefined cases instead, which will make this negligible. The option can be nice to have, especially if one wants this functionality. It will make the system a bit more complex, but possibly make it better. Should be tested in the next iteration of this system. The case is that one can set all filters and main views which one wants to show, maybe also a date span (which can serve as a date template over different years or months), which can be saved and retrieved later on.

The ability to undo actions made

This holds mainly for the filters, main views and what trips to show. As it isn't always as easy to remember what one did in the last action. One can encounter states of the map where it gets messy, and there isn't necessarily easy to remember how one can get back to a satisfied view. The user doesn't possibly remember what filters and main view was turned on. An undo button or something can solve this, where the user can back-trace what was turned on and off. Implementing this will follow the sixth law of Shneiderman better, as the actions can be more easily be reverted.

Look for alternative designs for the side view

The side view can be moved from the right to the left in the view, as applications often have such functionality at the left of the view. It should as well look into moving some of the functionality onto the map, such as the main views for example. As one now have to navigate much between the different tabs, always show this on the map, can make it more simple to use and remember what was the intention of the action. This may cause that one can have a larger map, or other information which can use the space in a better manner. The prototype did not use the whole available space on the screen, which it possibly should. As there can be much navigation between the different tab views, for doing filtering and seeing statistics, looking for alternative solutions for this can be interesting.

Multiple statistic graphs

When comparing years, there could have been nice to have a graph which shows the total number of sheep observed for each month. One can therefore, more directly, compare the data observed with the last years data. This can also be combined with the area choosing option, such that one can compare how many sheep there have been observed in a given month for the last two years. The graph should also have some information on the x-axis, as this was a bit unclear from the graph as of now. This will solve the problem when no data is showing for trips which aren't containing any observations.
Another wanted ability for the statistic graph was that one could show the sum for all trips, and not just for each trip, similar as the comparing view. Could be solved with radio buttons on top of it for example. The colours used in the graphs should also match the trip colours. This was a bit confusing, and breaks the consistency of the colour usage. As well, depending on the data, the y-axis of the graph should only show integers, and not decimals.
When in "predator mode", one should only show predator data in the graph.

Have the ability to compare with other years, with different views

One should have the option to compare multiple years with each other. For example compare this year's trips with the trips conducted for three years ago. One can create some patterns to see what areas where there were many sheep for a long time ago. This can also be done for months in the concurrent year. This comparing view could have different options. For example by showing data from this year on a map at the left of the view, and another map at the right, which will show the data from the last year. One can therefore easier visualise the difference between those two years directly using two maps side by side.

Elements on the map should be even more interactive

Many of the participants tried to click on more elements on the map, which weren't interactive. Each point on the map would pop up a box with information, the same should have been done with the trip route. When clicking the trip route, one could for example zoom onto the trip area, as well as showing some information about the clicked trip. Another feature which would make the map more interactive, when hovering over a trip, the map should indicate that this trip is selected, for example by alter it's colour a bit. An example could be when hovering over a trip, it's metadata could be highlighted in both the timeline and in the graph.

Have a tutorial before using the application

This isn't maybe necessary for the next iteration of the application, but some kind of tutorial should be implemented for the application when it is ready for deployment. This will make the system easier to use, as well as one can more easy find out what the different predefined cases do, and how the map view works. Especially for the main views and the filters, the user should know what features the application offers.

More focused predefined cases

As the groupings for the predefined cases were mainly intuitive, the cases should have been more focused and more distinguishable from each other. There should be a good indication on which one to use for achieving the farmers' goals.
The overall groupings made sense, but their content could be better, maybe except the time based ones. "Last trip conducted" should for example be moved to the time based section. Under the area based ones, "trip" and "observation" should be merged together. For example if one is to search for observations, trips with none observations should not be included in this analysis. Similar, if one only want to see trips with no observations on the map, trips which contains observations, shouldn't be included. There should as well be possible to distinguish between the different trips when in the comparison mode, for example by using two different maps to show the information, as mentioned.

Movement behaviour for the observed sheep

The option to see where the same flock of sheep have wandered could be an interesting feature. The same flock needs to be tracked and found equal between multiple observations and trips. One could for example see where the route this flock have walked over time. Similar to the sketch in Figure 6.6 in Section 6.

Sheep per kilometer in a given area

Maybe in combination with the heatmap, one can show a number indicating the density of sheep in a given area. If doing this, one should have in mind that the same flock of

sheep could be counted multiple times. In other terms, this should be combined with the movement behavior, so one doesn't count the same sheep multiple times.

Show the number of lambs in the flock

If there are observed lambs in the flock, this should be shown as well. This prototype had this functionality, but the mock data used did not contain any lamb data. In the next iteration, lambs should be included in the mock data to get some other aspects of the information. The mock data should possibly contain trips that have been walked in the field.

Distance and duration of the walked trips

This is information which can be interesting to show for the farmer. Maybe as well weather conditions and other aspects one can add to the trip when it is conducted. If the farmer want's to go in that area once again, it can be interesting to know how the conditions of different trips in that area was, and the duration used on the trip.

More detailed views should be enabled for the external predator data

The ability to show the external predator data as a heatmap for example, would make this information more usable. The state of it now is still usable, but expanding the external predator data features could be interesting. The ability to see density, maybe over years could have some interesting aspects as well. Should be included in the next prototype. Create similar functionality for the external predator data as the observations done on supervision trips, maybe in combination of a "predator" and "sheep" mode.

11. Conclusion

There are numerous sheep which die on grazing fields each year. The sheep farmers need to keep track of their animals. There exists multiple solutions today which tries to achieve this goal, but as lots of information is spread over different systems, not everything is easy to track. There is room for an overview tool which can help with solving this issue. The prototype developed during this thesis and the conducted user tests, gives an indication of what kind of functionality is needed for such a system. Showing both observations from predators and sheep, can give indications of areas of interest, especially when one is to try to find new areas to put out sheep.
After the prototype was developed, there were conducted six user tests with six different participants. These tests gave good and valid feedback regarding usability, functionality and design in the prototype. The optimal would have been to have a user test or a demonstration before a sheep farmer, but at this time, this wasn't possible to achieve. As the spring and early summer contains lambing and putout of sheep on grazing fields, the farmers are quite busy. For a later iteration of this application, there is recommended to conduct such a demonstration for a sheep farmer, as they can give even more feedback regarding the functionality implemented.
Another aspect which has some impact whether a system is functional or not, is how usable it is. If the system is a hassle to use, the functionality can disappear in frustration. With a usable prototype, the participants could focus on the functionality instead. With an average score of **71.4** for the SUS questionnaire, regarding usability, the prototype developed is considered to be over average usable. The participants could therefore use their time thinking of how well the functionality worked and finding out which features missed. Four of the six participants achieved a score which is beyond the *68* limit for a usable system. As two of the participants got a total score in the SUS questionnaire, which was below this limit, there is still room for improvement regarding the usability. For achieving this, the focus should be on getting the system less complex, making some of the functionality more focused, and try to find more predefined cases which can more easily solve the tasks a sheep farmer is to solve. According to the user tests conducted, the displaying of sheep and predator data on a map, is well suited. There is easy to locate the geographical data displaying it on a map, and one can experiment using different shapes and colours to make the farmer interested in the data. For the system to be used on a regular basis, what is visualised, should peak some interest. Wether it is used as a pure predator observation tool, or for holding track of sheep, visualising with shapes and different colours can make the data more interesting. In combination with the predefined cases, which should be more focused on different functionality, will work well with the map solution. The predefined cases will make the system more easy to use, and the farmers would use less time on fine tuning of filters and such.

Reference List

[1] *Ben Shneiderman*. URL: http://www.cs.umd.edu/~ben/ (visited on 02/18/2019).

[2] *The Eight Golden Rules of Interface Design*. URL: https://www.cs.umd.edu/~ben/goldenrules.html (visited on 02/18/2019).

[3] Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Steven Jacobs, Niklas Elmqvist, and Nicholas Diakopoulos. *Designing the user interface: strategies for effective human-computer interaction*. Pearson, 2016.

[4] J. Preece, Y. Rogers, and H. Sharp. *Interaction Design: Beyond Human-Computer Interaction*. Wiley, 2015. Chap. 9. ISBN: 9781119020752.

[5] John D Gould and Clayton Lewis. "Designing for usability: key principles and what designers think". In: *Communications of the ACM* 28.3 (1985), pp. 300–311.

[6] Carolyn Snyder. *Paper prototyping: The fast and easy way to design and refine user interfaces*. Morgan Kaufmann, 2003.

[7] *About usability.gov*. URL: https://www.usability.gov/about-us/index.html (visited on 03/05/2019).

[8] *Usability Testing*. URL: https://www.usability.gov/how-to-and-tools/methods/usability-testing.html (visited on 02/28/2019).

[9] *Planning a Usability Test*. URL: https://www.usability.gov/how-to-and-tools/methods/planning-usability-testing.html (visited on 02/28/2019).

[10] *Running a Usability Test*. URL: https://www.usability.gov/how-to-and-tools/methods/running-usability-tests.html (visited on 03/04/2019).

[11] *Reporting Usability Test Results*. URL: https://www.usability.gov/how-to-and-tools/methods/reporting-usability-test-results.html (visited on 03/04/2019).

[12] *System Usability Scale (SUS)*. URL: https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html (visited on 03/19/2019).

[13] Renien John Joseph. "Single page application and canvas drawing". In: *arXiv preprint arXiv:1502.03530* (2015).

[14] *Geografiske Koordinater*. URL: https://snl.no/geografiske_koordinater (visited on 03/15/2019).

[15] *Degrees/Minutes/Seconds (DMS) vs Decimal Degrees (DD)*. URL: https://gisgeography.com/decimal-degrees-dd-minutes-seconds-dms/ (visited on 03/18/2019).

[16] *More than you ever wanted to know about GeoJSON*. URL: https://macwright.org/2015/03/23/geojson-second-bite (visited on 03/19/2019).

[17] *geojson.io*. URL: http://geojson.io (visited on 03/15/2019).

[18] *Rapportering til Husdyrregisteret - Småfe*. URL: https://www.mattilsynet.no/dyr_og_dyrehold/produksjonsdyr/merking_og_registrering_av_produksjonsdyr/rapportering_til_husdyrregisteret__smaafe.4941 (visited on 01/28/2019).

[19] *Erstatning for sau drept av fredet rovvilt*. URL: https://soknadssenter.miljodirektoratet.no/Soknader/Info.aspx?id=1&soknadsaar=2015 (visited on 01/28/2019).

[20] *Tap av sau til rovvilt historisk lavt*. URL: https://www.miljodirektoratet.no/aktuelt/nyheter/2019/januar-2019/tap-av-sau-til-rovvilt-historisk-lavt/ (visited on 01/18/2019).

[21] *Home site Telespor AS*. URL: https://telespor.no (visited on 01/31/2019).

[22] *Home site Findmy*. URL: https://www.findmy.no (visited on 01/31/2019).

[23] *Home site Shiip*. URL: https://www.shiip.no/ (visited on 03/20/2019).

[24] *Brukerveiledning Radiobjella - Telespor*. URL: http://docs.telespor.org/manual/Telespor_Brukermanual.html (visited on 02/01/2019).

[25] *Telespor iTunes*. URL: https://itunes.apple.com/no/app/telespor/id1336492037?mt=8 (visited on 01/31/2019).

[26] *Home site Smartbjella*. URL: https://smartbjella.no (visited on 03/20/2019).

[27] *NIBIO - About us*. URL: https://www.nibio.no/om-nibio/om-oss (visited on 02/05/2019).

[28] *Norsk lisens for offentlige data (NLOD) 1.0*. URL: https://data.norge.no/nlod/no/1.0 (visited on 02/15/2019).

[29] *Download of data from Nibio*. URL: https://www.nibio.no/tjenester/nedlasting-av-kartdata (visited on 02/05/2019).

[30] *Nasjonalt overvåkingsprogram for rovvilt*. URL: https://www.rovdata.no/Nasjonaltoverv%C3%A5kingsprogram.aspx (visited on 03/14/2019).

[31] *Norwegian Institute for Nature Research*. URL: https://www.nina.no/english/Home (visited on 03/14/2019).

[32] *Databaser og karttjenester i NINA*. URL: https://www.nina.no/NINAs-tjenester/Kart-og-data (visited on 03/14/2019).

[33] *Om Artsdatabanken*. URL: https://www.artsdatabanken.no/omartsdatabanken (visited on 03/14/2019).

[34] *Species Map from NBIC*. URL: https://artskart.artsdatabanken.no/app (visited on 03/14/2019).

[35] *What is Geographic Information Systems (GIS)?* URL: https://gisgeography.com/what-gis-geographic-information-systems/ (visited on 03/18/2019).

[36] *ArcGIS Online*. URL: http://www.arcgis.com/index.html (visited on 03/16/2019).

[37] *Spring Boot*. URL: https://spring.io/projects/spring-boot (visited on 04/03/2019).

[38] *MySQL*. URL: https://www.mysql.com (visited on 04/03/2019).

[39] *What is HTTPS*. URL: https://www.cloudflare.com/learning/ssl/what-is-https/ (visited on 04/03/2019).

[40] *The general HTTP authentication framework*. URL: https://developer.mozilla.org/en-US/docs/Web/HTTP/Authentication (visited on 04/03/2019).

[41] *The Progressive JavaScript Framework*. URL: https://vuejs.org (visited on 04/06/2019).

[42] *Introduction to Vue*. URL: https://vuejs.org/v2/guide/index.html (visited on 04/06/2019).

[43] *Website for Leaflet*. URL: https://leafletjs.com (visited on 04/06/2019).

[44] *Website for Material Design*. URL: https://material.io (visited on 04/08/2019).

[45] *Highcharts Website*. URL: https://www.highcharts.com (visited on 04/08/2019).

[46] *Highcharts, Highstock and Highmaps documentation*. URL: https://www.highcharts.com/docs/ (visited on 04/08/2019).

[47] *Highcharts Non-Commercial License*. URL: https://shop.highsoft.com/faq#Non-Commercial-0 (visited on 04/08/2019).

[48] *Kartverket - Lage kart på nett*. URL: https://www.kartverket.no/data/lage-kart-pa-nett/ (visited on 03/12/2019).

[49] *Vilkår for bruk av Kartverkets åpne data*. URL: https://www.kartverket.no/data/Lisens/ (visited on 03/12/2019).

[50] *Kartverket Website*. URL: https://www.kartverket.no (visited on 03/12/2019).

[51] *Behandle personopplysninger i forskningsprosjekt*. URL: https://innsida.ntnu.no/wiki/-/wiki/Norsk/Behandle+personopplysninger+i+forskningsprosjekt (visited on 04/03/2019).

Appendices

A. Mock Trips

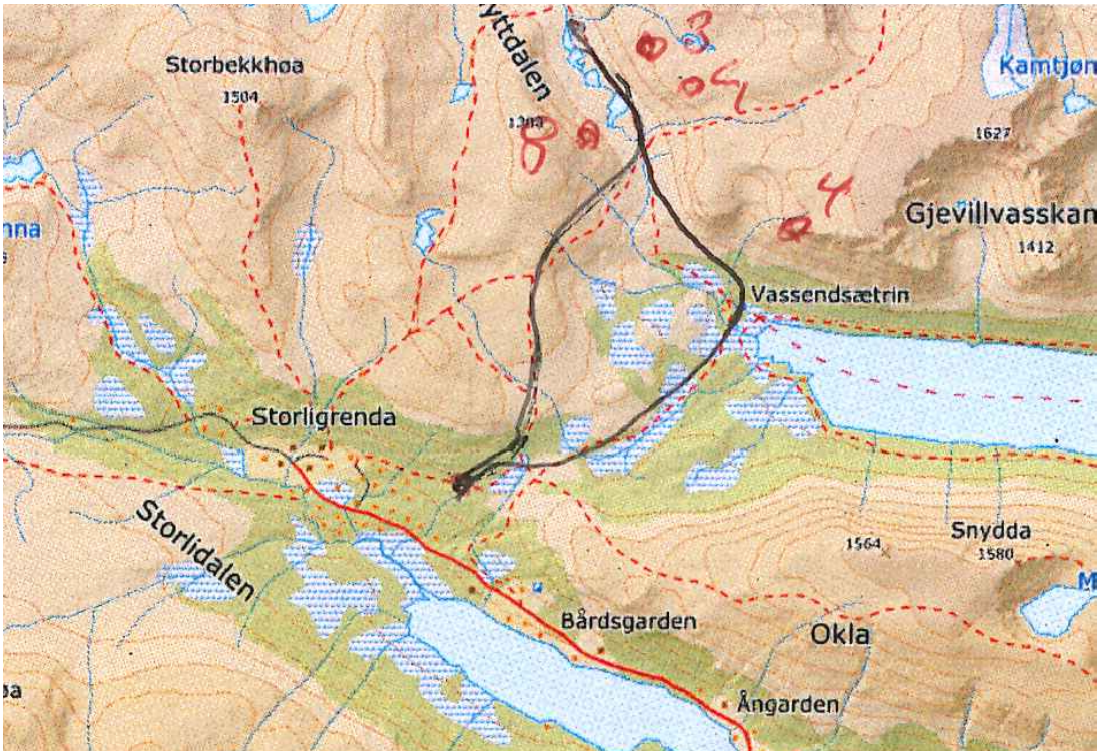

Figure A.1.: First Mock Trip

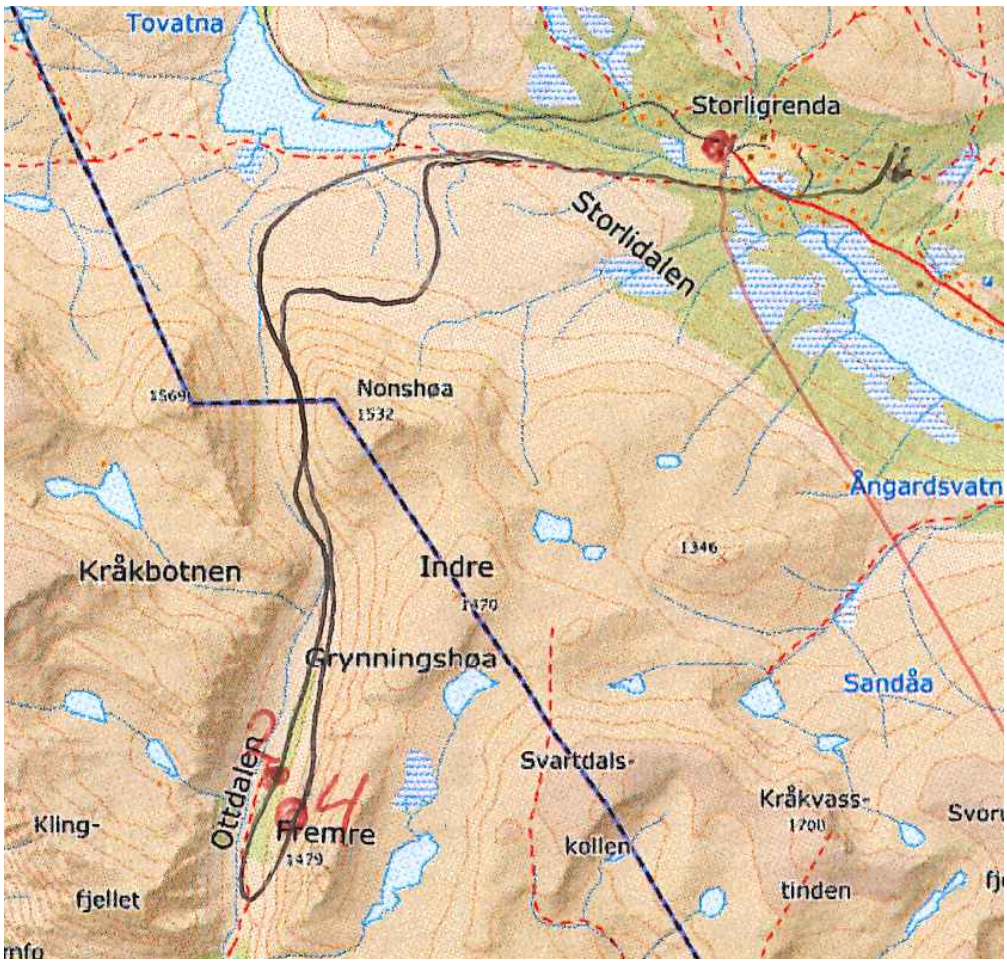

Figure A.2.: Second Mock Trip

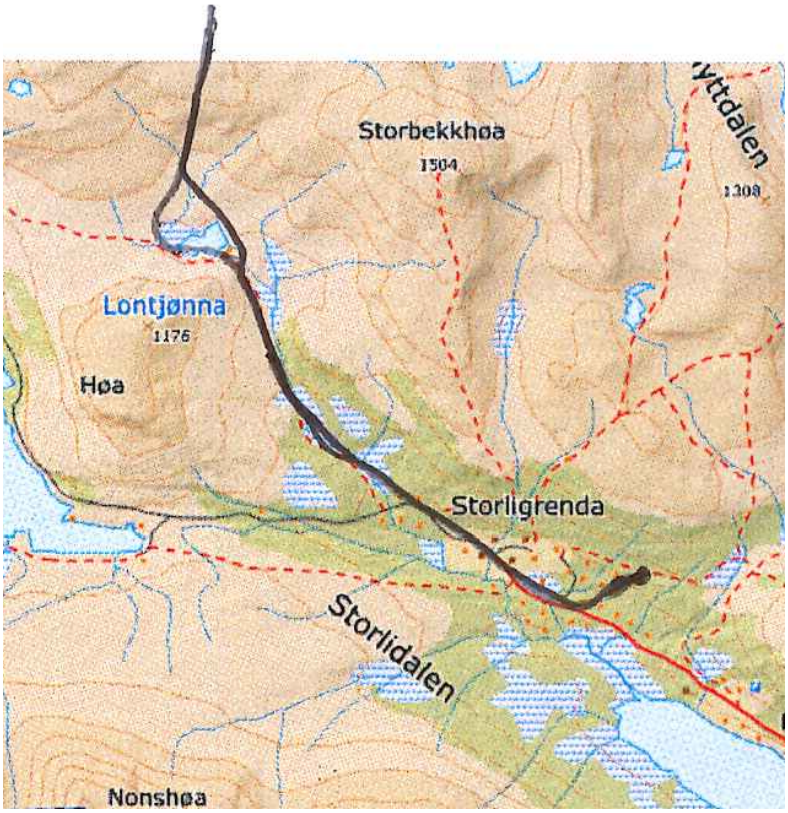

Figure A.3.: Third Mock Trip

Figure A.4.: Fourth Mock Trip

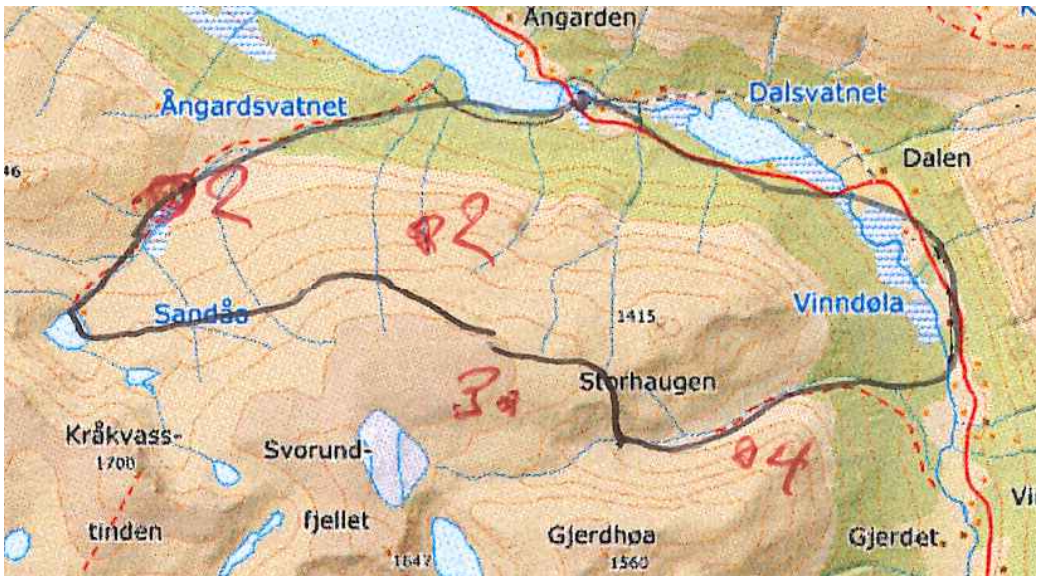

Figure A.5.: Fifth Mock Trip

B. Predefined Cases Screenshots

Each of these figures is showing the analysis view which is shown when clicking the predefined cases. The data shown in the figures, are based on the mock data, described in Section 5.4.

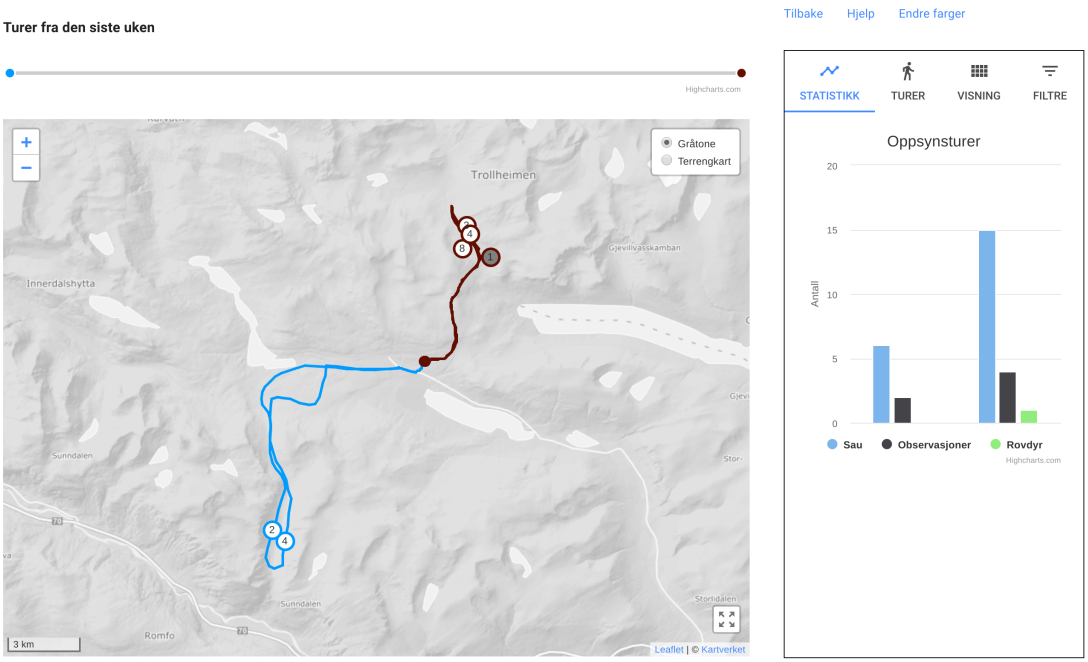

Figure B.1.: Trips from Last Week

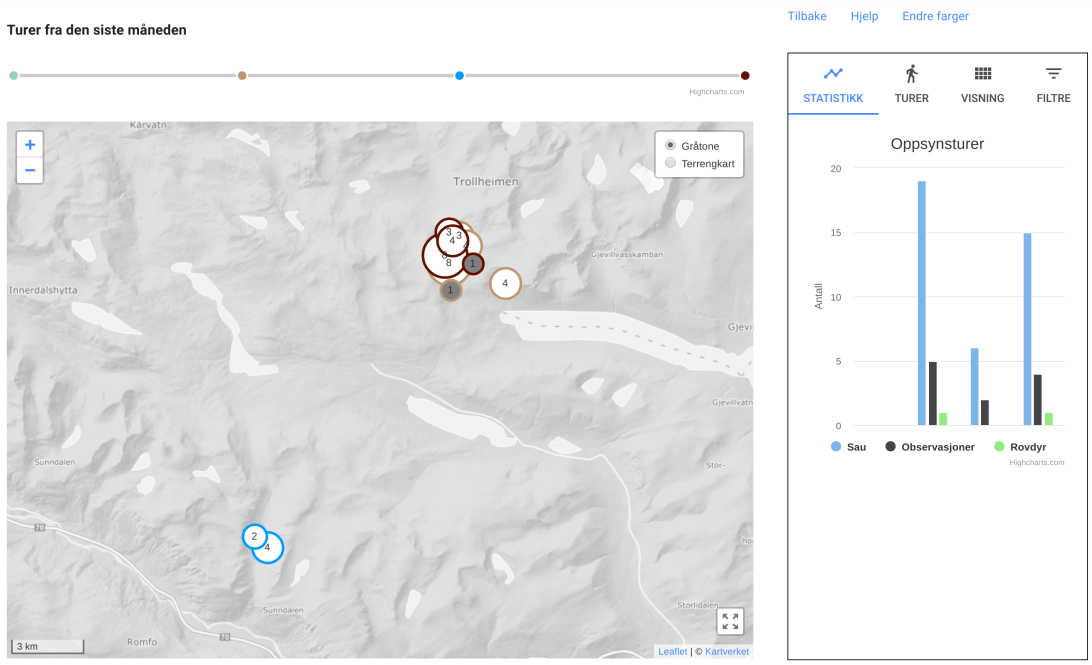

Figure B.2.: Trips from Last Month

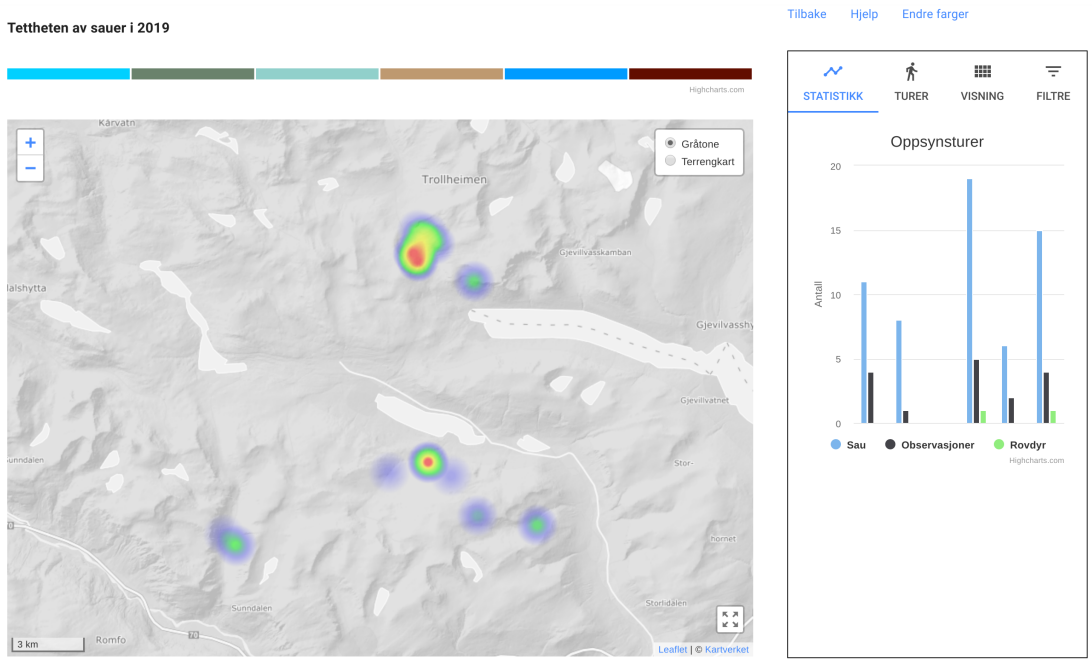

Figure B.3.: Density of Sheep from this Year

Figure B.4.: Trips from a Given Area

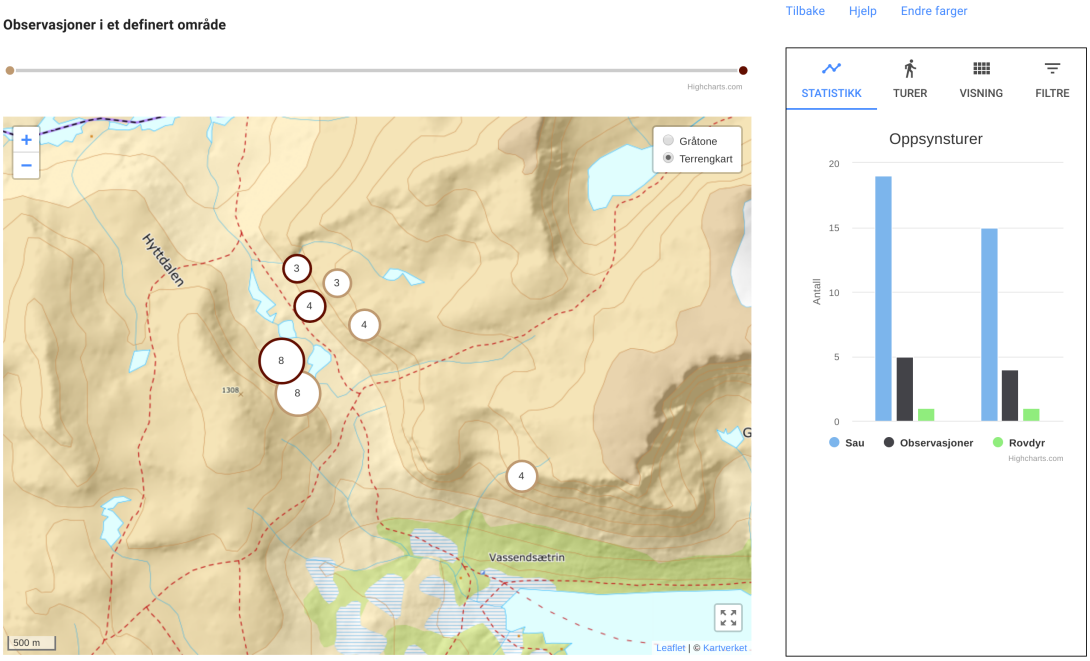

Figure B.5.: Observations from a Given Area

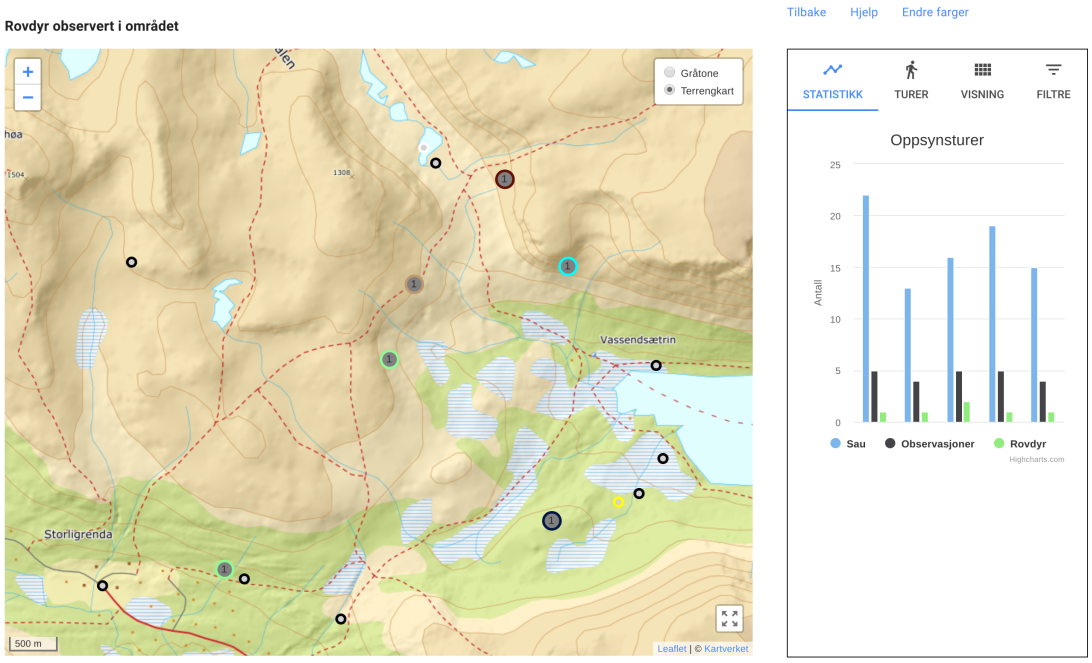

Figure B.6.: Predators from a Given Area

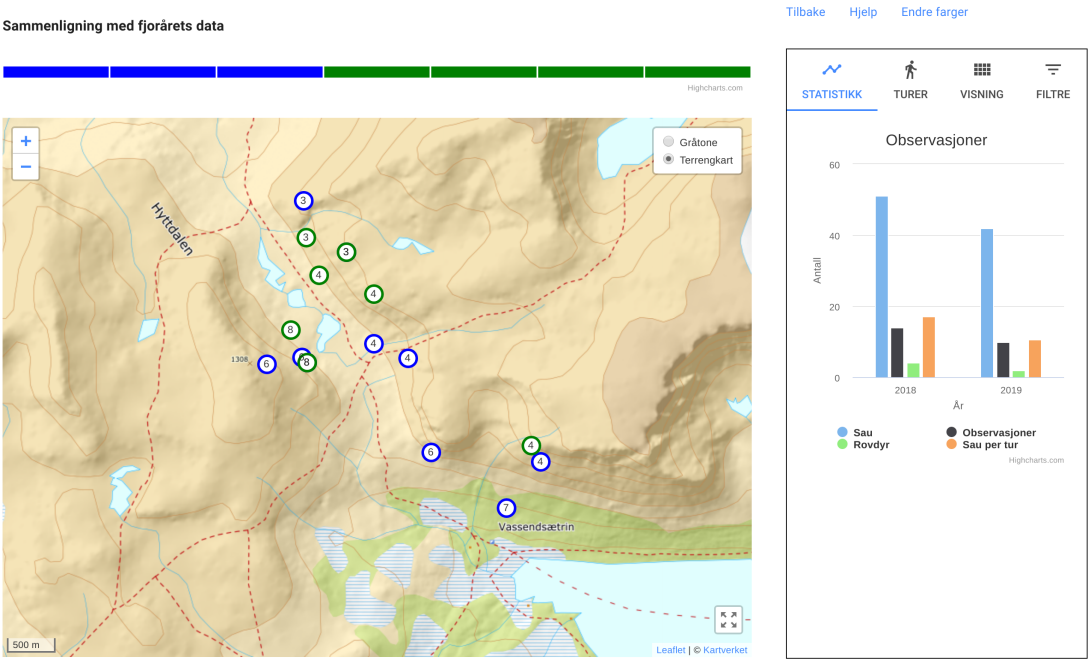

Figure B.7.: Comparison of Trips from last Year in an Area

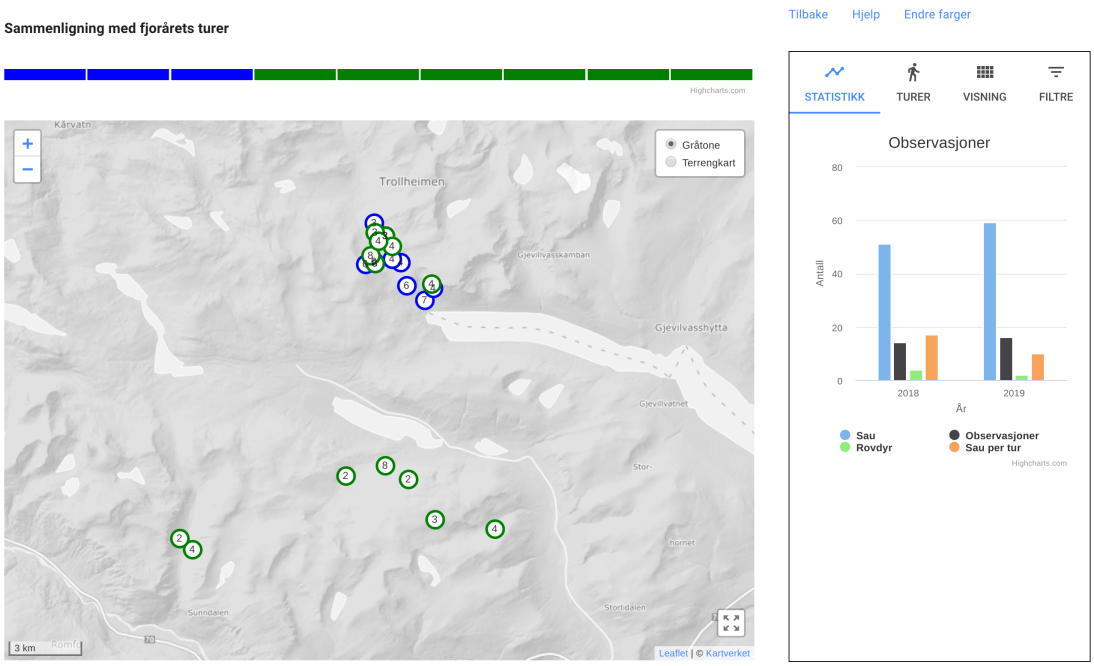

Figure B.8.: Comparison of Trips from last Year

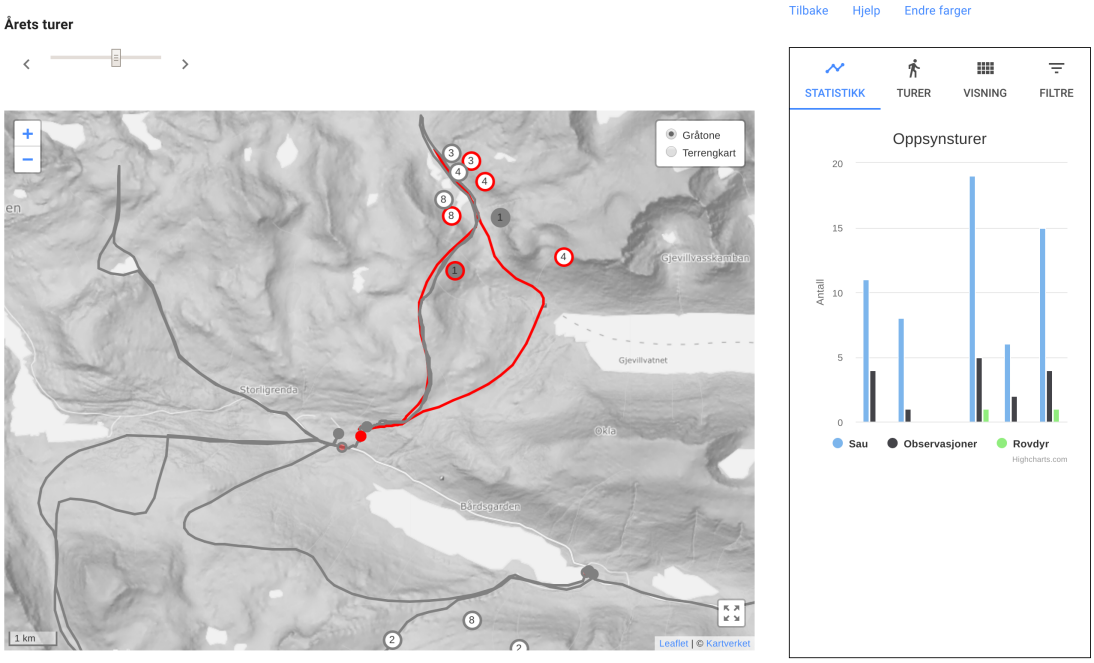

Figure B.9.: Comparison of Trips from this Year

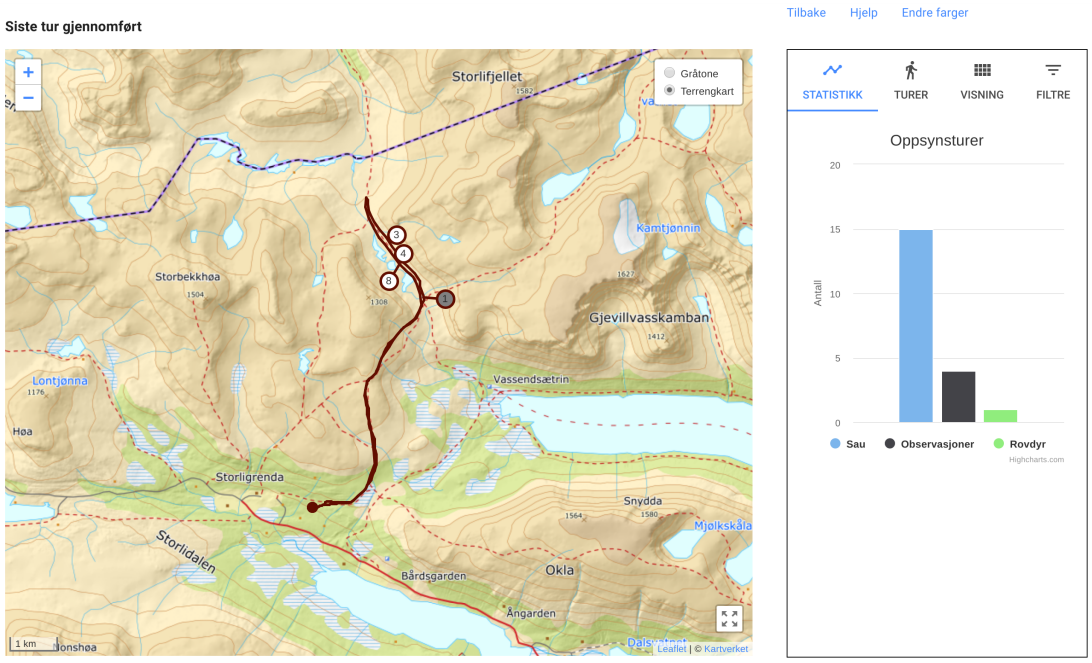

Figure B.10.: Last Trip Conducted

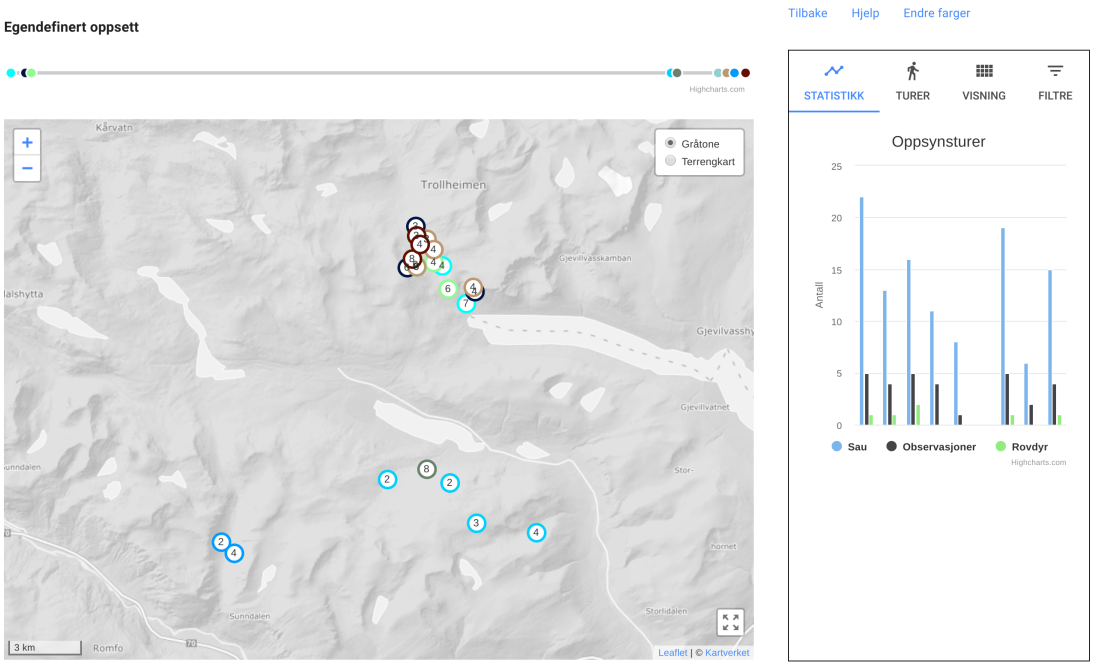

Figure B.11.: Custom Setup

C. Notes from the User Tests

This section includes all notes taken during each user test. The results and the conclusion are based on these tests. For the interview part, not all questions were asked to all subjects, as some of them already was answered or not relevant for their usage.

C.1. Participant One

This user test started with the usage of the manuel filters. The user therefore got some tasks to solve. The following were noted during this test:

Overview Tool

Edit the Colour to a Trip

Starts to investigate the view. Sees the map with different colours, what do the dots on the map means. Misses a legend which displayed all the time. Sees the timeline at the top and tries to understand what the meaning of it is. Understands gradually that it is a view of all the trips in chronological order based on when they were conducted. Goes to the trip view and understands that the points on the map maybe is coherent with the trips. Interprets that the points are observations. Clicks on multiple trips and sees that the map zooms, maybe randomly, at a point on the map. Gradually sees that it zooms into a trip. Some colours are difficult to see, suggest the usage of dark colours, but sees that this is difficult in combination with the map view.
Finally presses the "change colour" button. Thought that the placement of this element was quite weird. The colour editor were to complex, had an transparency option and showed hex-colours which a sheep farmer maybe not have need for.

Show Trips Conducted in 2019

Had already been into the trip view, and understood that they had the option to be filtered away. Removed the three trips that were conducted during 2018. Commented that if there were many trips, this job would have been more complex. Maybe an option could have to group the trips by year and then filter them away by year.

Change Map Type to Wanted Type

Went straight to the map type changer (up right of the map). Tried to understand the info of the map tiles, did not understand all of them. Notes that if the person have been

from the area, possibly more knowledge of the area have been known. Likes the use of contour lines in the regular map.

Find out how Many Sheep which were Observed i the First Trip Conducted in 2019

Goes to the trip list again. Hovers over some of the trips and finds out that the number of observations pops up. Goes back to the statistic view, and sees the same information in the graph. Does not understand immediately that each column group belongs to each trip, but finds this out when hovering the columns. Thought this should be showed from the start.

Find out How Many Observations Conducted on each of the Trips

Filtrates the graph, clicking on the buttons below and removes the predator and sheep columns on the graph. Does not know how many there were on the trips.

Find Data for how many Sheep there are in the Grazing areas around

Goes to the filter tab – thought there was filtering of trips in that view, which isn't the case. Does not understand the text "external statistics". Activated the grazing area filter, but as the zoom was quite high (deep onto the map), the contrast were difficult to see. Wasn't easy to see that the areas popped up. When zooming out, it was more clear and understandable. Tests the predator data as well. Did not try to press the grazing areas on the map, and hence did not get any more information about the areas beyond the geographical grazing areas.

Zoom in on Trondheim and Find the way Back to the Trips

Uses both the zooming buttons upper left on the map, as well as the touch pad. As the pad is a bit unfamiliar, the zooming buttons are preferred. Does not use the touch screen. Clicks on the trip point on the timeline to go back to the area with the trips.

Find out if there are observed golden eagles in this area

Clicks the help button and finds the golden eagle symbol, which is a white dot with gray fillings. Does not see that the dot is white before hovering over the table row as the background is white when not hovering. When hovering the row gets a gray background, sees that the sign is white. Turns on the predator statistics, and sees that there have been observed golden eagles in the areas.

Tasks

You just have been on a supervision trip. Find out how many sheep that were observed in this trip.

Goes to the "Last week" case. Comments that there are other colours in the graph other than the trip colours. Should probably be the same as the trip. Understands that there are multiple trips in the graph, but misses a clear knowing of this.

The grazing season have ended and all sheep have been collected. You want to see where there have been observed most sheep and the density of these

Looks at the cases and is wondering which one to choose. Picks the observation case in the area group. Struggles to understand what to do next. Sets the dates but does not know for a start how choosing an area works. Thinks that maybe it is to zoom into the correct area. Is a bit annoyed that the starting point is whole of Norway and not just the area where the sheep have been observed. Goes back from the analysis view, but expects the choosing area site when going back, which is not the case. One is sent back to the predefined views view. Tries again, and expected that the fields were pre-filled from last time. Tries to go further but no response from the system as there wasn't zoomed into an area.

You know there are missing some sheep from a given area. Find out if there have been observed any predators there

Presses the predator case in the area grouping, and zooms into the correct area at once. No indication on what case is chosen when in the choosing area view. The analysis view shows and predator data is loading. Notices that the predator data shown also includes data not from this year, thinks that it should only show predator data from the same time area chosen. Inconsistent use of points for external predator data, and predator observations done on a trip, should possibly use the same stylings.

General Notes

Starts to use the help section and reads it. Does not understand what the views means before trying them. Could maybe have been solved with a tiny image showing the view. Did notice that the observations on the map had the same colour as the trips. The term low zoom were inaccurate, what does it mean. Did not understand the meaning of the trip borders. Did not give any relevant information. Too much information on the map, quite complex at times. Did not use the filters to a great extent to prevent the complexness.
Tried the "trips of the year" with the time view. Used the bar to through the trip where the red one is in focus. When using the bar, the user thought that one trip was moving around, instead of going to the next trip. Understands this gradually. A bit weird colours of choice (red for selected and gray for the rest), but did understand why

did was done. Thinks that too much information lies hidden behind different meanings, too much to remember. This includes usage of colours for trips and observations. Much one needs to remember for each point, for example which trip they belongs too. When changing colours of the trips, remove unnecessary information, only show the colour changer. Likes the grouping, rather than the long list for the predefined.
The statistics did not have any interacting elements, and was therefore a bit diffusing compared to the rest of the tab menus.
Thinks there are too many options of choice. Thinks the application should focus more on the predefined views and find the optimal for these. Could maybe implement a regular and expert view, where one can edit the filters and views in expert mode if wanted. In the regular view, the filters should be disabled and maybe not showed at all. As there was too many options to choose from, it got a bit messy and complex. This was also the case for the map view – to much information were at some points visible on the map, and should maybe have been toggled by default when moving out. Did not understand why there were two different views for regular view and the pointing view. Thought that the grouping should be applied at all times for the pointing view, which would have made it a bit more easy to use.
Thinks the statistic graph is showing "boring" data. One suggestion is to group for each year, how many sheep which were observed by each month in the comparing view. Can also be interesting for only years or maybe down to week view. Compare from one month to another.

Interview

What did you like with the application?

That the application was map based as it makes the information quite graphical and not so much text. Very concrete. The grouping of the predefined cases gave sense and made the front view more lucid.

What did you not like about the application?

It was a bit unfocused, too much information showed, too many choices in regards to filters and views, and hard to distinguish between those two. Maybe have two modes for the application, where the standard mode only includes the usage of the predefined cases where one cannot do the filtering for one self. The expert mode can include this and will be intended for expert users. In general, have a more pointed focusing of the application with less choices.

The predefined cases, did they make sence?

The area based were a bit confusing as it was difficult to know how the choosing of area was. Maybe draw a rectangle on the map for the searching area instead of using the zoom level. The dates for this view should have been removed, or one could have predefined dates. The comparison cases did not give so much relevant information.

Was the help view useful as a legend and description of functions?

The legend was useful, but should have been showed at all times, or maybe in the side view. So one could use it while the map was "active". For the view types, a bit difficult to understand what the meaning of the was. Too petty to choose between the regular view and the sizing view, should probably find out which of these two are the best for this usage. The views should have a preview maybe, and be moved from the side view to be chosen on the map. Supplement with icons could also be useful.

Do you have any suggestion to new functionality or improvements?

Preview of the maps, such as it is done at ut.no. To much unused space in the application, could use the whole screen instead of the fraction used now. Show the dates on the timeline under the points, makes it more clear what the intention of the timeline is.

Key Notes from the Test

What should be edited?

- The area based cases had some problems, such as setting dates was possibly too petty, difficult to set the wanted area and show what case is selected.
- The analysis view and map was too complex at times, and there were too much to remember to get hold of the information. Suggests to show the legend at all times and maybe not hide too much information.
- The use of colours in trips and changing of these. The columns in the graph should have the same colours as on the map.
- Inconsistent use of styling of predator data, in regards to own observations and external.
- Have a graph which can be structured into months where one can see the progress easier, such as for this year, or when comparing years.
- Remove some filters and combine som views, too petty choices.
- Have a predefined case where one can choose which trips to show in the analysis view. Should be possible to see information about the trips in this view as well.
- More pointed focus for the application and less choices.
- Use more of the unused space.

What did work as intended?

- That the map contained contour lines, so one knows how the terrain looks like.
- The graphical representation of the trips, such as on the map and the usage of this.
- The grouping view was nice and made the map less complex.

C.2. Participant Two

The participant uses the application a bit before looking into the cases.

Tasks

You just have been on a supervision trip. Find out how many sheep that were observed in this trip.

Have read over the different predefined cases, so the subject saw the "last trip" case. Clicked that one and saw that this was the correct view. In addition to looking onto the trip's date, it was clear that the trip were conducted today. Tries some of the different observation views, such as the heatmap. Likes the heatmap as it gives an indication on where there have been observed most sheep and predators.

The grazing season have ended and all sheep have been collected. You want to see where there have been observed most sheep and the density of these

As the case indicates that there is to be found an area, the subject chooses the "observation" case in the area grouping. Sets the dates, but does not understand how the choosing of area works. Goes back to the overview and chooses the time based case "density of sheep in 2019". As the view now is showing all trips of 2019, and the observations are showed as a heatmap, the subject thinks that this is the correct view as the heatmap indicates the density of observed sheep. Tries out the different views, such as the grouping view and the

You know there are missing some sheep from a given area. Find out if there have been observed any predators there

Chooses the case "predators" under the area grouping. Same as the last case, sets the date first, and does not zoom on the map. Tries to press the go further button, but no feedback from the system is shown. As last time, does not understand how to find the area, and goes back and chooses the comparison case of this years trips. Switches to the heatmap view and goes to the filters. Turns off the sheep observations and turns on the predator observations. Loads the predator data as well and finds out that there have been observed predators in the area.

In the area around Hyttdalen, there have been observed 34 sheep during this year, but you cannot remember how many which were observed last year. Find this number

Goes back to the overview. Looks onto the comparison cases, and chooses the comparison for this and the last year. Misses that the only area interested in are near Hyttdalen, and not the entire data basis from 2019 and 2018. This is possibly because the subject did not understand how the area choosing worked. Finds the total number of sheep from 2018.

On trip "Tur tre", there wasn't observed any sheep. You want to walk this trip once again to see if there are some sheep there this time. Find the route

Stays in the comparison view from last case. Tries to filter away all trips which aren't the selected trip. This works, but since the trip route isn't filtered on, nothing on the map shows. Did not try to turn on the trip route and did not understand the reason why nothing were showed on the map. As the trip did not contain any observations, nothing showed. Went back to the overview and tries the observation on area based cases. Now the subject tries to zoom the map, and finds the correct area. As well understands that the trip isn't showing as there aren't any observations in it. Turns that filter on and finds the correct trip route.

The government wants information on the data collected last year. Show only data from 2018.

Tries to filter away all trips, but understands that no trips from last year are present. Goes to the overview and uses the "custom setup" case. Filters away all trips and turns on only trips from the last year. Finds the correct data.

You want to see the trips done in this year and see the data from them stepped

Stays in the current view and un-checks all irrelevant trips and checks the trip wanted. Does this to all trips from this year. Did not use the time view which would have solved this more easy.

General Notes

The application crashed during changing of the main views. Did refresh the application, and it worked again.

Interview

What did you like with the application?

Liked the way that anything could be changed and edited, even on such a detailed level as wanted. The map experience was nice and smooth. Liked that there were high

granularity of the data and the possibility for detailed information. The heatmap was quite nice and did fulfilled it's intention, too show the density of observations. The design was simple and easy.

What did you not like about the application?

Much information to take into one self and much to grasp over. At the start, I was a bit puzzled, and did not know what to do. No clear road to follow to achieve the goals. A tutorial or introduction to the system could have solved this. A bit difficult to switch between the different views, much data which was to be processed and showed. Much complexity because of much data. The benefit outweighs the experience. It was difficult to search for the area one wanted to search in, did not understand that one.

The predefined cases, did they make sence?

Did not think about that the predefined cases where largely different only by the filters and the views. Else they worked mostly as intended, but liked more the manuel use of the filters and the views.

Was the help view useful as a legend and description of functions?

Did not use the helping during the session, as it was a bit too hidden.

Do you have any suggestion to new functionality or improvements?

Automatic zoom on the area where there have been observed sheep in whole of the system. Then one could easier search for local areas in that area after sheep. The ability to compare two cases or years with each other, with two maps side-by-side. For example with data from one year at the left and another year to the right. Add functionality from digital bells so one can compare these data view with the actual position of the sheep in real time. In manner of the design, a bit boring maybe, but the design should be simple and easy to understand, which it is.

Key Notes from the Test

What should be edited?

- Choosing the area for the area based cases, was not intuitive and difficult to use. No feedback if not zooming the map first and then trying to go further.
- Does not click much on the map, so it should be more clear and intuitive that the map is clickable. Some information are therefore lost.
- As some functions, information and views of the application can be a bit complex, a tutorial could have made things easier. Mentions that the application could have

made less complex, but it's purpose could loose some of it's meaning if this would happen.

- When starting to use the application, was a bit helpless, did not know where to start.
- Much information, which should either be made more intuitive, have a more visible legend, or create a tutorial which can show how to use the system and achieve some goals. The benefit outweighs the experience.
- The help button were too difficult to see and did not want to interact with it.
- The ability to compare two cases or years with each other, with two maps side-by-side. For example with data from one year at the left and another year to the right.
- Implement the digital bell solutions with this, so one can compare and see the position in real time compared with the observations done on trips. Also with predator data.

What did work as intended?

- Likes the way they predefined cases are grouped, makes it more clear what the meaning of each of them is and how to use them. Not that all cases made clear sense, but the groupings made an intention of usage and meaning.
- The use of heatmaps on the map as it makes a good indication where there have been made many observations.
- The grouping view made sense as well, as the map gets less complex when this view is turned on.
- The colour indications between the trips, list, timeline and points on the map belongs to each other.
- That anything could be edited, but mentions that this is to the subjects background with computers. Maybe this would not be the case for a sheep farmer.
- The use of the map worked quite well and smooth.
- The level of detail and high granularity of data.
- Easy and simple design.

C.3. Participant Three

Tasks

You just have been on a supervision trip. Find out how many sheep that were observed in this trip.

Choses the "week view" in the time based grouping. Tries to filter away the other trips, except of the correct one. Finds the data. Goes back and wants to try another predefined case. Tries the area based trip choice. Does not zoom the map and goes up. Afterwords the subject sees the last trip conducted choice.

The grazing season have ended and all sheep have been collected. You want to see where there have been observed most sheep and the density of these

Tries to continue in the same view, but sees that not all trips are present in that view. Goes back to the overview. Picks the time based "density in 2019" under time based grouping. Sees the map, but wants to try out another main view. Understands the meaning of the heatmap, but misses that nothing indicates the values of the heatmap. Likes the other main views better, specially the regular and the sizing view, as these two gives an exact number of how many sheep are observed where.

You know there are missing some sheep from a given area. Find out if there have been observed any predators there

Enters the area based view with predators. Changes the map type to the terrain map. Changes the map type almost every time the map is involved as all maps defaults to the gray map. Struggles to find the area which is to be searched in. Insecure of how exact the location chosen have to be to include the trips. Not as intuitive as intended to find the area. Enters a area where there are no trips, and the system gives feedback. In addition, sets the dates for the search, for 2019. Finds the area and enters the analysis view. Doesn't understand the difference between the observations done on a trip and the predator data from Rovdata.

In the area around Hyttdalen, there have been observed 34 sheep during this year, but you cannot remember how many which were observed last year. Find this number

The trip choice under the area views are chosen. Sets the dates there, and is a bit annoyed that there aren't predefined years, that this have to be set all the time. Tries to change the main view to see if it make any other difference. Does not know wether the same sheep are observed multiple times over different trips. Want a unique number which takes this into account, but sees that there isn't support for this at the moment.

On trip "Tur tre", there wasn't observed any sheep. You want to walk this trip once again to see if there are some sheep there this time. Find the route

Uses "trips" under area based groupings, and uses dates. Goes back and uses the trips of the year instead. Tries to filtrate away all trips except "Tur tre", but the map shows nothing as the trip route filter is turned off. Could the system maybe have set the filters automatically if such a case appeared? Does not find the wanted information, goes back to the overview. Annoyed of the position of the back link, should be placed at the left. Enters the year comparison, and tries the same. Did now understand that the trip did not show last time, since no observations were present in that trip.

The government wants information on the data collected last year. Show only data from 2018.

Enters the area based trips and sets the dates there for whole of 2018. Should not have to find the area each time one enters the area view. Either "remember" the last setup, or always start where all the trips have been conducted. This because all trips conducted by a farmer are in the same area, and the system is to be used by one farmer only. Finds the correct data.

You want to see the trips done in this year and see the data from them stepped

Uses the comparison grouping with the year's trips, but does not use the slider to look at the trips on the map. Instead, uses the information shown in the graph. Likes this.

General Notes

Understands that there are multiple trips shown in the graph view in the statistics. Clicks regularly on the points on the map to find information about the points. Did assume that no sheep were observed when nothing were showed from trip three. Preferred to use the custom view so the filters could be set manually. Did not think about that the predefined cases contained different filters. Did understand what the map data meant.

Interview

What did you like with the application?

Liked the design, simple and lucid. It made sense. Liked that there were many choices one could make, a lot of freedom in the application. To see where predators were observed was liked. How the predefined cases where structured made sense and made the overview quite lucid. Liked the statistic graph and thought it gave valid information back.

What did you not like about the application?

Should possibly be a tutorial or something so one understands the meaning of the program and how to interact with it. "I don't have sheep all around the country" - therefore the area based cases should auto zoom into this area. As well quite confusing when one had to set the dates in the same view. Expected that the trips was set in the area view when the dates were set. Would like to see areas with predators not just containing conducted trips. For example potential areas where no trips have been conducted.

The predefined cases, did they make sence?

Would like the opportunity to save own setups for later use. Else some of them worked as intended.

Was the help view useful as a legend and description of functions?

Did not use the help view.

Do you have any suggestion to new functionality or improvements?

If one is to use area based cases, one could for example first choose the area and dates, then the trips, and finally let the user decide what case to be shown. Want to see predators in the same view as the sheep. Would like the option to select between multiple years, months and weeks, not just have predefined for the last week, month and year. The ability to save custom setups. Wants to see if same flock of sheep have been observed over multiple trips and some behavioural information about this.

Key Notes from the Test

What should be edited?

- "Last trip conducted" should be placed under the time based scenarios instead of others.
- There should be a legend or something indicating a value of the heatmap. For example if the colour is red, it should give an exact number of what this means.
- The program should remember which map type is used between different views.
- Search map view is difficult too zoom into the area wanted. Should auto zoom into the largest area where there are found sheep. A sheep farmer doesn't have sheep over the whole country. This includes the dates which is set in this view. Possibly show the trips at this view as well, and filter them when setting the dates.
- Make a more clear difference what the statistical data from Rovdata is.

- Thought about unique sheep, and their movement behaviour when trying to find all sheep in an area. Create something that can indicate and map different observations to each other. If the same flock of sheep have moved for example.
- If the system tries to show a trip(s) with no data currently visible, since filters are turned off, turn some filters on automatically such that something is showed on the map.
- The meaning and the functionality of the time view is unclear.
- The ability to save a custom setup of the trips, filters and the views for later usage.
- See areas with predators where there haven't been conducted any trips.
- Make the system go through more, less detailed, phases where the user can answer different questions and then enter a analysis view which is meant for that purpose. For example if one wants an area based view, one can find the area, and then decide what case to use, or even combine cases. Make the application more stepwise.
- Would like easy access to see data from different time periods, not just this year, last month and last week. Could for example give the option to select what month to search into.

What did work as intended?

- Likes that numbers are present in the observations indicating the number of animals in each point.
- Did understand that there were multiple trips shown in the graph, but a bit insecure if it was such from the beginning. But understood it intuitively.
- Interactivity with the map, such as clickable points etc.
- The different main views, like that one could switch between them.
- The design was clean and lucid, understandable.
- The ability to see where there were observed predators, both from trips conducted and the statistics.
- Liked the freedom the application gave, nothing were locked and one could edit everything.
- The main groupings of the predefined cases.
- Did like the use of statistic graph on the side.
- Lucid design and visualisations.

C.4. Participant Four

Tasks

You just have been on a supervision trip. Find out how many sheep that were observed in this trip.

Wonders a bit what cases are, and does not understand how they work and what they do. Decides to use the "trip view" in the area grouping. Does not know what date to set. As the only information the subject knows, that there have been conducted a trip today, the subject tries set the today's date. The date is set in both fields, but when trying to go further, nothing happens. As the view is not zoomed into, and the dates ignores the time, there is no time gap to search into. Goes back to the overview and enters the week view. Used the timeline to zoom onto the correct trip.

The grazing season have ended and all sheep have been collected. You want to see where there have been observed most sheep and the density of these

Exits the last view and enters the overview. Uses the "density in 2019" case in the time based groupings. Assumes that the heatmap shows the density of the sheep observed, but is not sure. Want's to see exact numbers and switches to another main view. Does like that one can switch between the main views to show different kind of information.

You know there are missing some sheep from a given area. Find out if there have been observed any predators there

As there is mentioned area in the task, thinks that this maybe is a good idea. Picks the predator field in area based. Difficult to know wether one has too zoom or not, and that the area zoomed into will be searched for. Uses some time finding the dates to use, the usage of dates should be made optional. The zoom is a bit tricky. Enters the statistic view. Sees the dots on the map, but doesn't know what they mean. Does not interpret the help-link as a legend description, and does not use it. Misses a visible legend, and want's a description on where the external predator data comes from.

In the area around Hyttdalen, there have been observed 34 sheep during this year, but you cannot remember how many which were observed last year. Find this number

Uses the touch screen when interacting. Not sure what predefined case to use. Decides to use area based, but are not quite sure if one should use trip based or observation based. Doesn't know the difference between those two. Ends up using observation, and set the dates for the last year and finds the correct area. Comments that the toggling button makes sense, if the trip should be hundred percent into the area chosen. Enters the analysis view. Want's the program to sum the number of sheep observed automati-

cally. The selected view have no support of this as of now. Thinks it is unnecessary to distinguish between trip and observation in area based views.

On trip "Tur tre", there wasn't observed any sheep. You want to walk this trip once again to see if there are some sheep there this time. Find the route

Enters the overview. Would like a list over all trips, such that one could either pick one or multiple here and then enter the analysis view. Picks the last month, assumes that the wanted trip is in this selection. Filters out all trips, except of "Tur tre". As this view doesn't show the trip route, nothing shows on the map. Assumes that this is the case, therefore goes to the filters and turns on the trip route and sees it on the map.

The government wants information on the data collected last year. Show only data from 2018.

Have done this in a previous task, so skipped over this task.

You want to see the trips done in this year and see the data from them stepped

Enters the overview and picks the time view "density in 2019". Turns on all visual filters, such as sheep and predator observations, trip route and observation points. Filter out all trips and selects the first one and shows the information. Does this for all trip conducted in 2019.

General Notes

Did generally not read the predefined cases well enough. Did among others skip the reading of the others cases, as others can be vague and easy ignorable. Did miss some task as the predefined wasn't read good enough. Did generally come to a good enough solution nevertheless. Did not click the map as much, and hence some information were lost. Uses the touch screen sometimes, in addition to the touch pad when navigating and interacting. Some elements, such as hovering are not working quite as well when using the screen.
Notices that there are mainly only different filters for each case, and different trips showing. The header in the area view is unchanged, does not know which case was selected. Needs to set the date when comparing between two years, should not be the case and should be pre-set. In the comparison view, understood that blue indicated trips from 2018 and green was from the current year.

Interview

What did you like with the application?

The application was tidy and clean, good usage of colours, did always know where I should see. The navigation between views worked well. As the usage of the map was

smooth and worked as intended, this experience was well. Visualisation of the trips, observations etc. worked well on the map, especially in combination with the colours. Liked that one could edit the filters "on the fly", and the interaction was fast and dynamic, as well as it was intuitive. The icons gave mainly sense, except the one for "the last trip conducted" case.

What did you not like about the application?

The way the cases was grouped, time based and area based did not agree with the expectations. "Last trip conducted" should for example be listed under time based, and the difference between trip and observation in the area based, was not clear what the difference was. The naming in the time based should have been more consistent, and maybe contain more options for what times to choose. The comparison grouping pretty much made sense. The meaning of the timeline and the purpose of it was not clear. Did think it was a progression bar or something, which was draggable.

The predefined cases, did they make sence?

Worked mainly as intended, but as mentioned, some of the groupings were inconsistent and not all of them did what the expectations were. Thought that the title in the page should have been hidden as it made the view more complex and was unnecessary. Did not understand why one could switch between the two views in the main views, one with groupings and one with the list. Stick to the grouping one.

Was the help view useful as a legend and description of functions?

Did not know that it existed. Should use another word than "help", as this can indicate other functionality than explaining functionality and showing a legend. Thought it was more like a helping tool, such as screen reader for example. Maybe use a question mark, and another location for it. Longer down on the page, where such interacting elements have more tradition to be located. Change colour made more sense, most because of the intention of the text.

Do you have any suggestion to new functionality or improvements?

The titles should be larger, throughout the application. Mainly too small, and hence made them less readable and easier to skip. At least when they included important information on what case chosen etc. The "back" button should be moved on the left side.

Key Notes from the Test

What should be edited?

- Assumed correct with the meaning of the heatmap, but misses a deeper explanation, and want the system to show an indication on how many sheep the heatmap are indicating. Such a colour scale which shows what blue, green and red means and how many individuals they are indicating.
- The searching for area view are a bit complex, as well as the dates should have been set before, or made optional.
- The help view was a bit hidden, and the name "help" is not indicating enough it's meaning. Maybe should say legend more clearly.
- Remove predefined cases which are fairly similar, such as trip and observation in area based. Or make them more distinguishable. Call trip "trip routes" instead for example.
- Be able to switch between different statistic views, for example a sum of all trips in selected, not just per trip. Similar functionality as in the comparison cases.
- Want a list of the conducted trips in the overview where one can pick trips for one self. Similar to the custom mode, but as a stepwise view instead.
- The colours used in the graph should be the same as the trips' colours.
- The vertical line in the graph should only show integers and not floats.
- The meaning of the timeline is not clear.
- The area view should show what case is selected.
- Not all predefined cases made sense regarding the overlying group.
- Move the "back" button the left side of the view.

What did work as intended?

- Likes that one, in an easy manner, can change between main views, such as heatmap and sizing view. Mainly easy to distinguish between the views.
- Interacting with a touch screen was mainly easy, in addition to using the touch pad.
- Thinks the inclusive or just touched the area button in the area view makes sense.
- Easy to distinguish between the different trips, and the usage of colours. Same for the sheep and predator observations – understands the use of colours here.

- Likes that one can edit anything and the usage of the filters. Feedback is nice.
- Good navigation between views – fast and simple.
- Tidy and clean design of the application.
- The usage of maps, and visualisation on that worked well and made sense. As well as the showing of trips and usage of filters. Most icons on the overview page made sense and worked as intended.
- Use the grouping overview.

C.5. Participant Five

Tasks

The subjects starts with getting a bit familiar with the application. Tries out the routing, and some of the different predefined cases. When in the analysis view, checking out the different main views, the statistics and the filters.

You just have been on a supervision trip. Find out how many sheep that were observed in this trip.

Understands that this is a trip which is conducted today. Enters the "last week" in the time based grouping, and enters the analysis view. Goes to the trip list in the side view and filtrates away the other trip shown. Goes back to the statistics tab and sees how many sheep which have been observed in this trip. Notices that the trip shown is from today.

The grazing season have ended and all sheep have been collected. You want to see where there have been observed most sheep and the density of these

Thinks that this can be solved with a time based case, and selects the "density in 2019" case. Is a bit insecure if another predefined case is to be used. Sees the heatmap view but is insecure if this is solving the problem. As the heatmap only give an indication on where there have been observed most sheep, it doesn't give any numbers. Changes to the sizing view and thinks on what area name which is the answer to the question. Still thinks the heatmap will give the correct area, but misses a number for the density. Shows the area, using heatmap, to show where there have been observed most sheep.

You know there are missing some sheep from a given area. Find out if there have been observed any predators there

Stays in the same view as last task. Try to use the graph to get the information wanted. Changes from heatmap to regular view, but doesn't see any predators, only sheep. Goes

to the filters and removes the sheep observations, and adds the predator observations. Does not add the external predator data from NINA, and therefore not sees this. Does not go back to the overview page to select the "predator" case under area based views.

In the area around Hyttdalen, there have been observed 34 sheep during this year, but you cannot remember how many which were observed last year. Find this number

Goes back to the overview. A bit insecure what case to use, what case to use. Tries multiple cases, such as the area based "observation" and "trips". Thinks it is difficult to search in the wanted area, as whole of Norway is showing, not just the area where there are registered trips. Zooms far into the map, and gets less information as wanted. Struggles to find the area around Hyttdalen. Did not set the date this time, thought that it was forgotten. Only gets one trip, as the area was too specific. Tries again, misses that the area is not "remembered" between cases. Does zoom too far in again and gets the same trip up as last time. Goes back to the overview, and tries again with "trip" in area based again. Sets the correct dates, and zooms into the correct area. Finds the data using the graph. Does not use the comparison case.

On trip "Tur tre", there wasn't observed any sheep. You want to walk this trip once again to see if there are some sheep there this time. Find the route

Assumes that this trip is in the last month, therefore use the "last month" view in the time based grouping. Filters away all other trips than the wanted one, but the map isn't showing anything. Because trip route is not turned on, nothing is showing. Goes back to the overview and tries the "trips of the year" under the comparison grouping. As the time main view is turned on, one cannot filter away trips, does not think of this. Tries to use the graph, but as no observations are done in this trip, nothing is showing in that column. Could maybe have been easier to understand if the name have been set on that axis. Understands this and sets another main view, filters away other trips and turns on the trip route. Then the correct information is shown. A list of the trips would be nice to solve this, the subject mentions. Does not use the custom setup, which could have solved this easier.

The government wants information on the data collected last year. Show only data from 2018.

Enters the "observation" view under area based grouping. Types the dates first. Still troubles with selecting the area to choose from. Types the date, does not use the date picker, as it takes to much time to find the correct date using that. Uses the touch screen for zooming the map. Sets the wrong ending date, in the future, so all trips will be selected. As all trips are shown, too much information is showing on the map the subject thinks – nine trips in all. Puts on the grouping main view to get the map less complex. Did not find the correct number as all trips were selected. Due too the date chosen not showing on the analysis field, this information were lost.

You want to see the trips done in this year and see the data from them stepped

Goes back to the overview and enters "trips of the year" under comparison view. Would like to filter out all trips and then go through each trip manually with the trip filter. Does not see the slider one can use for this, so the subject thinks correct, but did not see the slider. As some of the trips have similar colours as the grazing fields on the map, this gets a bit confusing. Try not to select colours for the trips which are similar to the ones used by other components in the application.

General Notes

Preferred to use the filters instead of the predefined ones, but with another layout with less navigation between tabs. Used both the touch pad and the touch screen for both navigating and interaction. Did not think about that the predefined cases mostly differed by different filters. Felt that too much was predefined, wanted more power over the application. The system was a bit complex some times, would like a bit more overview of the application.

Interview

This interview were more a discussion on all the aspects which the questions asked. It became more like a conversation. Aspects which were noted during the interview is therefore filled directly into the next section.

Key Notes from the Test

What should be edited?

- There should be a number indicating what value the heatmap is showing at the current view.
- Maybe put in a number of the density of sheep per square kilometers in a given area? Which the heatmap already is indicating.
- Too much navigating between the different tabs in the side view. Most of these elements could have fit the same view.
- Want's a list for all trips in the overview, where one can filter out what trips one wants, and then set the filters interesting.
- When searching an area, the opportunity to search for a place name, and then zoom into the map with that information.
- In the area view, one could show all trips with their route, and group the together with the grouping function when one is to search for trips in a given area.

- Would like a tab view where one can select trips, and what kind of information one wants on the map, like a guided tour or step like approach.
- The legend and help button have to be more visible, did not find it during this test.
- More consistent use of colours, and not use overlapping colours for different elements of the application.
- Too many choices which can be done in the application.
- Wants a more interactive map, where one, for example, can click on a trip on the map, and information of that trip would be highlighted other places in the view. For example in the trip list, timeline and in the graph. In other terms, mark a trip on the map.
- Cumbersome to switch between the tabs all the time. Things which is to show information to the user, should be visible all the time, such as the graph. The actions, should be placed in one page so one didn't need to switch between so much. As one have to navigate between the views, it can be difficult to remember what action to take.
- The stepper in the time view looks like a zooming tool, should be more clear what the functionality of this one is.
- Area based cases have some problems as there are too much work to be done when one enters this area.
- The system should remember which map is chosen between views.
- The manuel input of dates was quite difficult and did not know what format to use.
- When something is loading from the api, it should have a larger indication that something is loading.
- The custom setup page should have been called something else, and should have been more shown.
- The colour chooser was too detailed, should be more simple, both the process and the chooser.
- A bit more use of colours in the design to make the application more interesting.
- The predefined cases was too wide, wanted more key functionality with those cases, and maybe have less of those.

What did work as intended?

- Filtrating of the trips worked as intended.
- Did use the main views and the filters active to reduce the complexness of the map.
- The system was very interactive and responsive. The action elements gave good feedback and was well linked to the map.
- The overview page was quite lucid. The grouping based on time was liked the best as one got into the analysis view without anymore interaction.
- Clean and simple design.
- The integration with the trips, map and graph worked well.
- The automatic zoom when one filtered trips was nice. As well as when clicking the trips and filtering out the trips, the map automatically zoomed into the correct area.
- Did not use the help page, but did not miss it as much. In other terms, the most of the signs gave sense.
- Thinks the system would have worked nice on a touch device like a touch pad.

C.6. Participant Six

This subject was not given the main tasks, as we wanted to see how well the participant used the system in combination with the predefined cases. The user have some kind of knowledge about sheep.

General Notes and Overview Tool

Starts the test by entering the "trips in the last week". Starts by looking at the map and tries to zoom onto the map. Is a bit unfamiliar with the touch pad on the computer, and therefore struggles a bit with the zoom. Decides to use the zooming buttons on the map instead. Registers that the participant tries to press the trip on the map to get some more information about it, but nothing happens. When zooming far out, struggles a bit to zoom in again. Does not use the auto zoom button into the area.
Looks at the observations and tries to understand their meaning. Presses the observation and sees that it pops up that there are seen sheep and a given number. Does understand that the white ones are sheep and the gray ones are predators. Would like the system to show lambs observed as well – this is actually implemented, but the mock data does not contain any observations with lambs in them.
Enters the tab view with the trip list and registers that there are two trips in this view.

Registered that this was the case in both the graph and the map. Nice that one could see the date here, but would like this to be shown other places as well. Such as on the graph, the time span should have been shown, or at least what trip is where.
No tries to change the main views and see what they mean. Presses the sizing main view, and understands from the map that points with more sheep, are larger. Likes this one. Switches map type to terrain, and likes that much. Sees more detailed information about the area, such as contour lines and other interesting aspects of it. Likes this map type more than the gray scale one. Tries the time main view and sees that one of the trips gets red. Reads the header of the view, which indicates that this is trips from the last week. Interprets that there have been two trips in the last seven days. Looks at where the trips have went, in regards too where the route was and where observations were made. Would like to see how long the distance walked was. Feels finished with the last weeks trips and goes back to the overview.
Now tries the next one in line, "trips conducted in the last month". Edits the map to use the terrain layer once again. Assumes that the gray observations are predators, and presses them and sees that they are predators. Haven't been in the filter tab yet. Comments yet again that there are no dates on the statistic graph. Unlike the last case visited, the subject notices the timeline at the top of the analysis view. Tries to press the dots on it, but misses, and therefore no auto zoom to that trip is showing. Notices that there are same colour on the map and the timeline and that these colours are indicating the same kind of information. Assumes that as this is the last month shown, then there are showing all trips from the last four weeks. Looks at the density of sheep in the most dense area with observations.
Sees that there only are represented three trips in the graph, but doesn't understand why there are four trips on the map and three here. As trip three doesn't have any observations, this should have been indicated. Some of the trips have too similar colours, the system should fix this automatically. Likes what is seen! Easy to see what areas where there are sheep. Enters the main view and tests out the heatmap view. Understands what the meaning of it are, but misses the ability to see what the colours of the heatmap are indicating – an actual number. Exits to the overview.
Tries the last predefined case in the time grouping. Zooms into the high density areas, as the heatmap are showing. As the timeline looks a bit different from the last view, thinks that this is indicating the colours of the heatmap, which it doesn't. Does like this view and what information it shows. Understands that this is for all trips conducted the last year. Exits to the overview.
Decides to try out the area based cases and enters the trip mode in area based cases. Enters the area deciding view, and gets a bit overwhelmed that the whole of Norway is showing. Zooms into the correct area but expects to see the trips in this view. Presses the further button and comes into the expected view. Did not use the date fields. Looks at the timeline once again, and tries to map the trips shown to the points on the timeline. Does the same with the graph and the map and sees that there are missing one trip in the graph. Finds out gradually that there are no observations done on that trip. Is impressed with what the system can show.
Enters the main views and presses heatmap. As indicated before, "the more observa-

tions, the hotter the view". Likes this view. Tries the time view once again. Likes that one can go through all the trips with the bar and that the selected trip gets another colour than the others.
Enters the overview again and wants to try the predator view, and enters the area choosing view. Wants to check another thing in the overview and navigates back with the browser back button. This is sending the participant to the analysis view which is not what expected. Should have been sent to the overview. Presses back and enters the overview. Enters the predator in area based once again. Tries to go further without zooming, which is not working as expected. Finds the given area, still struggling a bit with the zooming. Does not understand why the system not auto zooms to the area where there have been observed sheep. Enters the view, but the predator data is loading quite slowly. Tries to map the colours on the statistic graph with the data on the map, but the colours have no interaction with each other as the colours in the graph are the same for all. Should only show predator observations in the graph for this view. Does not understand the difference between the external predator observations and the external ones, does not conclude with the difference.
"If I was a sheep farmer, I would have been very satisfied with this system." – the test participant. Interprets the predator statistic data, does not understand the difference between the points. Does not use the helping screen for this session. Wants the ability to use the heatmap on the external data as well as the observations done manually on trips. Wants the ability to filter away trips with the timeline, and/or the timeline to show the dates on it. Will be easier to see when the trips are conducted.
Enters the overview and wants to try one of the comparison cases. Enters the comparison for all data between this year and the last one. Understands that each colour is indicating which year they are from, but misses the ability to distinguish between the trips inside a year. They could for example have a different shade of the "year colour".

Key Notes from the Test

What should be edited?

- Get some information about a trip when pressing the trip route, such as number of observations, length of the walking trip. Maybe also mark in the graph which trip is showing. Or just when hovering the trip, some information should show.
- Have mock data which shows lambs and other characteristics as well.
- The graph with statistics doesn't show any information in the x-axis, should show the name of the trip, or their date.
- The colours in the graph is not matching the trip colours used elsewhere in the application.
- The auto zoom button should have another icon which is indicating that this is a auto zoom button. It is not clear as of now.

- Not directly clear that each predefined case uses the same view, but only with different filters.
- Preferred the terrain map, and changed to this often. The system should remember which map type is preferred throughout the application.
- Show the distance of the walked trip.
- Did not use the filter tab at all – maybe this one is unnecessary as the subject got the information wanted regardless of this?
- As no observations were seen on "Tur tre", nothing is showing in the graph. Should still indicate that there is a trip regardless of this on the graph.
- Use more distinctive colours for the trips as some of them can become too similar.
- Give a number of what the colours in the heatmap are indicating.
- The "large" timeline can be a bit diffusing, as it could indicate the density of the heatmap.
- The area picking view should only show the area where there have been observed sheep. This mode is a bit tricky to understand.
- When pressing the back button in the area view, one gets sent back to the analysis view and not the overview. The reason for this is that there are no actual routing happening in the area view, only change of stage variables in the application.
- The ability to apply the main views for the external predator data as well. Find a more clever way of visualise the external and the self-observed predator observations.
- The external predator data is loading quite slowly, should either have a better indication that there are loading data or/and load the data with a faster routine.
- Should only show predator observations in the graph when in predator mode.
- The help section needs to be more visible.
- Get a better view of when the trips are conducted in comparison to each other.
- The opportunity to distinguish between the trips conducted in the same year in comparison mode.

What did work as intended?

- Nice to have multiple ways to zoom on the map, such as the touch pad and the zooming buttons.
- The colour usage of the observations makes sense – gray for predators and white for sheep.
- The time based views makes sense and contains interesting information – which is the case for all predefined cases in this grouping.
- That the same colours are used for the different trips in more views are working as intended. For example that the dots on the timeline, points on the map and the trip colours are the same. Works as intended.
- Easy to find areas where there have been observed sheep.
- Regardless if there are missing a number on the heatmap, easy to understand that the heatmap are indicating density areas.
- The information the system can show is much, and often done in a nice manner and not too complex. Especially when one uses the predefined cases.
- The time view have some interesting features.
- That there are external and real statistics with predator observations in the area.
- "If I was a sheep farmer, I would have been very satisfied with this system."
- The information shown in the system is quite usable, and thinks it is easy to follow what the system is made for and lays up to.
- Likes the area based and the time groupings best.

C.7. Questionnaire

Here is the raw data from the questionnaire written down with the raw results from the attendants. All answers will be given under each question with the average score if relevant. The individual answers have not been noted here, but the total for all have been included.

C.7.1. SUS Questions

I found the system unnecessarily complex (1-5) – where one strongly agrees and five strongly disagrees. Total score is **25**. The average score is **4.2**.
(i) **2** (ii) **5** (iii) **4** (iv) **4** (v) **5** (vi) **5**
The system was easy to use (1-5) – where one strongly disagrees and five strongly

agrees. Total score is **22**. The average score is **3.7**.
(i) **2** (ii) **4** (iii) **4** (iv) **3** (v) **4** (vi) **5**
I would have needed user support to use the system (1-5) – where one strongly agrees and five strongly disagrees. Total score is **23**. The average score is **3.8**.
(i) **3** (ii) **5** (iii) **4** (iv) **2** (v) **4** (vi) **5**
The different features in the system were well integrated (1-5) – where one strongly disagrees and five strongly agrees. Total score is **25**. The average score is **4.2**.
(i) **3** (ii) **4** (iii) **5** (iv) **4** (v) **4** (vi) **5**
I thought there was too much inconsistency in this system (1-5) – where one strongly agrees and five strongly disagrees. Total score is **24**. The average score is **4**.
(i) **3** (ii) **5** (iii) **5** (iv) **3** (v) **4** (vi) **4**
I think the system is easy to learn fast (1-5) – where one strongly disagrees and five strongly agrees. Total score is **22**. The average score is **3.7**.
(i) **2** (ii) **3** (iii) **4** (iv) **4** (v) **5** (vi) **4**
The system was cumbersome to use (1-5) – where one strongly agrees and five strongly disagrees. Total score is **21**. The average score is **3.5**.
(i) **2** (ii) **4** (iii) **3** (iv) **3** (v) **4** (vi) **5**
I felt very confident using the system (1-5) – where one strongly disagrees and five strongly agrees. Total score is **23**. The average score is **3.8**.
(i) **2** (ii) **5** (iii) **4** (iv) **4** (v) **3** (vi) **5**

C.7.2. Functionality

This section of the questionnaire contains questions which asks how well the functionality worked in the application.
For the display of the observations, which of the main views worked and made sense?

- *Regular view* – six out of six votes
- *Sizing view* – four out of six votes
- *Heatmap view* – six out of six votes
- *Grouping view* – five out of six votes
- *Time view* – three out of six votes

How well did the filters work? (1-5) – where one strongly disagrees and five strongly agrees.
(i) **3** (ii) **5** (iii) **4** (iv) **5** (v) **5** (vi) **3** – this one is ignored as the participant did not use the filters during the test.
Did you use the filters actively to prevent the information on the map being too complicated?

- *No* – one vote

- *Yes* – four votes
- *Did not know their function* – one vote

It was easy to see how many sheep there were observed in each trip (1-5) – where one strongly disagrees and five strongly agrees.
(i) **2** (ii) **5** (iii) **5** (iv) **5** (v) **3** (vi) **5**
There were easy to see where the observations of sheep were on the map (1-5) – where one strongly disagrees and five strongly agrees.
(i) **4** (ii) **5** (iii) **2** (iv) **5** (v) **5** (vi) **5**
It was easy to distinguish between predator and sheep observations (1-5) – where one strongly disagrees and five strongly agrees.
(i) **4** (ii) **5** (iii) **5** (iv) **4** (v) **5** (vi) **5**
It was easy to distinguish between the different trips (1-5) – where one strongly disagrees and five strongly agrees.
(i) **3** (ii) **5** (iii) **4** (iv) **5** (v) **4** (vi) **5**
Did you use the help section?

- *Yes* – one of six votes.
- *No* – three of six votes.
- *Did not know it existed* – two of six votes.

How well did the system work as an overview tool for sheep? (1-5) – where one strongly disagrees and five strongly agrees.
(i) **3** (ii) **5** (iii) **5** (iv) **5** (v) **3** (vi) **5**
How well did the system work as an overview tool for predators? (1-5) – where one strongly disagrees and five strongly agrees.
(i) **3** (ii) **5** (iii) **5** (iv) **5** (v) **4** (vi) **5**
How was the amount of functionality in the system?

- *Too much* – two of six votes.
- *Just right amount* – four of six votes.
- *Too little* – zero of six votes.

Other comments on the functionality (optional):

- There were a lot of ways one could see the same data, would have been better if it were more focused, generally. Did like the predefined views, so one didn't need to relate to all of the alternatives. Some of the alternatives were too detailed, as they did not really matter so much.
- It was just the right amount of functionality, but some of the groupings could have been more intuitive as some of the predefined views were too equal, for example the time based trips and observations.

C.7.3. Predefined Cases

This section will cover the predefined views and their functionality. How well integrated they were and other relevant data for these.
How well did the time based views work? (1-5) – where one strongly disagrees and five strongly agrees. The average of the time based is **4.2**.
(i) **4** (ii) **4** (iii) **5** (iv) **5** (v) **3** (vi) **4**
How well did the area based views work? (1-5) – where one strongly disagrees and five strongly agrees. The average of the area based is **4**.
(i) **2** (ii) **5** (iii) **5** (iv) **4** (v) **3** (vi) **5**
How well did the comparison based views work? (1-5) – where one strongly disagrees and five strongly agrees. The average of the comparison based is **3.8**.
(i) **2** (ii) **5** (iii) **4** (iv) **4** (v) **3** (vi) **5**
Would you have preferred too use the predefined views or set the filters manually?

- *Predefined* – two of six votes.
- *Manually set* – one of six votes.
- *Both* – three of six votes.

Would you have liked to save custom setups as a filter?

- *Yes* – four of six votes.
- *No* – one of six votes.
- *Maybe* – one of six votes.

C.7.4. Design

The focus on design is also important in the application, therefore there were added a design section.
Did the icons used work well? (1-5) – where one strongly disagrees and five strongly agrees.
(i) **2** (ii) **5** (iii) **5** (iv) **5** (v) **3** (vi) **5**
Was the side view ever in the way, or was the map at any time too small?

- *Yes* – zero of six votes.
- *No* – five of six votes.
- *Maybe* – one of six votes.

Was the information on the map too complicated at times?

- *Yes* – three of six votes.

- *No* – three of six votes.

If yes, what was the reason for this?

- *There were too much information on the map* – two votes.
- *Too many different colours* – two votes.
- *Too equal colours* – one vote.
- *Too small signs and icons* – one vote.
- *Too many overlapping elements* – three votes.

How well did the system give feedback when using interactive elements? Such as usage of buttons, filters, data loading etc. (1-5) – where one strongly disagrees and five strongly agrees.
(i) **4** (ii) **5** (iii) **3** (iv) **5** (v) **4** (vi) **4**
How consistent was the design throughout the application? (1-5) – where one strongly disagrees and five strongly agrees.
(i) **3** (ii) **4** (iii) **5** (iv) **4** (v) **5** (vi) **5**
Something else you want to comment? (optional)

- Too much information that is "hidden", which needs hovering too show. More information should have beed showed by default without hovering.
- It would have been nice if the map had zoomed onto the area where the data existed, in the area based views.

C.7.5. Summary

For each of the main views, the two ones which worked best were heatmap and regular view. Some of the participants mentioned that it was a bit pedantry to differ between the regular and sizing view, as well as the grouping view should be shown at all times. The heatmap view worked well and as intended. The time view had some bugs containing it, as well as the switching bar was too tricky to overlook and was not interacted with. Four out of six participants used the filters actively to get a less complicated map. This was the intended use for the filters, but as some of they mentioned during their tests, there should be less of them, and maybe more of them should make more sense.
For the observations on the map, both for predators and sheep, this worked as intended. The participants understood what the points meant, mainly, and they thought it was easy to see where there have been observed sheep. As it was quite simple to distinguish between the predator and sheep observations, none of the participants mixed those two. For the trips, this was seen to be the same, most of the participants did see the difference between the different trips, mainly because of the colour aspect and the different areas they were conducted in.
For the help section, only one participant used it. During the testing and interview

session, it became clear that it was too hidden in the system. As well as the name "help" wasn't too obvious for the participants. They would more like that the legend were visible at all times, instead of hidden away.
Most of the participants did think that this system could work as an overview tool for both predators and sheep. As one both could see where the sheep have went, as well as see their trip route and see areas with points of interest. The participants mainly agrees that there were the just right amount of functionality, with some thought it were too much. As some of them as well thought the map could be too complicated at times, it could be recommended to strip down some of the functionality and instead combine some of them.
From the predefined views, they worked mainly as intended, based on the questionnaire. The one that worked best were the time based views, but some of the participants would like to set the dates for them selves. The comparison did get the lowest score on average, and we think the reason for this is that some of the predefined views did not make too much sense. The same with the area based, they worked, but how one could find the area for the sheep, was too difficult. To start with the whole map of Norway, made it a bit complicated. Some of them mentioned as well that the time based did it so well because it was only one click away from the analysis view. The area based had a problem that it was two steps which one had to set. The one comparison that made most sense was the one that compared all data from this year and to the last. One step and straight to the analysis view.
Mainly, the participants would like to have the option to use the predefined views, but also have the option too change some of the information manually. As well as the opportunity to save filters which one could reuse later.
The use of icons were mainly done intuitively, in other words, what they symbolised, worked as intended. There were some flaws, as the "last trip conducted" did not make as much sense, but for trips, area based and predators, these icons symbolised what was wanted. One thing that should be edited is that the information on the map had a tendency to get too complicated. For example if much information were shown at all times, or too many trips. One thing which could have solved this could be that grouping should be turned on at all times, as well as the trip grouping if the zoom was too low. The main concern for the complexity of the map, was that there were too many overlapping elements at times, for example with overlapping trip routes and observations.
Mainly, the usage of interactive elements was quite nice. The system was interactive and the buttons worked well quite fast. For example when pressing the filters and main views, these changes was applied on the map directly without loading, which the participants liked.

Jørgen Aunet Halvorsen

Sheep Shepherding - Grand Overview

NTNU
Norwegian University of
Science and Technology