

Egil Uggerud

Sheep-and-Go

Creation and evaluation of a mobile application
with regard to observation of sheep

Masteroppgave i Informatikk

Veileder: Svein-Olaf Hvasshovd, Nina Langhelle

Juni 2020



Sheep-and-Go

Creation and evaluation of a mobile application with
regard to observation of sheep



Egil Uggerud

Master thesis, 2020

Acknowledgements

I would like to thank my supervisors, Svein-Olaf Hvasshovd and Nina Langhelle for the advice, supervision, and guidance throughout this project. I appreciate all the encouragement I have received, and feel lucky to have received the opportunity to be your student for this period of time. I would also like to thank Halvor Mjøen for part taking in informative meetings, and for sharing ideas and providing insight to the world of tracking technology.

Abstract

There is an ongoing globally trend of children spending their leisure time in front of their computers in the comfort of their home. At the same time sheep farmers profit less than ever before, and the once popular and traditional profession is now in most cases done as a hobby. This thesis aims to incentivise children and their parents to spend their time outside, by conducting fun observing activities. The byproduct of these activities will be sheep location data, which is beneficial for sheep farmers, and enables them to save resources or animal tracking.

Sammendrag

Det har i dag blitt en vane at barn bruker fritiden sin foran datamaskinen og TV-skjermen, i komfort av sine egne hjem. Samtidig lever vi i en tid der norske sauebønder profiterer mindre enn de noen gang har gjort før, og det som en gang var et populært og tradisjonelt yrke, blir nå hovedsakelig gjennomført som en fritidsaktivitet. Denne oppgaven har som mål å oppmuntre barn og deres foreldre til å bruke mer tid ute i naturen, ved å gjennomføre morsomme observasjonsaktiviteter. Biproduktet av aktivitetene vil være lokasjonsdata på sauer. Dette er informasjon som er verdifull for sauebønder, og vil som resultat føre til at de kan spare dyrebare resurser på sporing.

Contents

1	Introduction	5
1.1	Project Charter	5
1.2	Motivation	6
1.3	Goals	7
1.4	Terms and Abbreviations	8
2	Related Work	9
2.1	FindMy	9
2.2	ChoreMonster	10
2.3	Geocache	10
2.4	Pokémon GO	11
2.5	Relations to Sheep-and-Go	12
3	Project Scope	13
3.1	Product Statement	13
3.2	Main Components	14
3.3	Development Methodologies	14
3.4	Milestones	16
4	Requirements	18
4.1	Functional Requirements	18
4.2	Non-functional Requirements	20
4.2.1	Usability	20
4.2.2	Security	20
4.2.3	Reliability	20
4.2.4	Scalability	20
4.2.5	Performance	20
4.3	Use Cases	21

5	System Architecture	25
5.1	Project Management Tools	25
5.1.1	Trello	25
5.1.2	Overleaf	25
5.1.3	Github	26
5.2	System Development Tools	26
5.2.1	Git	26
5.2.2	React Native	26
5.2.3	Firebase	27
5.3	Discarded Tools	28
5.3.1	Google Docs	28
5.3.2	Unity	28
5.4	User Interface	28
5.5	Controls	29
6	Design Implementation	31
6.1	Sheep-and-Go Concept	31
6.1.1	Register and Login Screens	33
6.1.2	Home	35
6.1.3	Observations	38
6.1.4	Prizes	39
6.1.5	Navigation Footer	41
6.2	Technology Rationale	41
6.2.1	Platform	41
6.2.2	Operating System Popularity	41
6.2.3	Component Compatibility	42
6.2.4	Developer Knowledge and Practicality	42
6.2.5	Rationale Conclusion	42
7	Software Architecture	44
7.1	Kruchten's 4+1 Architecture	44
7.1.1	Logical View	45
7.1.2	Process view	47
7.1.3	Physical View	51
7.1.4	Development View	53
7.1.5	Scenarios	56

8	User testing	57
8.1	Test Subject	57
8.2	Paper Prototype Testing	57
8.2.1	Prototype Results	62
8.3	Planned User Testing	62
8.3.1	Hike	63
8.3.2	Point Deducting	63
8.3.3	Questionnaire	63
8.4	Force Majeure and Adjustments	63
8.4.1	Impact of COVID-19	64
8.4.2	Adjustments and Post Adjustment Testing	64
8.4.3	Wizard of Oz	65
8.5	Questionnaire	66
8.5.1	Validity and Reliability	66
8.6	Results	67
8.6.1	Questionnaire Segment One	67
8.6.2	Questionnaire Segment Two	68
9	Future development	76
9.1	Future Development	76
9.1.1	Application Improvements	77
9.1.2	General Improvements	77
10	Evaluation	79
10.1	Product Evaluation	79
10.2	Project Evaluation	80
11	Conclusion	82
11.1	Conclusion	82
11.1.1	Satisfaction of Goals	82
11.1.2	Summary	84
	Appendices	92
A	Invitation letters for user testing	93
B	Questionnaire	95

Chapter 1

Introduction

In this chapter, the reader will be introduced to the motivation behind the project as well as its goals.

1.1 Project Charter

Project Charter	
Project name:	Sheep-and-Go, Spring 2020 Master thesis
Start date:	29.08.2019
Due date:	02.06.2020
Client:	NTNU
Developer:	Egil Uggerud
Point of Contact:	Svein-Olaf Hvasshovd & Nina Langhelle

Table 1.1: Project Charter

1.2 Motivation

According to an article[1] by The Norwegian Institute of Bioeconomy Research (NIBIO)[2], 95% of Norway's terrestrial area is range land and around half of this is considered applicable for grazing of livestock. With only 5% of the terrestrial area consisting of buildings, combined with a national tradition of farming, it is normal for farms to own large amounts of landmass. It is also worth to mention that it is tradition for farmers to keep their sheep at a distant location to their farm, as the grass closest to the farm usually is cut to become fodder for the sheep to eat during the off-season. As a consequence of the previously mentioned facts, it is not unusual for Norwegian flocks of sheep to have access to very large areas to graze in freely. Considering the large grazing areas a flock of sheep has to utilize, it is important for the farmers to keep track of their sheep during the grazing season, in order to reduce time and resource usage during the animal collection at the end of the grazing season. Considering that 70% of Norwegian sheep farmers don't profit on their sheep herding[3], and many farmers breed sheep solely as a hobby, the importance of unnecessary use of resources, increases. Until the 1950s[4] children and youth helped with tracking and collecting the grazing animals, but in the modern world, they rather be home connected to their monitors. According to a study conducted by the United Kingdoms Government, one third of British children are less outside than prison inmates [5]. The motivation behind this project is to reduce the costs of tracking sheep during grazing season as well as providing incentive for children to spend more time outside in the Norwegian nature.

1.3 Goals

This section shows the system goals of the master thesis, and should be used as a success measurement of the developed system.

- Goal 1. An incentive for children to be more outside.
- Goal 2. Appeal to both children and their parents.
- Goal 3. Enable observation of sheep to be registered.
- Goal 4. Help to reduce resources used by the farmer with collecting sheep.
- Goal 5. Help the farmer spot illness and strange behaviour in sheep.

1.4 Terms and Abbreviations

- Geo fence: A virtual perimeter for a real world geographic area
- App: Application
- Graze: To feed on growing herbage
- Modal: Dynamic pop-up window
- Geo location: Identification or estimation of the real real world geographical location of an object
- SDK: Software development kit
- AR: Augmented Reality
- GUI: Graphical user interface
- API: Application Programming interface
- Emulator: Software that enables one computer system to behave like another computer system
- Force Majeure: Extraordinary event beyond the control of any parties
- Test Subject: Person testing a product
- Booting: The process of starting a computer or a computer system
- UML: Unified Modelling language for modelling computer systems
- Operating System: Low-level software that supports the basic functions of a computer

Chapter 2

Related Work

This chapter describes works that in different ways are related to Sheep-and-Go, as well as how they are related. The works are presented in separate sub chapters which includes a description of the applications, as well as their history. These examples were specifically chosen after an extensive literature search on applications and technology available in the categories of observation and children games. Works which included functionality that matches the project description or could aid the development of the project, were deemed as relatable work to Sheep-and-Go. The following four sub-chapters describe the most notable ones.

2.1 FindMy

FindMy[6] is a company founded by Norwegian sheep farmers Marit Mjøen and Arnstein Solheim. The motivation behind the company was based on the farmers own experience with the struggle of keeping track of their sheep during grazing season. After a season where they counted a 23% animal loss [7], the two started looking for options to better track their herd. They realized that every solution on the market was based on cellular communication and therefore would not work in the Norwegian mountains where the farms are based, as a consequence of poor cellular reception. As a result, Marit and Arnstein developed a product called "E-bjelle"[8]. A light weight tracking device based on satellite communication instead of cellular communication. With their market advantage of not relying on cellular services, they were accepted by the market, and now have 30 000

active devices across the country. The "E-bjelle" product comes with a phone application with a map display that allows the customer to keep an overview about the position of their animals as well as a timeline of their animals movements. Services provided by the application currently includes the customers ability to create Geo-fences that notifies the animal owners when their animals cross the line. The animal owners are also notified if an animal is found to stand still as this can be a result of it being stuck or the contraction of a disease. The product is currently being improved with new functionality by Halvor Mjøen, one of the product owners.

2.2 ChoreMonster

ChoreMonster[9] is a set of two applications founded by Chris Bergman and Paul Armstrong, which they initially released in 2011. The set of applications encourages children to complete their chores around the house. One of the applications is for the parents to set chores that need to be completed within a household. The other application is intended for their children, and is where a child register their completed chores. When a parent creates a chore in the parent-application they also choose an amount of fictive points the children will receive if they complete the task in time. These points can be traded in for real life rewards decided by the parents. ChoreMonster provides reward examples such as ice cream, camping trips or an hour of television. When completing a task the children also grants the ability to "spin the wheel" at "Monster Carnival" which rewards them with costumes for their monster avatar inside the application. The motivation behind the application is to turn the whole concept of chores into a fun and convenient process for both children and their parents.

2.3 Geocache

Geocaching[10] is an application created by Groundspeak Inc[11] and initially released in 2013. The application is based on a game from the mid 1800s called Letterboxing[12]. It is a form for treasure hunting, where other application users hide small treasure chests in public, both in cities and in the nature. The Geocaching application is similar, but enables users to

create and hide treasure boxes as well as keeping track of previously found treasure. After hiding a treasure box the user provides text clues that tell other users which area to search and what to look for. When a treasure is found the user registers a code found within the treasure box. A marker is then displayed on the map within the application. This way users are able to track all previous found treasures with useful metadata such as position and time. There are several communities built around this application in order to both create and search for these treasures. Geocaching.com[10] estimates an amount of 3 million active treasures spread across the world. The motivation behind the application is natural and urban exploration, exercising and family activity.

2.4 Pokémon GO

Pokémon GO[13] is a mobile game application that is based on Augmented Reality (AR) technology. It was released in July 2016, by Niantic [14] in collaboration with The Pokémon Company[15]. The game uses the GPS location of a players device and displays a character, mimicking the players real-life location on a map in the application. Any movement and relocations done by a player will thus also move the players character in the game. The goal of Pokémon GO is to locate and capture creatures called Pokémon. These creatures can, after being captured act as guardians of the players character. By battling the Pokémon of other players to gain levels and skills. The interaction between a player and their Pokémon is done using Augmented Reality (AR), an interactive experience caused by enhancing the perception of the real world, by adding computer generated effects. In Pokémon GO a player can interact with their Pokémon creatures through the camera of their playing device. Animations of creatures are placed in the camera picture, simulating a real-life existence.

2.5 Relations to Sheep-and-Go

The works are vastly different from each other, but each of them includes some sort of relation to this project, either technical or usage similarities.

- Findmy: Like Sheep-and-Go, Findmy revolves around locating sheep. The technical similarities include a navigable map with location history displayed.
- ChoreMonster: The aspect of gaining fictive points and trading these in for real life prizes is the relatable to the point system of Sheep-and-Go
- Geocaching: The aspect of navigation towards hot spots displayed on a map is relatable to the game play of Sheep-and-Go
- Pokémon GO: Navigation towards and from certain mapped areas are relatable to the game play of Sheep-and-Go

Chapter 3

Project Scope

This chapter describes the scope of this project, the main components of the application and the development methodologies used.

3.1 Product Statement

The scope of the project is to deliver a mobile application that can be used by children and their guardians to observe and track sheep during grazing season. The product will include functionality to report current sighting of the animals, and include history of previously made observations. The application is meant to gamify the process of observation and will include so called hot spot markers, or incentives to where the children can look for animals. As well as Avoid markers the players should steer clear of. The main difference between Sheep-and-Go and its competitors like FindMy is that the animal tracking done in Sheep-and-Go is solely based on human observations with no use of GPS data, which is a popular choice for developers of similar products. The data produced when using the application will fill the need of a sheep farmer to have knowledge of the positions and state of their sheep.

3.2 Main Components

- Feature for registering observation of sheep to obtain fictional points
- Feature for the ability to upload comments regarding the situation
- Map view containing hot spot markers, avoid markers and icons for previous made observations
- Observation view with history and metadata of all observations previously made by the user
- Prize view with functionality to exchange points for real life prizes
- Prize history view for all previous point/prize exchanges

3.3 Development Methodologies

The project was developed in an agile way by adopting principles from the popular development frameworks, Scrum[16], Kanban[17], and extremeProgramming[18]. The key ideas of the three frameworks is to manage complex work, such as software development in an optimal way considering time and resource usage. The reason behind mixing some principles from the three previously mentioned, rather than completely adopting one is mainly because the three frameworks are aimed at team development. Since Sheep-and-Go is a project developed single handed, some of the principles does not directly translate. For instance principles such as "stand up meetings" and "team roles" found in Scrum, were not applicable in this project. The following principles were followed during the development phase.

- Sprints: In scrum the workload is split into different time-fixed iterations known as sprints. The developer in addition to the stakeholders schedule the functional requirements into different sprints, based on their importance. After each sprint the system is tested and necessary pivots are done before the next one. By splitting the workload into iterations, every stakeholder becomes up-to-date with the development at all times. With continuous communication between the stakeholders, any differences in opinion will be disputed

early, thus mitigating the chance for developing unsatisfactory software.

- Workflow: The agile framework techniques of Scrum and Kanban let the development begin early, by discussing and reviewing the state of development after each sprint. Other frameworks like Waterfall[19] does not begin development until the planning phase is completely done. The workflow of a project using Scrum and Kanban, will include planning, developing, and reviewing, in all stages of the process.
- Sprint Backlog: Before each sprint a backlog of tasks is made. This list of tasks are added in an order of priority to help the developer to focus on the tasks of priority and not tunnel vision on less critical parts of the process.
- Retrospective meetings: Since the development was done single handed, it was difficult to include retrospective meetings, however a review session was held after each sprint, evaluating the work that had been done.
- Kanban Board is a technique used to describe the current state of the components of a system being developed. Each component is illustrated as a block which is displayed on a board consisting of columns. Each column represent the state of completion. The use of a Kanban board creates a structured overview of all components and the state of their completion.

3.4 Milestones

This section describes the milestones of the Masters Project in the order of their occurrence. This schedule was created and shared with the stakeholders in order for every party involved to be involved in the current status of the project. Each milestone are described below, and the entire timeline is illustrated in Figure 8.4.

- Requirement Specification: This milestone includes defining the functional and non-functional requirements of the system.
- Literature Search: This milestone includes a literature search of sheep herding in Norway as well as related work.
- Sprint 1: Setting up the environment and creating paper prototypes.
- Sprint 2: Testing prototypes and beginning the programming of the main system components. This sprint includes continuous bug fixing.
- Sprint 3: Programming main components and continuing work remaining from the last sprint. This sprint includes continuous bug fixing.
- Sprint 4: Programming main components and continuing work remaining from the last sprint.
- Sprint 5: Sprint 4: Programming main components and continuing work remaining from the last sprint.
- Sprint 5: Making adjustments and finishing the application.
- User tests: Testing the system to gather feedback.
- Writing of thesis.

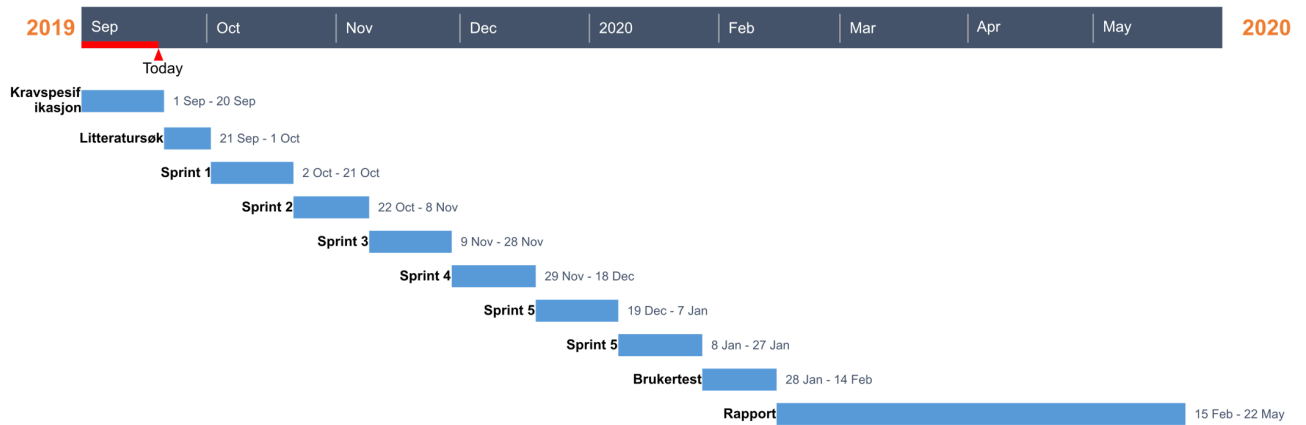


Figure 3.1: Gantt chart that shows the general outline of the project.

Chapter 4

Requirements

This chapter contains a description of requirements that the system must perform after. The functional requirements describes the functions that the system must perform while the non-functional requirements describes the quality attribute of a system. The chapter also contains use-cases which are examples describing certain situations of a user interacting with the system.

4.1 Functional Requirements

- An adult user of the system shall be able to:
 - Create a new account
 - Log in to the system
 - See previously made observations
 - Navigate a map view
 - Deduct fictive game points
 - Register observations of sheep
 - See history of point deductions
- A child user of the system shall be able to:
 - Navigate a map view
 - Register observations of sheep
 - See previously made observations

- See history of point deductions
- An administrator shall be able to receive observation data from all users

4.2 Non-functional Requirements

4.2.1 Usability

The application is created to be used by children accompanied by their parents or guardians, it is therefore important that the application meets properties of good affordance and easily understood user interface. The system should be quick for children to learn and easy to navigate and use.

4.2.2 Security

The system will contain e-mail and password login and will thus handle sensitive data. Any sensitive and personal data should not be available for any outsiders.

4.2.3 Reliability

The system shall behave as intended with no bugs or faults that prohibit the intended use.

4.2.4 Scalability

The system shall be able to handle multiple users at the same time.

4.2.5 Performance

The system is meant to be used outside in country regions in Norway and shall therefore be reliable through poor internet and mobile connections.

4.3 Use Cases

Use cases are steps of actions taken by an actor interacting with a system in order to achieve a certain goal. Each use case is told from the view of a specific actor. It begins with the description of a goal the actor want to reach, and ends with the goal being completed. Preconditions and exceptions included to provide information about what other actions a user will have to complete before accomplishing their goal, as well as possible obstacles met along the way. Uses cases are beneficial in that they explain how functionalities should be completed, common problems that can occur, and provide an overview over the complexity of a system.

Name	Create account
Primary actor	Adult and Sheep farmer
Description	The actor registers a new Sheep-and-Go account
Preconditions	-The actor has launched the system -The actor is located in the registering screen
Basic Path	1. The actor chooses a unique username 2. The actor provides a valid e-mail address 3. The actor chooses a password 4. The actor clicks the "Registrer bruker" button
Alternative Path	-
Exception Flow	-The actor fails to boot the application -The actor provides a username that already exists -The actor provides an invalid e-mail address
Post Condition	-The actor is authenticated and saved to the database -The actor has access to the application

Table 4.1: Create account

Name	Register sheep observation
Primary actor	Child with adult supervision
Description	The actor registers a new sheep observation
Preconditions	-The actor has launched the system -The actor is located in the observation register modal
Basic Path	1. The actor notes amount of sheep observed 2. The actor comments useful information 3. The actor clicks the "Send" button
Alternative Path	-
Exception Flow	-The actor fails to boot the application -The actor fails to understand the functionality
Post Condition	-The observation is registered in the database -The observation information is listed in the observation view -Points are added or subtracted from the total score -The observation register modal closes

Table 4.2: Register observation of sheep

Name	Deducting points (in exchange for prizes)
Primary actor	Adult
Description	The actor deducts score points (in exchange for prizes)
Preconditions	-The actor has launched the system -The actor has passed the password protection -The actor is located in the prize view
Basic Path	1. The actor notes amount of points to deduct 2. The actor comments the prize given to child 3. The actor clicks the "Send" button
Alternative Path	-
Exception Flow	-The actor fails to boot the application -The actor fails to understand the functionality -The actor cannot get past the password protection
Post Condition	-The exchange is registered in the database -The exchange information is listed in the prize view -Points are subtracted from the total score -The observation register modal closes -(The child is given a real life prize)

Table 4.3: Deducting points in exchange for a real life prize

Name	Gather sheep location data
Primary actor	Sheep farmer
Description	The actor gather location data of their sheep
Preconditions	-The actor has launched the system -The actor is located in the map view -There has been registered observations of sheep
Basic Path	1. The actor views the observation icons on the map 2. The actor clicks the icons for detailed information
Alternative Path	-
Exception Flow	-There are no observations made -The actor fails to understand the functionality
Post Condition	-The actor receives location information of their sheep -The actor receives sheep movement history

Table 4.4: Gather sheep location data

Chapter 5

System Architecture

This chapter describes tools used to develop and manage the project, as well as Tools initially used, but later discarded.

5.1 Project Management Tools

5.1.1 Trello

Trello [20] is a web-based planning application that was developed by Fog Creek Software [21] in 2011 and operated by Atlassian [22] from 2017. The software was used from the beginning, through out the entire duration of the project, to act as a Kanban board for organizing userstories during the development phase of Sheep-and-Go.

5.1.2 Overleaf

Overleaf[23] is an online $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ editor that provides version-control, chapter dividing as well as real-time collaboration. Overleaf is a great tool when writing large documents from multiple computers.

5.1.3 Github

Github[24] is a Git5.2.1 version control interface founded i 2008, currently owned by Microsoft[25]. It is a software hosting client, that enables developers to upload and merge their code. Github is widely used in co-operational projects as the merging functionality makes it uncomplicated for developers to combine their code.

5.2 System Development Tools

5.2.1 Git

Git[26] is a version control system that is used to trace changes in a code base. A developer uploads a partial or complete code snippet to a software host using Git. This code is then merged with the previously uploaded code base. A history of each version is saved, which enables the developer to backtrack to previous versions. Throughout the development phase of Sheep-and-Go, Git was used as a version control for all source code, even though Git mainly is used to coordinate the development of multiple people. Its ability to store code history and tracking code changes makes it a great tool to use for solo developers as well.

5.2.2 React Native

React Native[27] is an open source framework for developing mobile applications, Created by Facebook[28]. The framework is a combination of React[29], a Javascript[**javascript**] library for interface development, with native platform capabilities that let the developer use multiple different services of a device. By creating an application using React Native the program can be casted to run on both Android[30] and IOs[31].

5.2.3 Firebase

Firebase[32] is a development platform that provides services for mobile and web-development. This project uses Firebase Auth[32] as email/password authentication, where the user info is stored in Firebase. It also uses Firebase Realtime[32] Database as a back-end service. The use of Firebase enables full user authentication from client-based code. As well as a real-time database with zero cost. In addition to these services, Firebase also lets the developer get some insight in the applications performance. Which can be useful tools to discover weaknesses or bottlenecks in the code.

5.3 Discarded Tools

5.3.1 Google Docs

In the beginning the thesis was written in Google Docs[33] text editor, but was discarded for Overleaf after only two weeks of use. This is because Overleaf has a \LaTeX editor. \LaTeX provides more user functionality structural freedom than Google Docs and therefore seemed more suitable for the writing of the thesis.

5.3.2 Unity

It was discussed to include elements of Augmented Reality (AR) in the project. For enhanced convenience it was chosen to develop the entire project in the real-time 3D development platform Unity[34]. After further planning and discussing the direction of the project, it was decided to postpone the use of Augmented Reality (AR). This also changed the necessity for the use of any 3D development platforms, and therefore to instead develop the application in React Native.

5.4 User Interface

The user interface is designed with a focus of being easy to understand operate for children, the target group, as well as any monitoring adults. In order to achieve this, the interface follows general design principles and guidelines. According to University of Texas (UTHealth)[35]. The central principles are the following: Consistency, visibility, feedback, error messages, documentation, reversible actions.

- Consistency: In order to achieve consistency, elements that are similar in functionality are designed too look similar as well. In the Sheep-and-Go application, the map-view is populated by several hotspot-markers that a sheep farmer either want the players to indulge or avoid. They are designed with similar shape to illustrate the functionality of the marker.

- Feedback and error messages: Feedback is used in order to let the users know whether they successfully performed an action or not. It is important for the users to know their location within the system, and whether they have completed an action successfully or not. In Sheep-and-Go, the users get constant feedback from the colored bottom navigation bar to ensure their location. A successful action is notified by a toast message or an animation. If an action is unsuccessful an error toast is shown. The principle of clear closure is fulfilled with the toast as well.
- Documentation: All information regarding game play and system functionality is documented in its own help window which users can access from any location within the application.
- Reversible actions: The intent of reversible actions is that a user always should be able to backtrack and adjust an action. Sheep-and-Go handles this, by enabling backtracking and cancelling of actions through out the application.
- Visibility: If an item is hard to spot in the interface, the users of an application might not know of its function nor its existence. If an application is going to get used as intended, it is important that all the elements are visible and sticks out from the background.

5.5 Controls

The application is controlled by the touch gestures of a user. These gestures include Tapping, Swiping, Scrolling, and Zooming.

- Tapping is when the user lightly touches the screen with one finger. This gesture is used to activate buttons and traveling through the navigation bar.
- Touch-and-hold is similar to tapping, but once the finger touches the screen the user do not release the finger right away. Sheep-and-Go uses Touch-and-hold on the markers to get a popup modal with additional information.
- Swiping is the gesture of moving your fingers quickly across the screen. This is used to maneuver the map positioning.

- Zooming is a gesture where a user place two fingers on the screen and either drags them apart to zoom in or closer to each other to zoom out. Sheep-and-Go uses zoom in combination with swiping to maneuver the map.

Chapter 6

Design Implementation

This chapter describes the design and implementation of Sheep-and-Go, and will also describe rationale for the technologies and methods used.

6.1 Sheep-and-Go Concept

This part describes the concept of Sheep-and-Go, the features the game includes and how the game play unfolds.

Game Concept

The game itself is as stated earlier inspired by multiple similar applications that evolves around encouraging children to complete tasks for rewards, as well as keeping track of animals in nature. One of the inspirational applications, Pokémon Go 2.4, displays an avatar on a map of which location is based on the real-world location of the device which the game is played on. While moving around in real life, a player has a chance to meet characters called 'Pokémon'. These characters are displayed using Augmented Reality (AR) technology and can be viewed as animations through the camera of the user. Features called 'Pokéstops' and 'Pokémon Gyms' are static locations spread around the map, which users are encouraged to travel towards in real life. The 'Pokéstop' features are locations where these Pokémon characters frequently habit, and the users are able to catch these to display as digital pets, which they can collect and showcase in their inventory. Like in Pokémon Go, Sheep-and-Go displays

an avatar on a map, in this case Google Maps [36] which displays aerial imagery of the real-life location of the user. Also displayed on the map are different icons and marker features which affects the game play and strategy chosen by a user. Rationale and functionality of the icons is as stated below.

Hot spot marker

- **Game function:** The functionality of a hot spot marker is as an opportunity for a player to seize extra points during game play. A successfully registered observation yields a greater amount of points to a player if it is made within the near proximity of a hot spot marker, than if its not. The amount of points is based on the exact length between a users device and the nearest marker. The closer an observation is made, the more extra points is awarded. If a device is within 100 meters of one of these markers, the player will get extra points even if they haven't registered an observation. This is to encourage users to visit specific locations during a playing session.
- **Rationale:** The hot spot markers indicate locations where a sheep farmer wants users to scout for their animals. The location of these markers will vary based on what information the farmers get from previously made user observations. The idea is for the observations to create a situational awareness overview of sheep at any given location, so the farmers get more information as to where their sheep are located.

Avoid marker

- **Game function:** The avoid marker acts as an opposite to the hot spot marker, and will penalize points instead of award them if a users device is present, or an observation is registered in close proximity to a potential animal carcass. The amount penalized mirrors the amount awarded near a hot spot.
- **Rationale:** If there is information about an animal carcass or dangerous terrain being found, it is marked with its own Avoid marker, which indicates that the players should avoid to visit that specific location. The reasoning behind this is that the visuals of a rotten animal could

have a scarring effect on the target group, children, and that they should avoid dangerous terrain.

Sheep icon

- Game function: When the user taps on a sheep icon, a modal with information regarding the observation is displayed. It shows the number of sheep, their color, points rewarded, and the date and time of the observation. The sheep icons has different colors based on the most dominant sheep color of the observation that was made.
- Rationale: The sheep icons displayed on the map are observations previously made in that area. It is intended that each icon lasts for a given, short period of time, so the sheep farmers will receive new and relevant location data.



Figure 6.1: Markers listed from left to right: Hot spot marker, Avoid marker, Sheep icons

6.1.1 Register and Login Screens

When first launching the application the user will face the login screen, and if the user is new and does not yet have an account, they are met with an option to create one. This requires the new user to create a personal username to be displayed on the screen, as well as their email, and a password for security reasons. The screens are navigable from each other. The screens are displayed in Figure 6.8. and Figure 6.3. below.

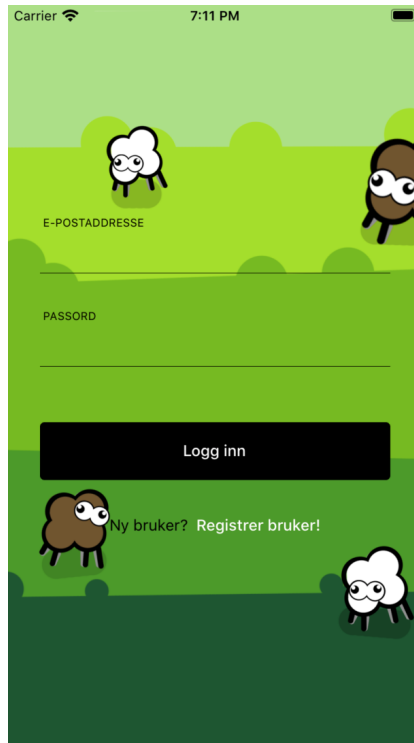


Figure 6.2: Sheep-and-Go login screen



Figure 6.3: Sheep-and-Go register screen

6.1.2 Home

Once a player opens the logs in with their personal credentials, they see themselves on a map zoomed in to the real-life location of their device, as briefly explained in Game Concepts 6.1. See screenshots in Figure 6.4. From this point, the user can navigate the map with Zoom and Swipe gestures to spot hot spots, plan a trip or to get information regarding observations other players have made. The home/default screen also contains a button, whose functionality is for registering a new observation. When pressed, a modal appears on top of the map. See Figure 6.5. This modal consists of sheep icons of different colors and sizes. The idea is for the users to log their observations by creating a representation of the collection of sheep they have spotted. In Figure 6.6. a player has observed two small white sheep, and one large grey sheep. In addition to this, the user can write an additional comment to provide extra information regarding their observation.

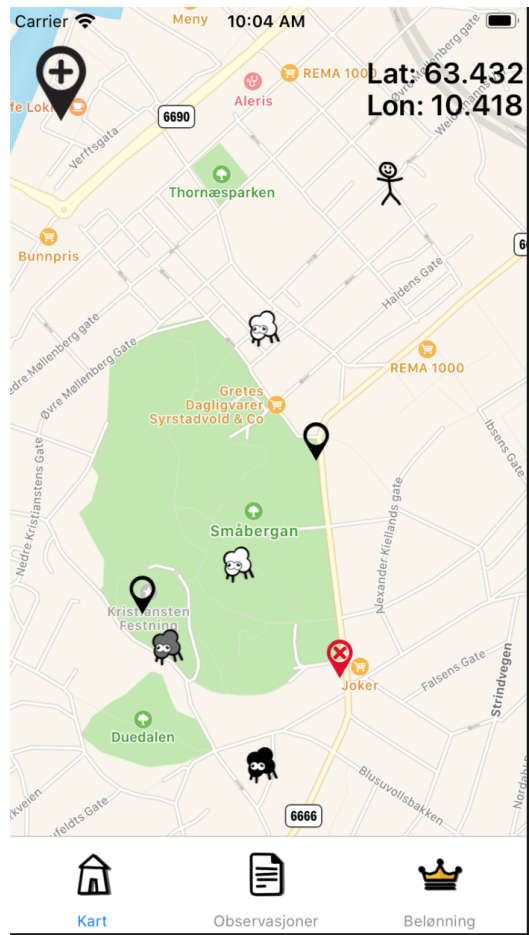
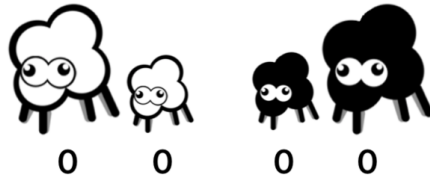
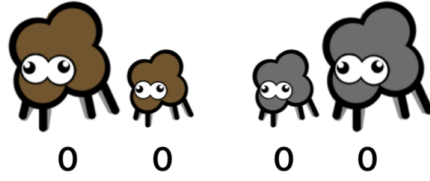


Figure 6.4: Sheep-and-Go map view (default page)

Cancel



Legg til en kommentar!

SEND

Figure 6.5: Observation registration modal



Figure 6.6: Observation example

6.1.3 Observations

The observation tab acts as an observation timeline and let the users see a list of their previous observations as well as see their current point score. Each element of the list represents one observation made. The observations are sorted by date, and contains all details registered by the user, in addition to an explanation to the number of fictive points earned for that specific observation. See Figure 6.7.

Historikk Totalpoeng: 100

nanoseconds=//3000000))
store svarte sauer: 0
små svarte sauer: 0
store hvite sauer: 0
små hvite sauer: 1
store grå sauer: 0
små grå sauer: 1
store brune sauer: 2
små brune sauer: 0
Poeng + distansetillegg: 40
Commentar: Observed by the river
.okasjon: (10.42 63.43)

Observasjon

Timestamp(seconds=1591085197,
nanoseconds=293000000))
store svarte sauer: 1
små svarte sauer: 0
store hvite sauer: 0
små hvite sauer: 0
store grå sauer: 0
små grå sauer: 0
store brune sauer: 0
små brune sauer: 2
Poeng + distansetillegg: 30
Commentar: Observed right next to the station
.okasjon: (10.42 63.43)

Observasjon



Figure 6.7: Observation tab

6.1.4 Prizes

As explained in section 3.1 the application is designed towards children accompanied by their guardians. The reason to include guardians is that the target group includes children that are too young to have their own e-mail addresses and will therefore not be able to operate an account on their own. The reward functionality is therefore prohibited the adults. In order to keep the children from accessing the reward functionality, the entire "Prizes" tab is password protected. Once a guardian has bypassed the password protection, they have access to a new set of functionalities.

These are functionalities for deducting points gathered by the very same account, for real-life prizes in return. The exact type of prizes are to be decided by the families themselves. Examples seen during user tests are minor rewards like ice cream or candy. After a guardian has registered a point deduction as well as commented the prize, the information is added to a timeline similar to the list on the "Observations" section.

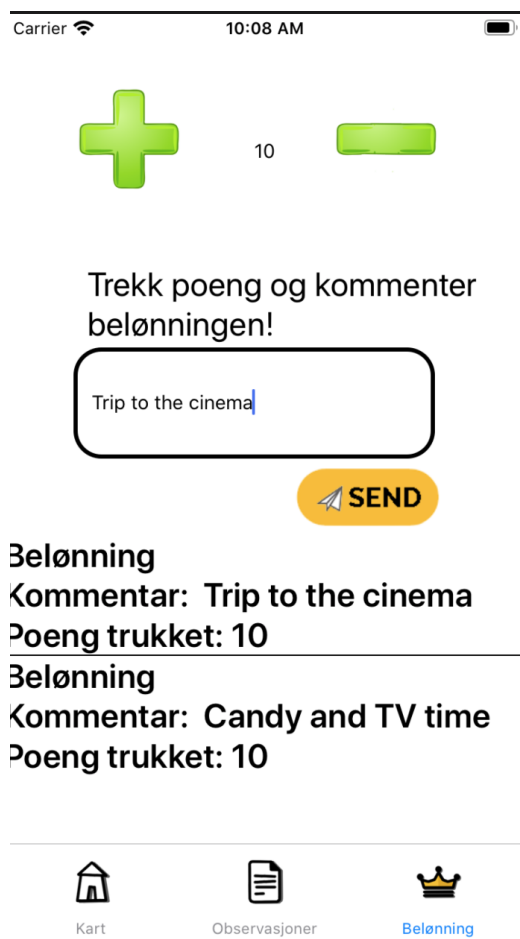


Figure 6.8: Prizes tab

6.1.5 Navigation Footer

The navigation footer is the only way to navigate between the locations within the application. The current view is highlighted in a blue color indicating in which location of the application the users are at any given time. See Figure 6.9.



Figure 6.9: Navigation bar with current location highlighted

6.2 Technology Rationale

This chapter describes the reasoning and logical basis behind the technical decisions taken during the development of Sheep-and-Go.

6.2.1 Platform

From the beginning it was clear that the game was going to be developed as a mobile application. Partly because it is more practical to carry a smaller device while hiking in mountain terrain, but also because the smartphone is the most popular device to own. According to Statista [37] there are currently (2020) around 3,5 billion smartphone users world wide. There exists several different mobile operating systems. Thus, in order to choose the right one for development, multiple factors were taken into consideration.

- Operating system popularity
- Component compatibility
- Developer knowledge and practicality

6.2.2 Operating System Popularity

Since Sheep-and-Go is developed in Norway for Norwegian families, this rationale is based on statistics from the current Norwegian market.

According to [38] the currently largest mobile operating system is iOS [31] which as of April 2020 had a market share of 54.74%, followed by Android [30] with a market share of 44.94%. The last 0.32% of market shares are divided between Samsung [39], Windows [40], and Blackberry OS [41].

6.2.3 Component Compatibility

Another factor of high importance is whether or not the operating systems in question are compatible with technology required to complete the development. Required technology are up-to-date map-libraries with support for Geolocation, and a stable good-fitting database.

6.2.4 Developer Knowledge and Practicality

The developer of Sheep-and-Go had some previous knowledge in working with both iOS and Android SDKs, but not significant enough to chose platform based on that.

6.2.5 Rationale Conclusion

Since the whole development process was done single handed, there was no option to develop several applications for different operating systems, due to the amount of extra work it would demand. It was therefore chosen to develop one single code base. Due to the low amount of Samsung, Windows, and Blackberry OS market shares, development in one of these operating systems would exclude over 99.68% of the Norwegian user market. They were consequently deemed unfit for the purpose of this development process. There is no clear indication to whether developing the application purely for iOS or Android would be more beneficial in this case, considering that both of the operating systems are almost equally popular. There is however, through a cross platform environment a way to develop a single code base that can cast to both iOS and Android at the same time. As the developer has prior knowledge in React development, it was chosen to develop the application in React Native. This choice satisfies the factor of market popularity as the application can be casted to the two operating systems that cover 99.68% of the market shares. It also satisfies the developers knowledge and practicality, as the developer had previous experience with React. React Native includes

support for map-libraries with Geolocation as well as compatibility with free-to-use database clients, such as Firebase.

Chapter 7

Software Architecture

This chapter describes the software architecture, the fundamental structure of the system. To better illustrate, the architecture is described using Kruchten's 4+1 method [42], which is a view model designed to describe software intensive systems.

7.1 Kruchten's 4+1 Architecture

The 4+1 Architecture consists of the following four views in addition to a last segment called scenarios. Each view will be described in their dedicated sub-chapters. The reasoning behind illustrating the software architecture using Kruchten's 4+1 method in contrast to other view models like Three Schema Approach [43] and Zachman Framework [44], is that Kruchten's 4+1 separates the different stakeholders of the application, instead of presenting a system from only one view. Since illustrating the software architecture in Kruchten's 4+1 separates into different views, this makes it easier for a certain stakeholder to understand the architecture without the models being clustered with information that do not address them.

The stakeholders of Sheep-and-Go are categorized as followed:

- Developer
- End user type: Child
- End user type: Adult
- End user type: Sheep farmer

Sheep-and-Go has several stakeholders with different interests and viewpoints which are listed as bullet points in the section below.

- Logical view
- Process view
- Physical view
- Development view
- Scenarios

7.1.1 Logical View

The logical view describes the functionality that is accessible to the end users. This section addresses the end-user stakeholders and includes the activity diagram of the application.

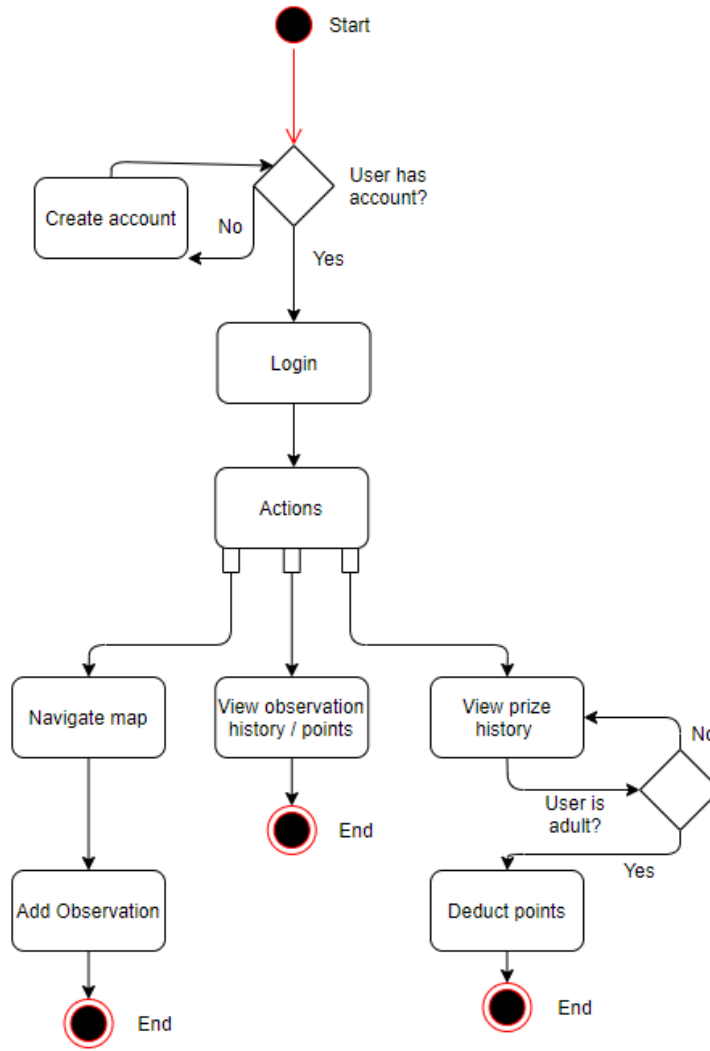


Figure 7.1: Activity Diagram

The activity diagram illustrated in figure 7.1 displays the process flow in which a user interacts with the Sheep-and-Go application. After booting the application, a user can either login or register an account and then proceed to log in. After the login process is completed, the user can choose between several actions. Either navigate and add observations to the app via the map view, check observation history and current point status in the observation

view, or view prize history in the prize view. In order deduct points in exchange for prizes, the system provides the user with a password protection obstacle to ensure that only adults can deduct points.

7.1.2 Process view

The process view describes the communication of the processes within the system, and concerns aspects of scalability, concurrency, performance, and distribution. This section includes, sequence diagrams presented in UML.

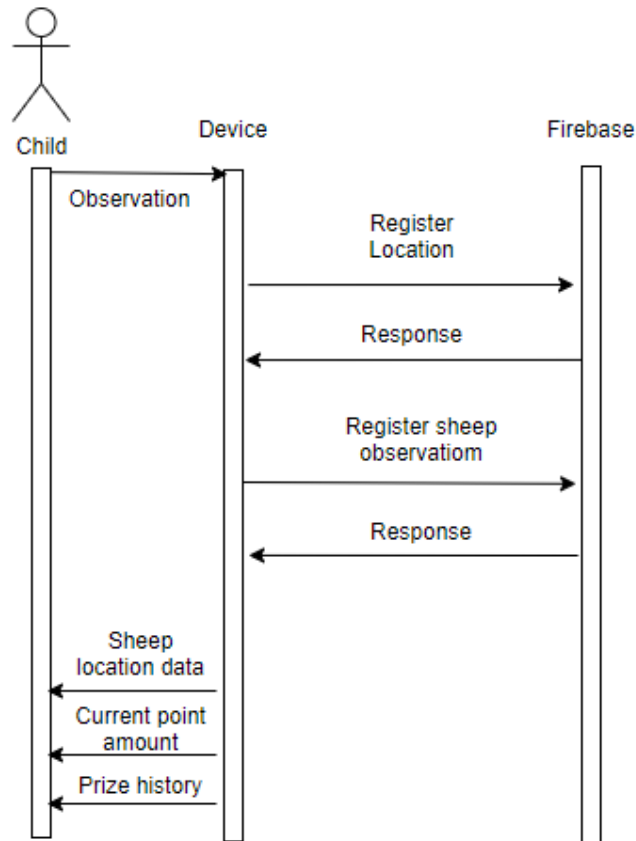


Figure 7.2: Sequence diagram of the interaction between a child and Sheep-and-Go

Figure 7.2 shows a typical interaction between a child user and the application. As a child is not intended to own the account themselves, they are not a part of the authentication process. The interaction process therefore starts at the registering of location. A child is then able to register observations of sheep, and view observation history, check their amount of points, and view the prize history.

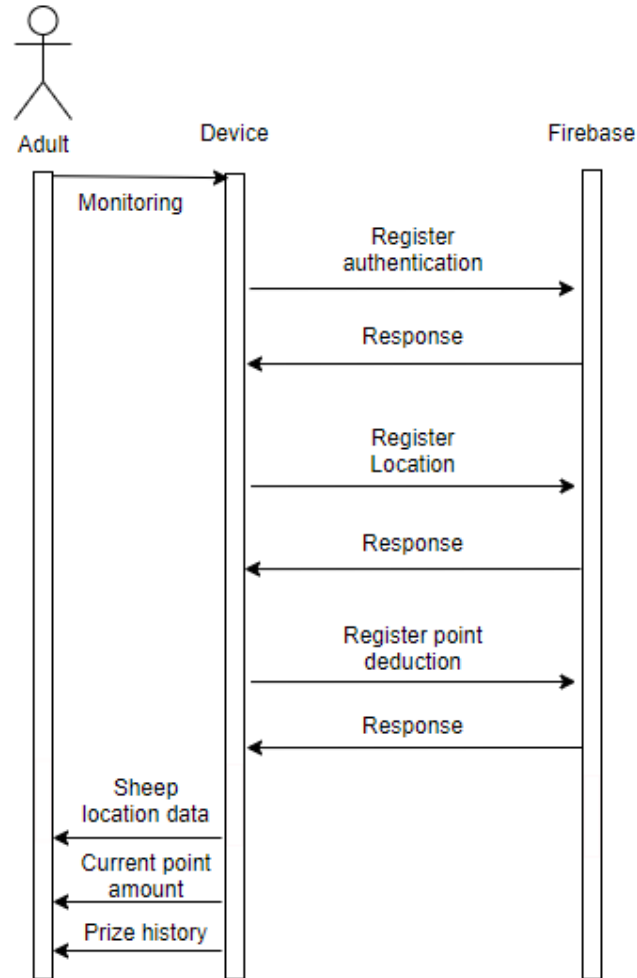


Figure 7.3: Sequence diagram of the interaction between an adult and Sheep-and-Go

Figure 7.3 shows a typical interaction between an adult and the application. It is intended that the adults of a family are in charge of the family's account, and are the ones providing their e-mail and choosing a password. Their interaction with the application therefore begins with authentication. They have access to the same functionalities as child users, see 7.2, however they are also able to deduct points from the account in

exchange for real life prizes for their children.

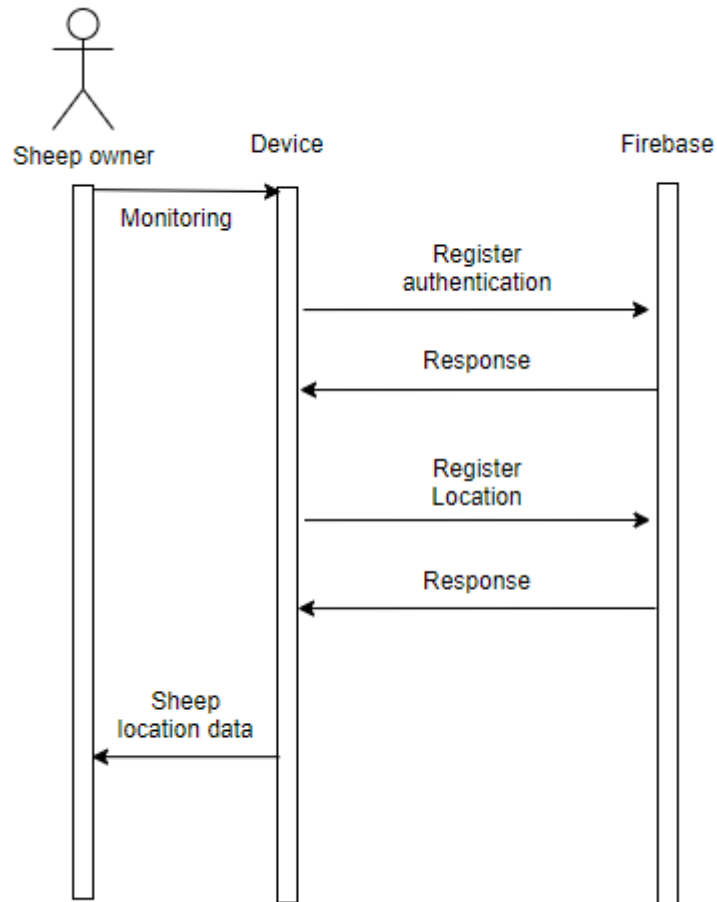


Figure 7.4: Sequence diagram of the interaction between a sheep farmer and Sheep-and-Go

Figure 7.4 shows a typical interaction between a sheep farmer and the device. Technically the farmer has access to the same functionalities as any other adult, however their motivation is to gather information about the whereabouts of their sheep. Registering sheep observations and point deduction is therefore not displayed in the sequence diagram.

7.1.3 Physical View

The physical view concerns the physical connection between the components of the system. It includes a deployment diagram and a client-server diagram presented in UML.

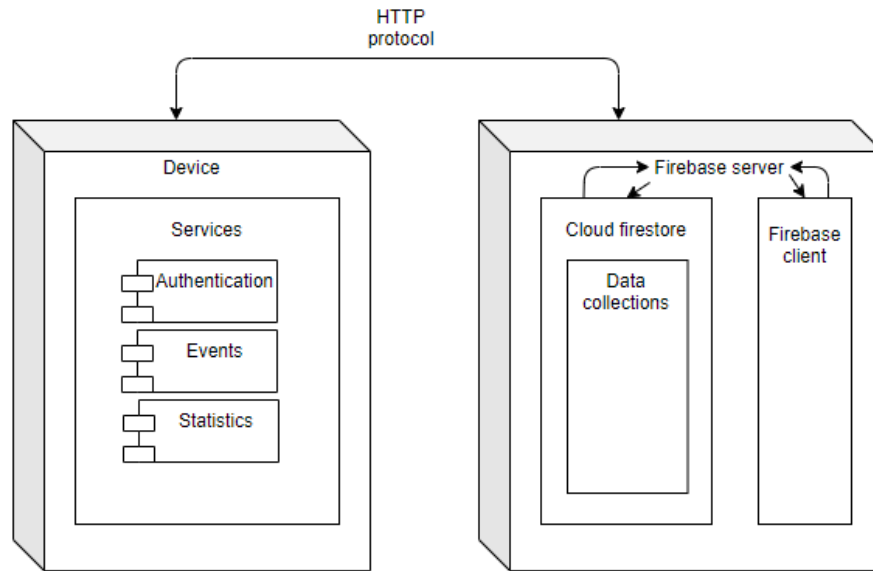


Figure 7.5: Deployment Diagram

Figure 7.5 show the deployment Diagram. This is a diagram of the two main parts, the Firebase server, and the iOS or Android device running the application. They communicate through a HTTP protocol. The device acts as a host for the react application package and contains the functionality for communication with the Firebase server. The Firebase server includes the Firebase client which manage the services of Firebase, and the Cloud Firestore which manages storage of the data, in collections.

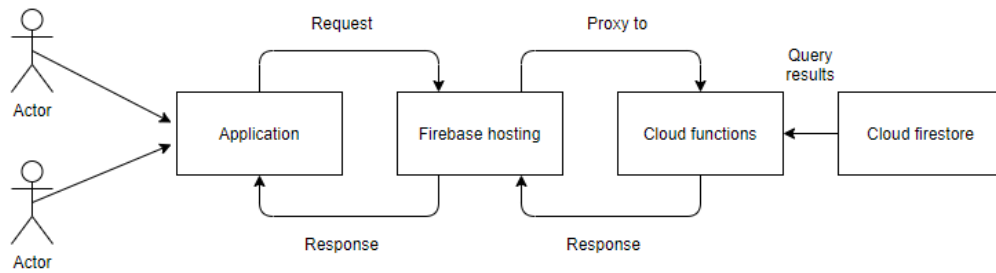


Figure 7.6: Client-server diagram

Figure 7.6 shows the client server diagram. It describes how actors use the application, and how the middle-managment Firebase services communicates with the data storage provided by Cloud Firestore.

7.1.4 Development View

The development view addresses the system from the view of the developer, and includes a diagram of the components within the application, presented in UML. The component diagrams are represented hierarchically. For presentative reasons the diagram is split into multiple smaller sections in order to make them readable in document form for any concerning audience.

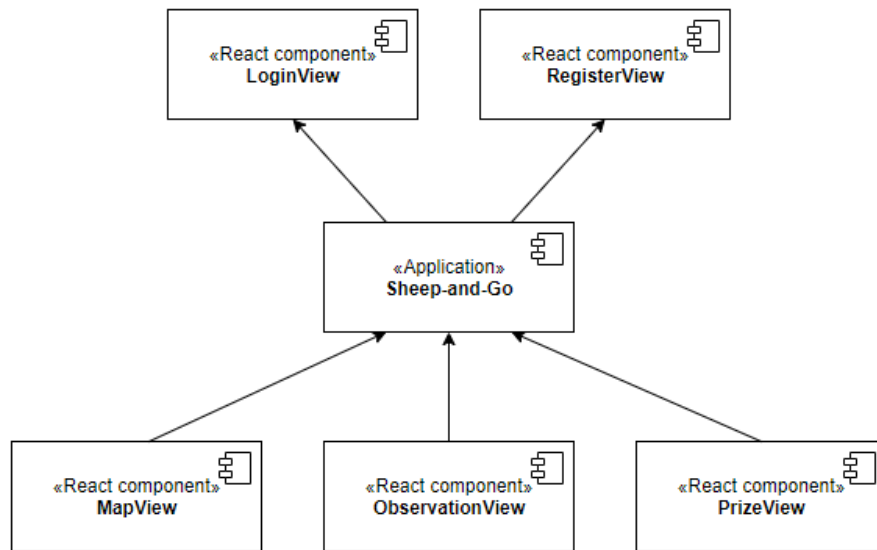


Figure 7.7: Navigation view

Figure 7.7 hierarchically illustrates how the different views are connected within the Application. The first level consist of the login view and the account register view. The reason they are placed on top is because interaction with either of them is unavoidable in order to reach the application content. The second level is the Application of Sheep-and-Go, followed by the three view components map view, observation view, and prize view on the level beneath.

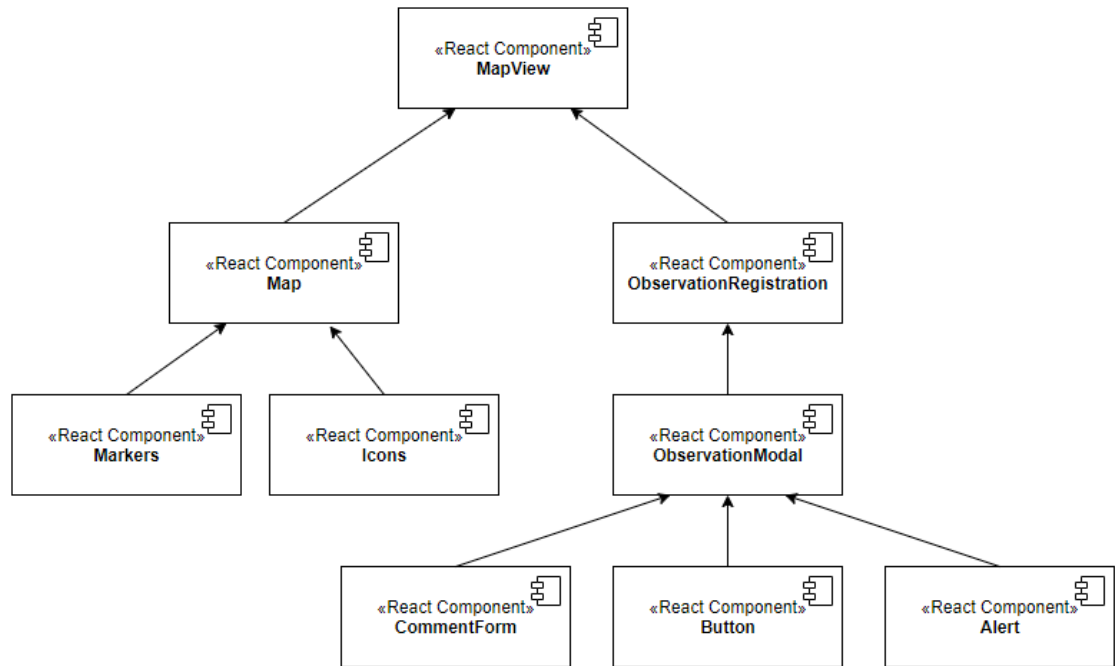


Figure 7.8: Class diagram of the map view

Figure 7.8 describes the components found within the map view. The top level is the component called map view, which is the same as the one illustrated in level three of 7.7. The child components of the map view is a continuation of 7.8 which as mentioned was split in multiple diagrams. One of the main components found in the second level, the map, implements Google Maps through a react-native-library. Displayed on this map are markers and icons, which have their own specific functionalities, thus are components themselves. The second main component is for registration observation of sheep. This includes a modal component with underlying components for commenting form, buttons, and alerts.

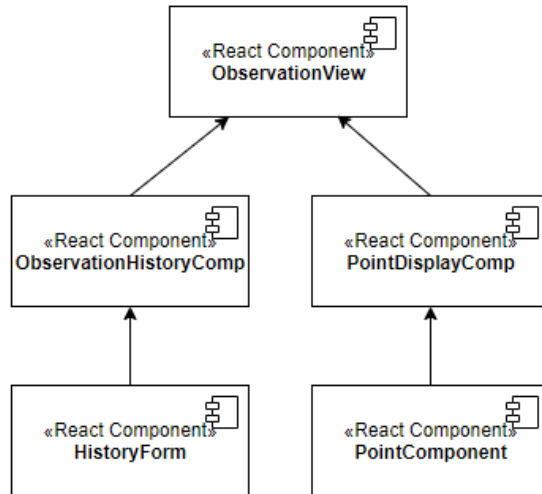


Figure 7.9: Class diagram of the observation view

Figure 7.9 is as 7.8 also a continuation of 7.7. The two main components are for display of observation history and for display of points. The child component of the observation history component, extracts user data from Firebase and displays the information in a sorted list format. The child component of the point display component calculates the current amount of points of a user and displays the number on the screen.

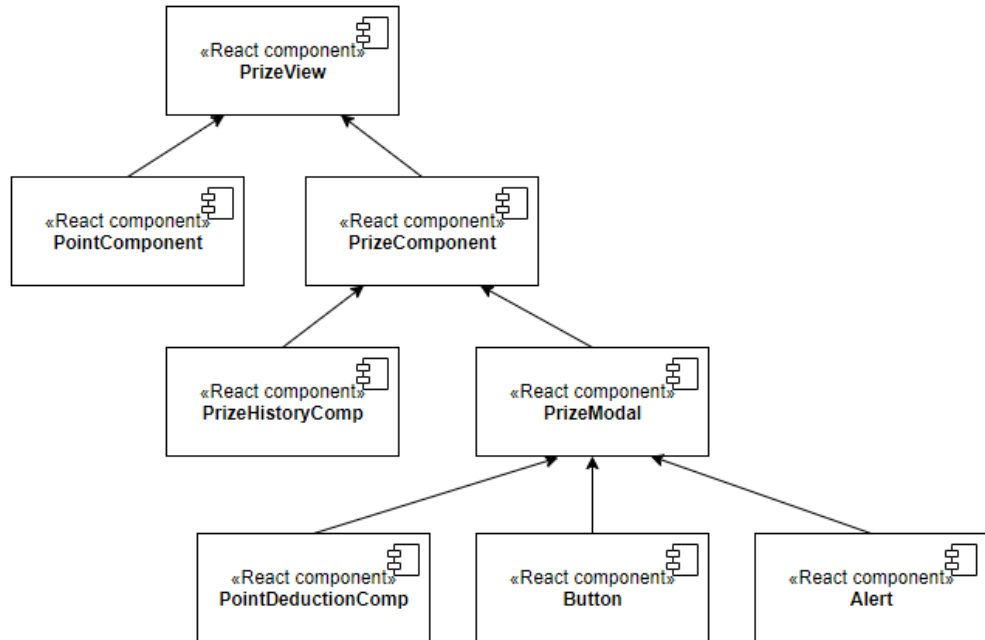


Figure 7.10: Class diagram of the prize view

Figure 7.10 describes a continuation of the prize view seen in level three of 7.7. The main components are the component for prize history and a component for deduction of points. The prize History component is similar to the history component in 7.9 as it displays history of prizes given to children in exchange for points. The prize registering component includes a modal with child components for buttons and alerts.

7.1.5 Scenarios

Scenarios is the last part of the model, which itself is not considered a view, but rather a set of use cases to describe the interaction between components and to validate the architectural design of the system. These scenarios are the same as the use-cases seen in section. 4.3, and will thus not be repeated.

Chapter 8

User testing

This chapter describes the different stages of Sheep-and-Go user testing. It contains early stages of paper prototype testing, planned user testing, force majeure and adjustments, as well as post adjustment user testing.

8.1 Test Subject

The target group for the user testing are children between 7-13, with guidance from their parents or guardians. Since some functionality within the application is targeted to only one of them. They are categorized as different test subjects:

- Child test subject
- Guardian test subject

8.2 Paper Prototype Testing

In the early stages of the project, a user test was conducted with voluntary participants within the target groups, using Paper prototyping. This is a common method in user-centered design processes. The aim is to gather feedback at a very early point in the development process, to be able to detect and overcome design obstacles before the programming starts. The method works by presenting a drawn sketch of the user interface to the test subject. This sketch acts as a substitute for an actual digital system. The test subject

executes actions with the same touch gestures as they would using a real application. The systems response is mimicked by new sketches layered over the previous one. This gives the test subjects a feel of navigation, actions, and responses of the system. By Paper prototyping a developer can save both time and resources by avoiding to redo irrelevant or wrong work, and seeing that Sheep-and-Go is a single handed developed application limited by time constraints, it was chosen to conduct such a test.

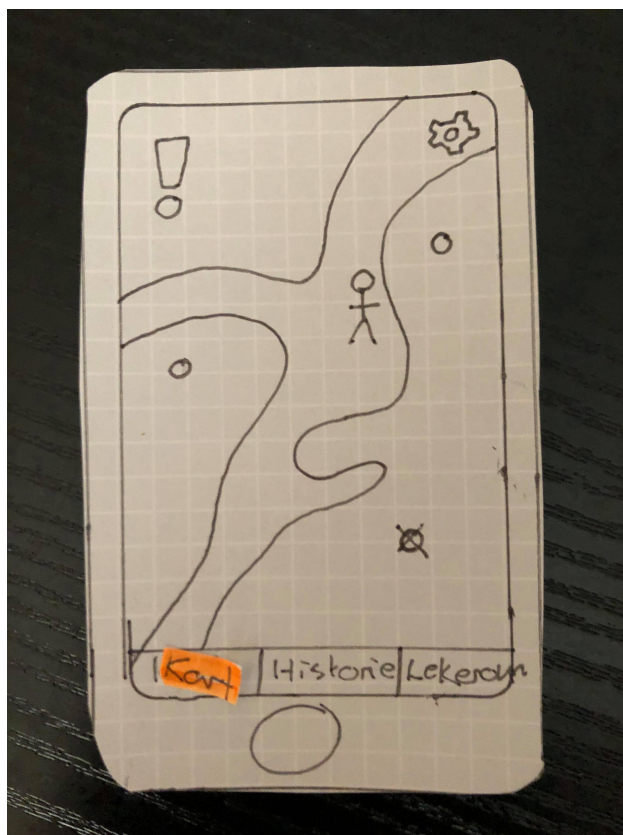


Figure 8.1: Paper prototype of the map view in an early version of Sheep-and-Go. The colored overlay indicates the current location within the application.

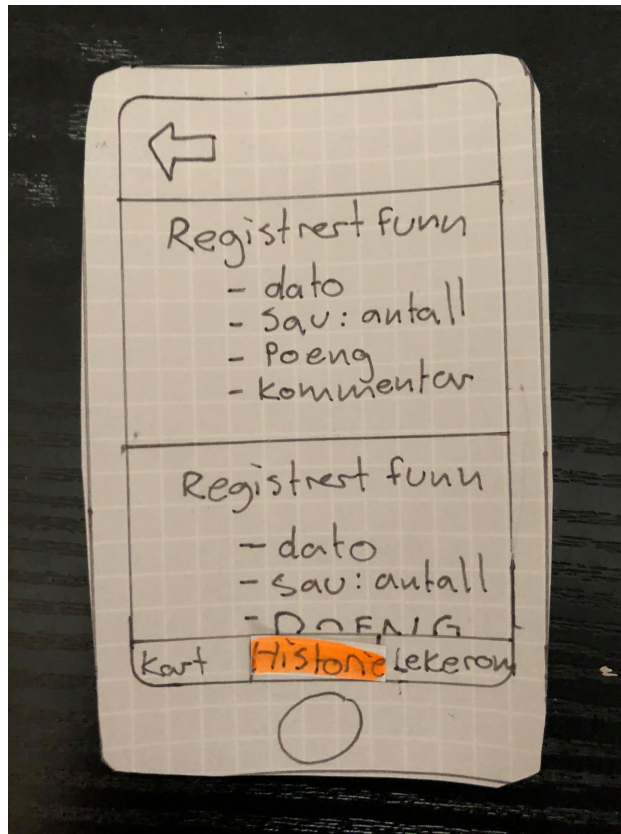


Figure 8.2: Paper prototype of the history view in an early version of Sheep-and-Go. The colored overlay indicates the current location within the application.



Figure 8.3: Paper prototype of a playroom view in an early version of Sheep-and-Go. The colored overlay indicates the current location within the application.

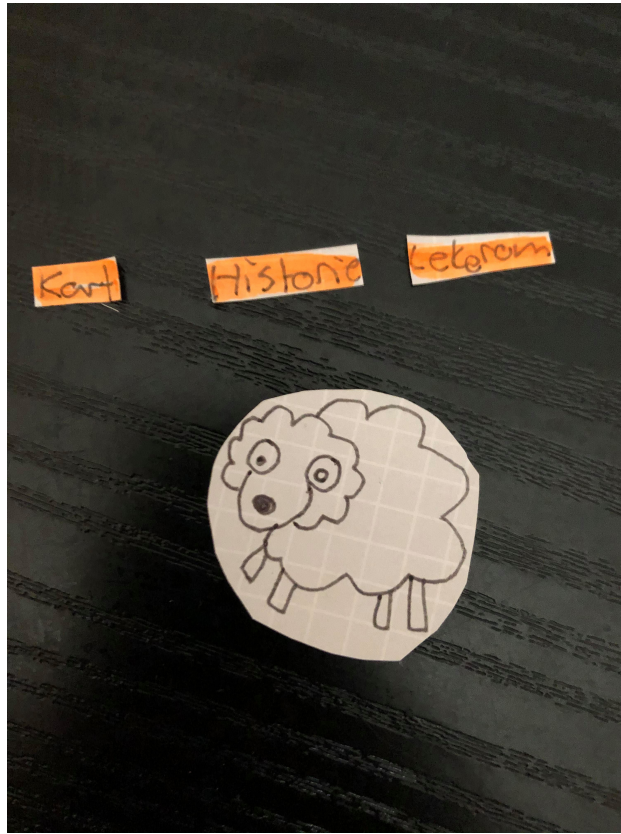


Figure 8.4: Paper prototype overlays to indicate the applications response to actions executed by test subjects

8.2.1 Prototype Results

The feedback from the test subjects resulted in changes to the original concept. A number of buttons were misinterpreted and misused as a consequence of poor affordance, thus received negative feedback from the test subjects. The design of these buttons were consequently adjusted to fit the need of the target group. Other feedback included the entire Playroom view. The idea behind the view is for each account to have a personal, customizable digital mascot that could be dressed up and that children could play around with. The feedback towards the Playroom was mostly positive, however argued the necessity for a digital mascot. The Playroom concept was after the test, deemed a non-necessity and deprioritized for more important functionality. To read more about the playroom, see 9.1.

8.3 Planned User Testing

The user testing was intended to be conducted in the park area surrounding Kristiansten festning[45], a historic fortress, located in Trondheim, Norway. The fortress is located in central Trondheim, which is easily accessed by the potential test subjects living in Trondheim. The park area is a field of approximately 1 km² of which terrain consists of grass and trees. This area would be ideal to mimic the typical landscape where Norwegian sheep habit. In preparation for the tests, the developer would hard code both hot spot markers and avoid markers on the map at specific points of the park area, as well as placing certain physical objects, visible from different angles of the park area, mimicking sheep. Each test was planned to consist of three parts:

- Hike with observation registering, conducted by child with adult supervision
- Deducting points for rewards, conducted by adult
- Questionnaire, conducted by child and adult

8.3.1 Hike

The child and parent test subjects are asked to plan and complete a hike whilst registering any observation of "sheep" on the way. Their goal is to maximize the amount of points earned and minimize the amount of points deducted.

8.3.2 Point Deducting

The parent or guardian of the child test subject is asked to navigate to the Prizes tab and deduct points from their children in return for a real-life reward, which in this case is provided by the developer.

8.3.3 Questionnaire

After the testing is completed, the families are handed a questionnaire where they are asked to answer questions on, regarding the system, providing the developer with useful information for further developing and adjusting.

8.4 Force Majeure and Adjustments

During the development of Sheep-and-Go an infectious disease caused by "*severe acute respiratory syndrome coronavirus 2 (COVID-19)*" [46] began spreading from its origin of Wuhan, China towards the rest of the world in a high rate. This virus is considered highly transmittable, and life threatening to a large part of the society. After The World Health Organization (WHO) [47] in March 2020 declared the outbreak as a worldwide pandemic, the Norwegian Government was forced to put the entire country of Norway into a lock down mode in efforts to reduce spreading of the infection. As the virus spreads between people during close contact and by droplets produced by breathing, talking, and coughing, all unnecessary human contact was to be avoided. This resulted in a situation where the government heavily prohibited unnecessary contact between members of the society by law, locked down schools, non essential businesses and encouraged all non-essential workers to stay at home at all times.

8.4.1 Impact of COVID-19

The development of Sheep-and-Go was not directly affected by the societal changes COVID-19 brought with it, as the application is developed single handed. The process of development could thus, with minor changes continue as normal. User testing of the application could however not be completed as planned because of human contact restraints. The original user testing plan was consequently canceled, and replaced with a new one with alterations added to be able to test the functionality.

8.4.2 Adjustments and Post Adjustment Testing

List of adjustments made to the original planned test, in order to be able to arrange user testing while following Government rules.

- Location: As any physical or close contact with the test subjects were prohibited, the user tests could not happen in person, as they were initially intended to. They were instead held over Skype for business[48], an instant messaging software that offers screen sharing functionality and web camera video calls.
- Game play: As mentioned in 8.3, a large part of the user testing consisted of planning and completing a hike in the park surrounding Kristiansten festning, in order to gather information and experience the location based differences in functionality. As the user interaction had to be held over an instant messaging software, the location tracking functionality would not work, and was consequently removed. Instead the developer hard-coded several scenarios that would result in different outcomes for the test subjects, in order for them to be able to test all the functionalities of the application. The physical objects that would be spread around the test area to mimic sheep were replaced by imaginary sheep, introduced in test-stories told by the developer at each new hard-coded location.
- Test method: Instead of the subjects manually testing the application in supervision of the developer, the method of testing was converted to a "Wizard of Oz" technique [49]. This technique is not as viable as manually tests as the moderator loses some key aspects of the test

subjects reactions and body language, that are more apparent via in-person testing. It is however applicable through instant messaging software via webcam video calls and screen sharing, thus an acceptable option considering the Force Majeure restrictions mentioned in 8.4

- Game mechanics: The intended testing included controlling the application on a smartphone, the intended device, using the touch gestures mentioned in 5.5. In the adjusted version of the user test, using the "Wizard of Oz" technique, the application was ran on an emulator on the moderators computer and made visually available to the test subjects through screen sharing. When the application is run on on an emulator, it is controlled using a computer mouse with different gesturing than on a smart phone. Instead of confusing the test subjects with controllers they are not used to, the test subjects explain their intended actions to the moderator which mirrors them in the application on the shared screen. The intended effect is that the test subjects experience that they are in control of the functionality.

The idea behind these adjustments was to retain as much similarity to the originally planned user test as possible, thus fulfilling the same goals as the original plan would have done. By changing the test location from in-person to remote, the user testing could still be conducted. By hard-coding locations, the test subject would still be exposed to all intended functionality without actually changing locations. And by using the "Wizard of Oz" test method, the developer of Sheep-and-Go would still be able to get reactions from the test subjects. Although it could be discussed whether or not the originally planned tests would yield better and more accurate results, the adjustments would make testing of the application possible while still applying to the initial test goals.

8.4.3 Wizard of Oz

The wizard of Oz method is a testing method where a user interface concept, in this case the current version of Sheep-and-Go is presented to a test subject, by a moderator "The Wizard" through screen sharing. The user is given different tasks that require them to move around and explore the GUI. In order to navigate the app, the user speaks loudly what actions they wish to performed. These actions are then answered by moderator controlling the

system, to simulate a real response. The Wizard of Oz test method allows the developer to get an immediate reaction from test subject, to functions of the concept that is being tested.

8.5 Questionnaire

In order to generate suggestive test data, the test subjects were sent a questionnaire via e-mail with easy to answer, multiple choice closed questions. The reasoning behind using a closed questions instead of other data gathering methods, is first of all that it enables the test subjects to answer from any location and time they want, which is a preferable option during lock down conditions. The closed question questionnaire also has advantages such as, cost effectiveness, convenient data analysis, anonymity maintaining, and the ability to get structured data without developer bias. The reasoning behind the low difficulty level of the questions was to be able to receive answers from all participants, both children and their parents. While these questions were easy to answer, they were nevertheless carefully designed and well defined so they would generate usable data. See appendix B for the entire questionnaire.

8.5.1 Validity and Reliability

The validity of a questionnaire is the assessment of the accuracy it has. In order to maintain validity in the case of Sheep-and-Go testing, the questionnaire was designed with regards to principles of questionnaire development, described by Bill Gillham in Developing a Questionnaire[50] and in a paper published by Sheffield Hallam University [51]. These principles in mention are:

- Question formatting: The questions should be easily understandable and no longer than 25 words each. The test subjects should be presented with examples on how to correctly answer.
- Choosing questions with the target group in mind: The users should be able to understand the scenarios and question text. In order for the developer to gather data, the questions asked should be developed with the target group in mind.

- Avoid ambiguity: In order to circumvent confusion it is important to ask obvious questions that don't have several interpretations or sentences that consists of several questions.
- Avoid leading questions: It is important to ask questions in an objective form so the answer of the test subject is not biased by the opinion of the person that made the questionnaire. In this case the developer.
- Consistency: Importance of consistency of the question type is up to discussion. Bill Gillham states that when creating a questionnaire, the developer should be careful of mixing open and closed questions. For analytic convenience he argues that it may be better to have all questions either opened or closed.

Reliability of a questionnaire is a way to assess the quality of the measurement procedure used to collect data. Since the questionnaire is entirely based on suggestive test data, it is not possible to achieve an exact quality of the measurement procedure, but are however steps to increase the reliability of the questionnaire. During the testing, all participants were presented with the same version of the application and given the same cases and tasks to solve. These precautions ensures that the answers they give are based on the exact same information.

8.6 Results

To gather accurate answers from both children and adults, the questionnaire was made in Norwegian. For presentative purposes and to keep the thesis paper consistent, the questions are translated to English in the following tables. The original questionnaire is located in Appendix B.

8.6.1 Questionnaire Segment One

The first segment of the questionnaire is an introduction where personal information of the test subjects are gathered. The entire segment consists of only two questions. The first being their age, and the second being an estimate of how many hours each week they spend on mobile games. The reason for this segment is to create a personality behind the answers without revealing any sensitive information. In table 8.1, we can see that the test

subjects consisted of 58.3% Adults and 41.7% children. Their mean usage of mobile games are above 1.79 hours per week.

Participant number	Role	Hours of mobile games per week
1	Adult	3-4
2	Adult	0
3	Adult	0
4	Adult	1-2
5	Adult	0
6	Adult	0
7	Adult	0
8	Child	6+
9	Child	6+
10	Child	1-2
11	Child	1-2
12	Child	1-2

Table 8.1: Test subject information

8.6.2 Questionnaire Segment Two

Section two of the questionnaire is composed of questions regarding the users experience and feelings towards their time playing Sheep-and-Go. The questions were as mentioned in 8.5 closed, presented in a multiple choice manner. These choices were the same to all questions and consisted of five options, with numerical translations. The options were translated into numerical values and color coded in order to differentiate and be able to compare them against each other. The answers and their numerical translations are the following.

- To a very large degree: Numerical value of 5, color blue.
- To a large degree: Numerical value of 4, color red.
- To a moderate degree: Numerical value of 3, color yellow.
- To a small degree: Numerical value of 2, color green.
- To a very small degree: Numerical value of 1, color purple.

Question	Mean	Median
To what grade did you enjoy the application?	3,9	4
To what grade did the application motivate you to go outside?	3,5	4
To what grade was the level of difficulty suiting for you?	4,3	5
To what grade was the goal of the application clear?	4,4	5
To what grade were the functionality of the application understandable?	4,0	4
To what grade were you please with the feedback?	4,1	4
To what grade were the rewards your motivation for using the application?	4,0	4
To what grade would you continue using the application?	4,0	4
To what grade was the application genre suitable for you?	3,2	3

Table 8.2: Statement results

To what extent did you enjoy using the application?

12 responses

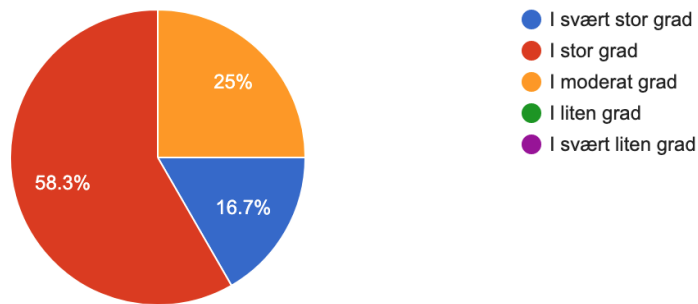


Figure 8.5: Extent of enjoyment

To what extent would you still use the application?

12 responses

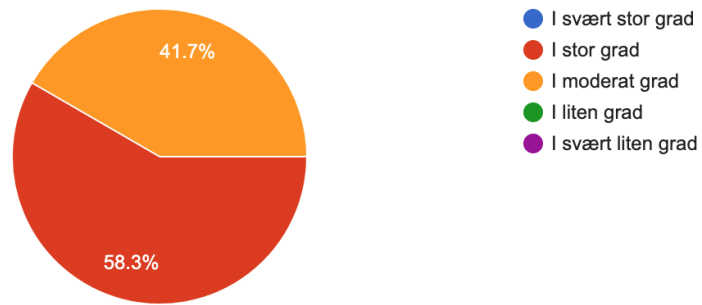


Figure 8.6: Extent of continuing use

Figure 8.5. and Figure 8.6. presents data regarding how the test subjects felt towards using Sheep-and-Go. The results show that every test subject at least moderately enjoyed and would continue using the application, and that the majority would, to a large degree. From these results it is apparent that the application is appealing to both children and adults, proving the intention of the developer.

To what extent did the application motivate you to go for a walk outside?

12 responses

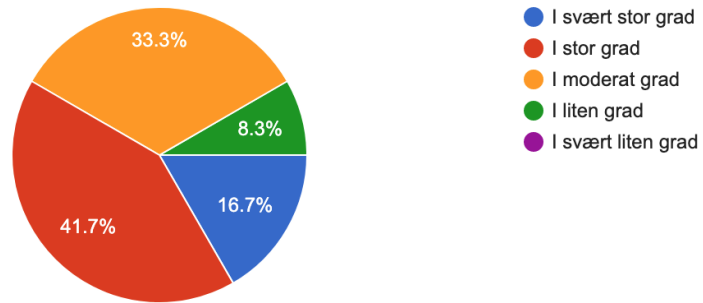


Figure 8.7: Extent of motivation to go for a walk

To what extent was the opportunity for rewards the motivation to use the application?

12 responses

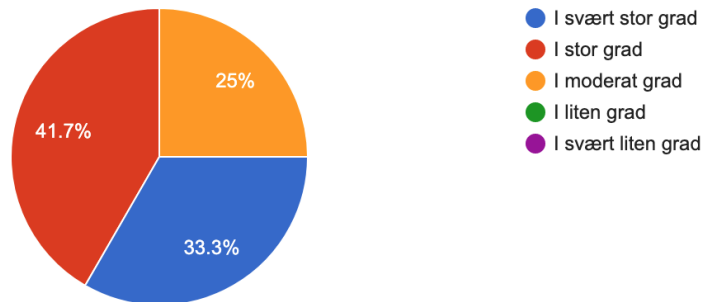


Figure 8.8: Extent for rewards being the motivation to use the application

Figure 8.7. and Figure 8.8. presents data regarding the motivation towards the use of Sheep-and-Go. A majority of the test subjects were motivated to go for walks outside after using the application. This

motivation was in a large degree based on the rewards the players get for observing sheep. These results shows that the application can help motivate both children and adults to go outside, thus spending less time in front of their computers.

To what extent was the genre of the application suitable for you?

12 responses

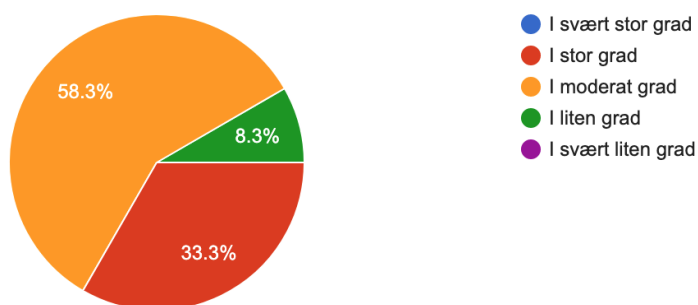


Figure 8.9: Extent of application genre being suitable

To what extent was the level of difficulty appropriate for you?

12 responses

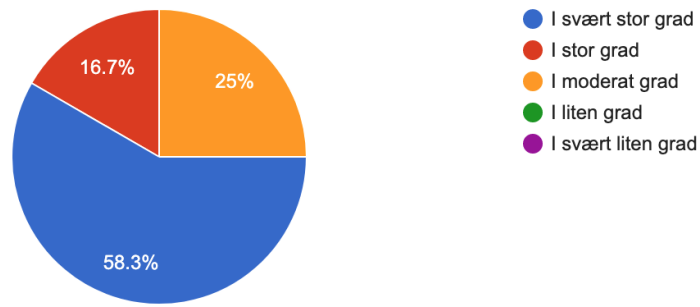


Figure 8.10: Extent of difficulty being appropriate

Figure 8.9. and Figure 8.10. presents feedback data regarding the degree of which the applications genre and difficulty matched the preference of the target group. The figures show that none of the participants found the application to be too difficult, however the mean of 4,3 indicates that it was challenging. The feedback seen in Figure 8.9. indicates that the genre is suitable for children and adults.

To what extent were the application's functionalities understandable?



12 responses

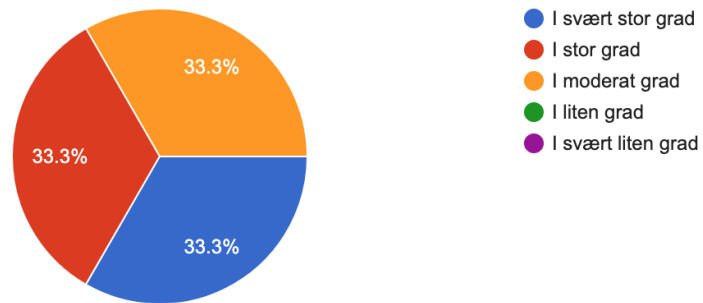


Figure 8.11: Extent of functionality being understandable

To what extent were you satisfied with the feedback you received after using the features of the application?



12 responses

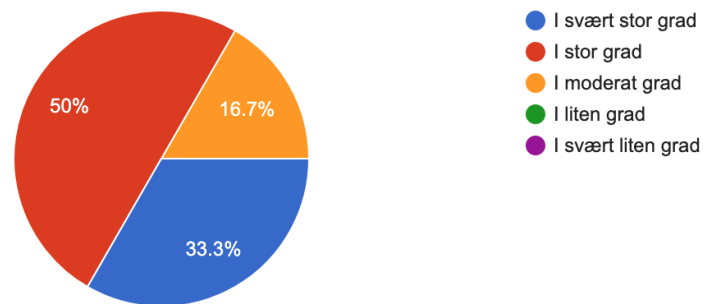


Figure 8.12: Extent of feedback satisfaction

To what extent was the purpose of the application clear?

12 responses

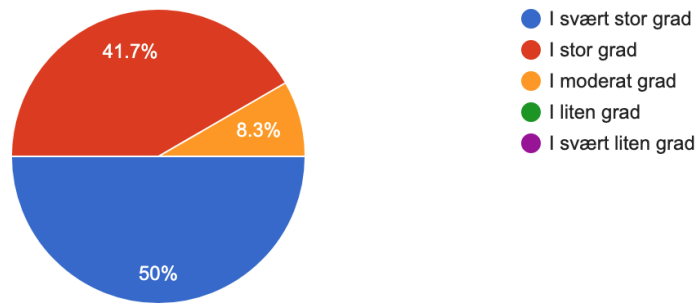


Figure 8.13: Extent of application purpose

Figure 8.11., Figure 8.12. and Figure 8.13. Presents data concerning the ability to understand the functionalities, the level of feedback satisfaction, and to which degree the purpose of the application was clear. From the figures it is clear that the test subjects in average were satisfied with the feedback received when performing an action, and that the purpose of the application is clear.

Chapter 9

Future development

9.1 Future Development

Due to the time- and workload restraints of a single handed developed application, Sheep-and-Go could not consist of an excessive amount of functionality. During the planning, development, and user testing of Sheep-and-Go, a bundle of great ideas were therefore deprioritized for functionality deemed more essential for this initial version of the system. According to the functionality it was planned to consist of, the Sheep-and-Go application was completed. This however, does not mean that the application does not have points of improvements. Points of improvements for both functionality within the application and choices of technology were uncovered consistently, most notably during the phase of user testing. In contrast to uncovering possible improvements alone as the developer did, the user testing let the target group test the application. The children and their parents came up with a lot of ideas that shed a light on possible future features. The points of improvements are categorized into two parts, application improvements, and general improvements.

9.1.1 Application Improvements

Application improvements are made up of modified or new functionalities that has the potential of enhancing the users experience in different ways. The following list consists of functionality that did not make it in the initial product, and ideas from the test subjects.

- A playroom section displaying an account bound sheep character, personalised with skins and behaviour bought with game points.
- Functionality for automatically generating walking routes for families. This generated route could be based on either the maximisation of points or distances suitable for children of different ages.
- Adding a separate account for sheep farmers, where they are able to select the placement of hot spot markers on their land.
- Making height of terrain a factor for generating points, see Technology Improvements
- Add functionality for players to invite more people to play Sheep-and-Go in exchange for points.

9.1.2 General Improvements

General improvements are made up of technology choices that could enhance either development or user experience. As well as non-specific points of improvements.

- Adding Google Elevation Service API [52]. This is an elevation API that let the developer extract the elevation data of any point of the terrain within Google maps mapping service. The addition of Google Elevation Service API would enable functionality for making terrain height a factor for generating points. The idea is that observations made on top of hills should yield more points as they are harder to get to.
- The user testing of Sheep-and-Go was completed using instant messaging software, due to Force Majeure 8.4. Even though these tests were a valid option, they were as mentioned in 8.4 not optimal.

Testing of the application in the intended environment would be considered optimal and would therefore be an improvement to the process.

Chapter 10

Evaluation

This chapter contains evaluations of both the product, and the project itself. Note that this chapter is written from the first person perspective of the developer, thus a different narrative than the rest of this project thesis. This is to more accurately reflect and evaluate the level of achievement of the project.

10.1 Product Evaluation

In order to ensure that the application is ready for further work and understandable for potential future parties, the code base was written as self explanatory as possible, and encapsulated into different classes based on functionality, creating a universally understandable structure. To ensure that the Sheep-and-Go code base is comprehensible for further work by either the current developer or new unaware parties, explaining comments were added to each function during development. Sheep-and-Go was developed strictly following the guidelines of the used frameworks, which themselves were chosen partly based on their grade of recognition and popularity. Using popular and widespread framework ensures that future developers can access a larger amount of documentation regarding its use, thus get a better starting point for continuing the development at a later point. The use of widespread frameworks will also increase the life span of the application, as they are updated according to market standards. A considerable challenge during the product development was to make the application appealing to both children and adults. To achieve this, a lot of

thought went behind designing the interface. By following general design principles described in 5.4, the intention was to make the entire application understandable, rewarding, and exciting for everyone involved. The feedback from the questionnaire see 8.6.2, does however show that some of the test subjects had trouble understanding how certain functionalities were meant to be operated. In retrospect it could have been beneficial to conduct a separate test for the design of buttons and functionality explanations. It is worth mentioning that testing the application over the instant messaging software does skew with the user experience, which may also have been a factor towards misunderstanding of functionality. This is however not possible to know at this point in time without further testing.

10.2 Project Evaluation

Admittedly the largest concern with the project itself was the aspect of single handed development. The task of planning, creating, and testing an entire application as a single developer affected the project in both positive and negative ways. The absence of teamwork affects creativity and learning fostering, as ideas are not bounced and discussed between members with different or opposite opinions. This absence creates a tunnel vision during development, since all functionality is based on the view of one single person. To contradict the possible tunnel vision, I, the developer of Sheep-and-Go made decisions on behalf of research of the target users. In cases where decision between different design and functionality options had to be made, pros and cons of each option were noted and weighed against each other. This was done to create as objective decisions as possible, not affected by my own opinions. Even though working as a team is considered a more effective way to make a product, developing alone was a positive experience, with a lot of beneficial factors. First of all it was challenging to build and create an entire full stack system, not relying on anyone else, but myself. Being in charge of every aspect of the process made me responsible for all positive as well as negative occurring situations. This sometimes resulted in stress, when obstacles occurred or the code would not compile, but it over all created a rewarding feeling of self accomplishment. Secondly working alone enabled me to set my own goals and schedule, without having to rely on other people. For me, it resulted in a feeling of freedom to work towards my own goals, at my own pace. Over all I am pleased with

the entire experience from beginning to the end.

Chapter 11

Conclusion

11.1 Conclusion

The incentive behind the project of Sheep-and-Go was to create a mobile application with the intention of motivating children accompanied by their parents or guardians to spend more time outside in the Norwegian landscape, in contrast to the ongoing trend of spending their leisure time in the comfort of their homes. This motive is created by tempting with rewards gained in exchange for points acquired from observing sheep. A byproduct of the observation activity, is data of the location of the sheep. This data is considered very valuable for sheep farmers, which in effect will decrease the amount of resources they use to locate the sheep themselves. To help guiding the product in the right directions a set of goals were made in the early stages of planning.

11.1.1 Satisfaction of Goals

In order to measure the success of Sheep-and-Go it is important to know to which degree these goals were satisfied. The goals were first described in 1.3, and are below, repeated with concluding remarks.

Goal 1. An incentive for children to be more outside

According to the results from the questionnaire, see 8.6.2, the average test subject within the target group are motivated in a moderate, large or very large degree, to go outside. This motivation does either come from the

application itself, or by the possibility of receiving rewards. Judging by this data, the goal is considered satisfied.

Goal 2. Appeal to both children and their parents

According to results from the questionnaire, see 8.6.2, the average test subject within the target group would continue to use the application after their initial test. The median also shows a moderate, large or very large degree to which the genre enjoys the application, is satisfied with its genre and difficulty. The goal is judging by this data, considered satisfied.

Goal 3. Enable observation of sheep to be registered

The development of the application went according to the time schedule, with all main components completed. Together these main components enabled the functionality for registering the observation of sheep. Observation data consisting of geographical location coordinates with additional observational information is stored in the Cloud Firebase database system. According to the previously mentioned facts, backed by a successful testing phase the goal is considered satisfied.

Goal 4. Help to reduce resources used by the farmer with collecting sheep

As the sheep farmer is able to receive detailed location data for some of their sheep, it should in theory consequently narrow the extent of searching, thus resources spent in order to find or gather them. This is not directly proven in this project, as the tests were conducted over an instant messaging software, as a result of the pandemic described in section 8.4. There is however enough plausible cause for the goal to be considered partially satisfied, as it is possible to register sheep observation and share this data.

Goal 5. Help the farmer spot illness and strange behaviour in sheep

In similarity with Goal 4. no testing was conducted towards the ability to spot illness of a sheep. It is nevertheless a possibility as the needed functionality is implemented. A sheep farmer is from the app able to gather data regarding the whereabouts of their sheep. The location data can be

reviewed and analysed for patterns similar to the ones of ill sheep. The goal is therefore considered partially satisfied.

11.1.2 Summary

Three out of the five goals are satisfied, where as two are as a result of plausibility considered partially satisfied. These results indicate that Sheep-and-Go is indeed a motivation for children and adults to go outside, and that the data generated using the app can have a positive impact on sheep farmers. The application still has many points of improvements and reasons for further work, but can in its current state create both joy, and relief, and is consequently a success in the eyes of the developer.

List of Tables

1.1	Project Charter	5
4.1	Create account	21
4.2	Register observation of sheep	22
4.3	Deducting points in exchange for a real life prize	23
4.4	Gather sheep location data	24
8.1	Test subject information	68
8.2	Statement results	69

List of Figures

3.1	Gantt chart that shows the general outline of the project. . . .	17
6.1	Markers listed from left to right: Hot spot marker, Avoid marker, Sheep icons	33
6.2	Sheep-and-Go login screen	34
6.3	Sheep-and-Go register screen	35
6.4	Sheep-and-Go map view (default page)	36
6.5	Observation registration modal	37
6.6	Observation example	38
6.7	Observation tab	39
6.8	Prizes tab	40
6.9	Navigation bar with current location highlighted	41
7.1	Activity Diagram	46
7.2	Sequence diagram of the interaction between a child and Sheep-and-Go	48
7.3	Sequence diagram of the interaction between an adult and Sheep-and-Go	49
7.4	Sequence diagram of the interaction between a sheep farmer and Sheep-and-Go	50
7.5	Deployment Diagram	51
7.6	Client-server diagram	52
7.7	Navigation view	53
7.8	Class diagram of the map view	54
7.9	Class diagram of the observation view	55
7.10	Class diagram of the prize view	56

8.1	Paper prototype of the map view in an early version of Sheep-and-Go. The colored overlay indicates the current location within the application.	58
8.2	Paper prototype of the history view in an early version of Sheep-and-Go. The colored overlay indicates the current location within the application.	59
8.3	Paper prototype of a playroom view in an early version of Sheep-and-Go. The colored overlay indicates the current location within the application.	60
8.4	Paper prototype overlays to indicate the applications response to actions executed by test subjects	61
8.5	Extent of enjoyment	69
8.6	Extent of continuing use	70
8.7	Extent of motivation to go for a walk	71
8.8	Extent for rewards being the motivation to use the application	71
8.9	Extent of application genre being suitable	72
8.10	Extent of difficulty being appropriate	73
8.11	Extent of functionality being understandable	74
8.12	Extent of feedback satisfaction	74
8.13	Extent of application purpose	75

References

- [1] Jon Schärer. “Norge et utmarksland”. In: (Dec. 28, 2016).
- [2] *Nibio homepage*. URL: <https://www.nibio.no/> (visited on 02/03/2020).
- [3] *SSB: profitt*. URL: <https://www.ssb.no/jord-skog-jakt-og-fiskeri/artikler-og-publikasjoner/1-av-3-gardbrukere-uten-jordbruksinntekt> (visited on 10/04/2019).
- [4] *Role of youth in herding*. URL: <https://no.wikipedia.org/wiki/Gjeter> (visited on 05/16/2020).
- [5] *Monitor of Engagement with the Natural Environment pilot study: visits to the natural environment by children*. URL: <https://www.gov.uk/government/statistics/monitor-of-engagement-with-the-natural-environment-pilot-study-visits-to-the-natural-environment-by-children> (visited on 10/14/2019).
- [6] *Findmy*. URL: <https://www.findmy.no/> (visited on 11/13/2019).
- [7] *23% sheep loss*. URL: <https://www.findmy.no/nb/om-oss> (visited on 11/13/2019).
- [8] *Findmy: e-bjelle*. URL: <https://www.findmy.no/nb/produkt> (visited on 11/13/2019).
- [9] *Choremonster*. URL: <https://www.commonsemmedia.org/app-reviews/choremonster-make-chores-fun-for-kids> (visited on 11/13/2019).
- [10] *Geocaching*. URL: <https://www.geocaching.com/play> (visited on 11/13/2019).

- [11] *Groundspeak INC.* URL: <https://www.geocaching.com/contact/> (visited on 11/16/2019).
- [12] *History of letterboxing.* URL: <https://web.archive.org/web/20130725140150/http://www.dartmoorletterboxing.org/history%5C%20of%5C%20Letterboxing.htm> (visited on 12/11/2019).
- [13] *Pokémon Go.* URL: <https://www.pokemongo.com/en-us/> (visited on 11/17/2019).
- [14] *Niantic.* URL: <https://nianticlabs.com/> (visited on 11/17/2019).
- [15] *The Pokémon Company.* URL: <https://www.pokemon.com/us/about-pokemon/> (visited on 11/17/2019).
- [16] Mattias Skarin Henrik Kniberg. “Kanban and Scrum, making the most of both”. In: (2010).
- [17] Mattias Skarin Henrik Kniberg. “Kanban and Scrum, making the most of both”. In: (2010).
- [18] *Extreme programming.* URL: https://en.wikipedia.org/wiki/Extreme_programming (visited on 02/03/2020).
- [19] *The Waterfall Method.* URL: https://en.wikipedia.org/wiki/Waterfall_model (visited on 05/17/2020).
- [20] *Trello.* URL: <https://trello.com/> (visited on 01/19/2020).
- [21] *Fog Creek.* URL: <https://medium.com/make-better-software/fog-creek-is-now-glitch-8d0308aaf69e> (visited on 11/12/2019).
- [22] *Atlassian.* URL: <https://www.atlassian.com/> (visited on 02/04/2020).
- [23] *Overleaf.* URL: <https://www.overleaf.com/> (visited on 02/12/2020).
- [24] *Github.* URL: <https://github.com/> (visited on 02/04/2020).
- [25] *Microsoft.* URL: <https://www.microsoft.com/> (visited on 04/07/2020).
- [26] *Git.* URL: <https://git-scm.com/> (visited on 03/11/2019).
- [27] *React Native.* URL: <https://reactnative.dev/> (visited on 03/11/2020).
- [28] *Facebook.* URL: <https://www.facebook.com/> (visited on 04/07/2020).

- [29] *React JS*. URL: <https://reactjs.org/> (visited on 06/01/2020).
- [30] *Android*. URL: [https://en.wikipedia.org/wiki/Android_\(operating_system\)](https://en.wikipedia.org/wiki/Android_(operating_system)) (visited on 06/01/2020).
- [31] *iOS*. URL: <https://en.wikipedia.org/wiki/IOS> (visited on 06/01/2020).
- [32] *Firebase Cloud Firestore*. URL: <https://firebase.google.com/> (visited on 04/06/2020).
- [33] *Google Docs*. URL: <https://www.google.com/docs/about/> (visited on 10/11/2019).
- [34] *Unity*. URL: <https://unity.com/> (visited on 03/08/2020).
- [35] *UTHealth*. URL: <https://sbmi.uth.edu/nccd/ehrusability/design/guidelines/principles/> (visited on 05/16/2020).
- [36] *Google Maps*. URL: <https://www.google.com/maps> (visited on 10/19/2020).
- [37] *Statista*. URL: <https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/> (visited on 10/21/2020).
- [38] *statcounter*. URL: <https://gs.statcounter.com/os-market-share/mobile/norway/#monthly-201903-202003> (visited on 01/12/2020).
- [39] *Samsung*. URL: <https://www.samsung.com/us/> (visited on 06/01/2020).
- [40] *Windows*. URL: <https://www.microsoft.com/en-us/windows> (visited on 06/01/2020).
- [41] *Blackberry*. URL: <https://www.blackberry.com/us/en> (visited on 06/01/2020).
- [42] Philippe B. Kruchten. “The 4+1 View Model of architecture”. In: (Nov. 1, 1995).
- [43] *Three-schema-approach*. URL: https://en.wikipedia.org/wiki/Three-schema_approach (visited on 12/04/2019).
- [44] *Zachman framework*. URL: https://en.wikipedia.org/wiki/Zachman_Framework (visited on 12/04/2019).

- [45] *Kristiansten Festning*. URL: https://en.wikipedia.org/wiki/Kristiansten_Fortress (visited on 03/11/2020).
- [46] *COVID 19*. URL: <https://www.who.int/emergencies/diseases/novel-coronavirus-2019/events-as-they-happen> (visited on 02/04/2020).
- [47] *World Health Organisation*. URL: <https://www.who.int/> (visited on 02/04/2020).
- [48] *Skype for business*. URL: <https://www.skype.com/en/business/> (visited on 03/07/2020).
- [49] *Wizard of Oz method*. URL: https://en.wikipedia.org/wiki/Wizard_of_Oz_experiment (visited on 02/18/2020).
- [50] Bill Gillham. "Developing a Questionnaire". In: (2000).
- [51] *Sheffield Hallam University*. URL: <https://www.shu.ac.uk/> (visited on 04/11/2020).
- [52] *Google Elevation Service*. URL: <https://developers.google.com/maps/documentation/elevation/start> (visited on 05/21/2020).

Appendices

Appendix A

Invitation letters for user testing

Brukertest av mobilapplikasjon

Egil Uggerud
IT3906 Masteroppgave i informatikk
NTNU

29. april 2020

Hei,

Mitt navn er Egil Uggerud og jeg er for tiden midt i innspurten av min masteroppgave innen informatikk. I den anledning har jeg den siste perioden jobbet med å utvikle en mobil-applikasjon som skal hjelpe norske sauebønder med å holde oversikt over lokasjonen på sauene sine, og gjøre det lettere for dem å hente dyrene hjem etter avsluttet beitesesong.

Appen fungerer som en form for spill/bellønningssystem for barn i aldersgruppen 7-13 år, og gjennomføres i samarbeid med en foresatt. Ideen er at barn som er på tur med sine familier i skog og mark, skal kunne registrere observasjoner de gjør av Sau. En slik registrering vil bli belønnet med fiktive poeng som deres foresatte så kan bytte inn i fysiske belønninger (tenk iskrem eller 30 minutter med TV). Målet med oppgaven er å insentivere barn til å dra ut på tur i skog og mark, og på samme tid hjelpe lokale sauebønder.

Jeg lurte i denne sammenheng på om dere ønsket å bli med på brukertesting av min applikasjon! Denne testen vil grunnet dagens omstendigheter skje over Skype gjennom skjermdeling, hvor både barn og foresatte vil få testet de forskjellige funksjonalitetene applikasjonen inneholder. Data som blir produsert av disse testene skal inkluderes i oppgaven min, men alle personopplysninger vil forbli anonyme.

Kontakt meg gjerne på epost (egiugger@gmail.com) eller telefon (90644855) hvis dette kunne være av interesse, eller hvis jeg kan svar på evt spørsmål.

Hele prosessen vil ta om lag 30 minutter, og vil finne sted i begynnelsen av mai, på et tidspunkt som passer for dere.

Vennlig hilsen, Egil



Appendix B

Questionnaire

Tilbakemelding

Er du over eller under 18 år gammel?

- Over 18 år gammel
- Under 18 år gammel

Hvor mye tid bruker du på mobilspill i uken?

- Spiller ikke mobilspill
- 1-2 timer
- 3-4 timer
- 5-6 timer
- Mer enn 6 timer

Next

Tilbakemelding

Tilbakemelding til brukertest

I hvilken grad likte du å bruke applikasjonen?

- I svært stor grad
- I stor grad
- I moderat grad
- I liten grad
- I svært liten grad

I hvilken grad motiverte applikasjonen deg til å gå på tur?

- I svært stor grad
- I stor grad
- I moderat grad
- I liten grad
- I svært liten grad

I hvilken grad var vanskelighetsgraden passende for deg?

- I svært stor grad
- I stor grad
- I moderat grad
- I liten grad
- I svært liten grad

I hvilken grad var målet med applikasjonen tydelig?

- I svært stor grad
- I stor grad
- I moderat grad
- I liten grad
- I svært liten grad

I hvilken grad var applikasjonens funksjonaliteter forståelige?

- I svært stor grad
- I stor grad
- I moderat grad
- I liten grad
- I svært liten grad

I hvilken grad var du fornøyd med tilbakemeldingene du fikk etter å ha brukt funksjonalitetene i applikasjonen?

- I svært stor grad
- I stor grad
- I moderat grad
- I liten grad
- I svært liten grad

I hvilken grad var muligheten for belønning motivasjonen til å bruke applikasjonen?

- I svært stor grad
- I stor grad
- I moderat grad
- I liten grad
- I svært liten grad

I hvilken grad var det å hjelpe bønder motivasjonen til å bruke applikasjonen?

- I svært stor grad
- I stor grad
- I moderat grad
- I liten grad
- I svært liten grad

I hvilken grad var sjangeren på applikasjonen passende for deg?

- I svært stor grad
- I stor grad
- I moderat grad
- I liten grad
- I svært liten grad

I hvilken grad ville du fortsatt å bruke applikasjonen?

- I svært stor grad
- I stor grad
- I moderat grad
- I liten grad
- I svært liten grad

