

Tittelside

Skal minst inneholde:

- Navn på dokumentet (prosjektet)
- Gruppenummer
- Navn på gruppemedlemmer
- Info om at dette gjelder bacheloroppgaven ved HiØ/IT
- Sted
- Tid

Ingen krav på design/formatering, men bruk gjerne bilder eller annen grafikk... husk at sensor ofte vil se denne forsiden, og at den er med på å “selge” oppgaven.



HØGSKOLEN I ØSTFOLD

Avdeling for informasjonsteknologi

Remmen

1757 Halden

Telefon: 69 21 50 00

URL: www.hiof.no

BACHELOROPPGAVE

Prosjektkategori: Bachelor Thesis	X	Fritt tilgjengelig
Omfang i studiepoeng: 20		Fritt tilgjengelig etter:
Fagområde: Information Systems		Tilgjengelig etter avtale med arbeidsgiver

Tittel Map Management system for orienterings sport in Østfold	Dato: 15. May 2018
Forfattere: Jonas Vestgarden, Michael M. Simon, Adis Jasarevic	Veileder: Børre Stenset
Avdeling / Program: Department of computer science (Information Systems)	Gruppenummer: BO18-G34
Oppdragsgiver: Halden Orienteringskrets	Kontaktperson hos oppdragsgiver: Svend Sondre Mjølnerød Jens Erik Mjølnerød

Ekstrakt:

The use of digital technologies and its importance when creating, updating, sharing information has been vital for effective process and increasing performance of different fields.

Østfold orienteringskrets planned to avail technology aiming to facilitate a system for sharing orienteering sport map, enhance competence of the teachers and increase interests around orienteering sport in the schools.

3 emneord:

Web application
Agile
Orienteringskart

Forord

Dette er en mal beregnet til bruk i Bacheloroppgaven ved HiØ/IT. Malen gir en pekepinn om både ° struktur og innhold, og hvordan ting kan løses rent skriveteknisk, typisk ved å klippe og lime. ° Malen er utformet som dokumentasjon på et fiktivt prosjekt, der form ° alet er ° å gjøre det lettere ° og enklere å dokumentere en bacheloroppgave (og liknende prosjekter). De fleste kapitler er ° innledet med generelle retningslinjer for hva som skal med (dette er uthevet i grønt). ° Det er tenkt at malen skal kunne brukes i alle de ulike prosjektypene: utvikling, utredning og medieproduksjon. Dermed er mange overskrifter generiske, og må selvfølgelig tilpasses de enkelte ° prosjektene. Det kan også være aktuelt ° å sl ° å sammen enkelte deler av malen, eller legge til kapitler. ° Det er ikke obligatorisk å bruke malen.

Sammendrag

Sammendraget er hele rapporten komprimert til max 1 side. Sammendraget skal gi leseren et godt og tilnærmet komplett bilde av innholdet i dokumentet. Akademiske sammendrag kalles på engelsk for "Abstract", og i mer kommersielle sammenhenger heter det gjerne "Executive Summary". I det siste tilfellet har sammendraget som hensikt å gi ledelsen i en bedrift nok informasjon til å ta økonomiske og/eller administrative avgjørelser. . . uten å lese hele rapporten (!). Tradisjonelt blir sammendraget formattert som et sammenhengende avsnitt. I et bachelorprosjekt, vil hovedformålet være å gi leseren (kanskje i første rekke sensor?) et informativt (og appetittvekkende) bilde av prosjektet. Det er ikke vanlig å bruke litteratur- eller kryssreferanser i sammendraget. Som en regel kan vi si at alt som står i sammendraget, kan det leses mer om i rapporten. Dermed blir utfordringen å belyse alle viktige hovedpunkter, kort og presist. For denne rapporten, kan det f.eks. bli som dette: De nye retningslinjene for evaluering av bacheloroppgaver ved Høgskolen i Østfold/IT legger større vekt på hoveddokumentet enn før. Denne rapporten er resultatet av et prosjekt der formålet var å gi pupilene en mulighet for å forenkle og forbedre dokumentproduksjonen. Rapporten er en selvforklarende mal som tar for seg innhold, struktur og layout av hoveddokumentet i bacheloroppgaven. I tillegg er den et konkret eksempel på hvordan man kan bruke LaTeX som dokumentverktøy. Dokumentet er en mal, dvs. et stilsett som brukes for å gi dokumentet ønsket layout. Det blir gitt eksempler på de viktigste teknikkene, slik som bruk av kryssreferanser, kildereferanser, figurer og tabeller, og eksempler på formattering av spesielle elementer, som lister, sitater, definisjoner og matematiske uttrykk. I de tilfellene eksemplene ikke er selvforklarende, blir det gitt råd om hvordan man skal få det til. Intensjonen er at malen kan brukes for alle de tre hovertypene av bachelorprosjekter ved HiØ/IT: Utredninger, mediaproduksjoner, og utvikling av programvare, maskinvare eller systemer. Der det er naturlig å differensiere innholdet i de enkelte kapitlene, blir det skissert mulige løsninger for alle typene prosjekt. Formgivingen er enkel, oversiktlig og tradisjonell. Utgangspunktet for strukturen er den generiske oppbyggingen av et teknisk-vitenskapelig dokument, slik det er beskrevet i Mayfield Handbook of Technical & Scientific Writing. Innholdet i denne rapporten er en (kanskje forvirrende) blanding av generiske retningslinjer og konkret eksemplifisering relatert til prosjektet med å utvikle malen.

Takk Til

Det er vanlig, men ikke nødvendig, å nevne personer og miljøer som har hatt en positiv betydning for prosjektet, f.eks. på denne måten:

Jeg ønsker å takke gode kolleger ved Høgskolen i Østfold, Universitet i Oslo, og Høgskolen i Oslo og Akershus for interessante og fruktbare diskusjoner om utforming, gjennomføring og evaluering av bachelor- og masterprosjekter. I tillegg retter jeg en varm takk til pansermallene Ole, Dole og Doffen for uvurderlig støtte under arbeidet med prosjektet.

Innhold

Forord	3
Sammendrag	4
Takk Til	5
Figurliste	9
Table list	10
Chapter 1 - Introduction	11
1.1 The group	11
1.2 Project owner	11
1.3 Task description	12
1.4 Goals, deliverables og methods	13
1.4.1 Goal	13
1.4.2 Deliverables	13
1.4.3 Metode	14
1.5 Report structure	14
Chapter 2 - Terminology	16
2.1 Programming languages and data formats	16
2.1.1 Libraries	16
2.2 Framework	17
2.3 Design	17
2.4 Software Engineering	18
2.5 Orienteering	18
Chapter 3 - Project process	19
3.1 Overall process	19
3.2 Planning and Research	20
3.3 Methodologies	20
3.3.1 Design Thinking	20
3.3.2 SCRUM	20
3.4 Implementation of methodology	21
Chapter 4 - Analysis	23
4.1 Interview research	23
4.1.1 Interview research method	23
4.1.2 Purpose	24
4.1.3 Goal	24
4.1.4 Interview with stakeholders	24
4.2 Interview Analysis	25

4.2.1 Method	25
4.2.2 Coding and describing data	25
4.2.3 Results	26
4.2.4 Interpreting the results	27
4.2.5 Limitations of the interview data	28
4.3 Current solution	29
4.4 Technology criterias	29
4.4.1 Programming libraries	30
4.4.2 Database system	30
4.5 Application functionality from a user perspective	31
Chapter 5 - Design	32
5.1 Domain description	32
5.2 Interface	33
5.2.1 Design Requirements	33
5.2.2 Use case	34
5.2.3 Statement (hva alt skal gjøre)	41
5.2.4 Information Architecture	41
5.2.5 Design iterations	41
5.2.5.1 First iteration	41
5.3 Functional Requirements	45
5.3.2 User stories	46
5.3.4 Sequence diagram	46
Chapter 6 - Documentation	47
6.1 Application Structure	47
6.2 Javascript documentation	47
6.3 Database documentation	47
6.4 HTML/CSS documentation	47
Chapter 7 - Testing	48
7.1 Test plan	48
7.2 Test types	48
7.3 Test criterias	49
7.4 Results and measures taken	49
7.5 Second iteration of testing	49
Chapter 8 - Discussion	50
Chapter 9 - Conclusion	51
Register	52
Tillegg A	53

Figurliste

4.1 Layout for recto sider	16
A.1 The LATEX “bible”, 2. edition	25
A.2 Input and result from running the Douglas-Peucker line simplification algorithm [1] (from [2])	29

Table list

Chapter 1 - Introduction

In this part we will have an introduction about the content that this specific chapter contains.

1.1 The group

All of the group members are third year Information systems students with different interests and competence.

Name: Jonas Vestgarden | Birth: 1993 | Course: Informasjonssystemer | Location: Halden

Is a third year information systems student. Interests are programming, modelling and consultation. After a completed bachelor's degree, he will pursue being a consultant or a game developer, if not at the very least try both fields

Name: Adis Jasarevic | Birth: 1995 | Course: Informasjonssystemer | Location: Sarpsborg

Third year information system student at Halden University College. His interests are programming languages, database design, 3d modeling programs (maya and inventor) and as well as some marketing and music production. After a finished bachelor, he wants to specialize himself in machine learning and therefore in the future after the master's degree collaborate with his brother.

Name: Michael M. Simon | Birth: 1994 | Course: Informasjonssystemer | Location: Halden

Michael is an Information systems student. He's interested in user experience, Human computer interaction (HCI), Front-End developing.

He's currently managing his own private business which mainly deliver service in graphic design for print and digital. He wishes to further develop his business with a variety of services in web technology and continue to study on a master's degree at the University of Oslo and specialize in Interaction design.

1.2 Project owner

Østfolds o-krets is an active association hosting multiple official competitions and training competitions each year. They are a branch of the Norwegian orientation association and were founded in 1940 (Jens Erik Mjølnerød, 2016, s.7). They have an active and engaged community and wish for more young people to join them, which is one of the reasons this project was created.

Beskriv oppdragsgiver, både firma og kontaktpersoner, snakk med Svend Sondre for mer info.**NOTE:** Finnes ikke veldig mye info om østfold o-krets på nett.

1.3 Task description

Different schools around Østfold implements orienteering sport in their curriculums. The purpose of introducing the sport is to enhance maps and map reading ability of pupils by applying enjoyable and inclusive activities.

The purpose of the project (FinnFram i Østfold) is to mobilize physical activity among pupils and improve their map using skills. In addition, the purpose is also to extend interest and spread awareness of the World Cup Orientation for the upcoming years.

Currently the maps that are in use by Østfold O-krets are digital maps which they wish to provide to elementary schools located in Østfold to replace the older paper based maps.

The system was designed to solve challenges around map access and provide help to Østfold O-krets to share several activities and news. Actions such as saving and editing the maps will be available inside the platform with the other functions such as sharing and searching for maps and publishing news.

Furthermore, the system will also be used to draw control points (indicates hidden points) and other relevant information for the map. This functionality is suggested by the group members with a hope of reducing the use of third party applications for drawing control points and other information that are required in orienteering maps.

There are two options of the application platform. One is to build a web application that runs on the browser. A web application (also called web app) is almost the same as a desktop application. They offer very similar functionality, but the main difference is that a desktop app runs locally on your computer, whereas a web app runs on the server. Since the goal is to create a system that allow teachers to get away from using different desktop apps, this solution is very viable. The second option is to combine a web app with a light weight mobile version with different functionalities depending on the platforms performance, applicability and usability.

1.4 Goals, deliverables og methods

1.4.1 Goal

- **Goal** Develop a platform for sharing and drawing orienteering maps that will be used in different municipalities in Østfold country. Users of the platform will be able to;
 - **Subgoal 1** Upload, store, download, edit and print orienteering maps.
 - **Subgoal 2** Select a map based on location.
 - **Subgoal 3** Suggest activities and materials for supporting the learning process.
 - **Subgoal 4** Make maps available to specific user groups.

Simple and effective

The platform should be simple, intuitive and communicate a clear language.

User friendly

The platform should be developed of consideration to best practices regarding user experience, solutions and standards.

Accessible

The platform is expected to comply to web content accessibility guidelines and principles. The project will be developed following the guideline specified in web content accessibility guidelines(WCAG) 2.0.

Responsive design

The solution will be responsive and the platform will be accessed from mobile, tablets and desktop computers.

1.4.2 Deliverables

(Jobb Videre med denne delen)

The final application will be a platform containing functionality such as uploading, sharing and editing maps used by teachers and activity centers from Østfold.

This application will allow the teachers to have a common place to view and share orientation maps for areas in Østfold. The maps will be saved in one database and the system should be available for all teachers in Østfold.

Users can upload, share, draw (set annotations), print and retrieve saved maps.

Users can automatically generate annotations on the maps.

Users can edit individual annotations on maps already saved in the system by other teachers.

The following functionality will be present in the webapp:

- Register user
- Login and Logout
- Post orienteering map

- Draw points in the maps using the platform
- Print maps
- Admincontrol (post news and learning materials)
- Import maps that are made using other map drawing application.

1.4.3 Metode

Beskriv konkret hvilken utvikling metodologi som skal benyttes, hvordan skal gruppen skal komme frem til slutt løsning, hvordan design thinking prinsippet skal implementeres(detaljert om de fem steger i design thinking prosess- research,define,ideate, prototype,testing). Hvordan gruppen skal benytte scrum og levering av sprint, møter og arbeidsfordeling.

Verktøy: Hvilken verktøyer som er benyttet for å strukturere, utføre, kontrollere arbeidsoppgaver. (Trello, github).

1.5 Report structure

This document is structured based on the template delivered by HiOF.

Chapter 1 - Introduction: An introduction of the group members and their affiliations, the project owner and their organization, the task, goals and methods used to achieve the result.

Chapter 2 - Terminology: Different terminology used throughout the document will be explained here.

Chapter 3 - Project process: This chapter will go through how the group worked during the project period, what decisions was made and what they were based on and ultimately how it was implemented.

Chapter 4 - Analysis: An analysis of the research done prior and during the project as well as the interviews, and what those results were used for. What does the user need, and what did we need to implement to achieve this.

Chapter 5 - Design: The system design, process, implementation and related decisions taken based on the requirements specification.

Chapter 6 - Documentation: Documentation of the most important code written.

Chapter 7 - Testing: The testing that was done during development, the criterias for those tests, how they were performed and what the results were and used for.

Chapter 8 - Discussion: A discussion of deliverables, implementation, the solution, further work and maintenance.

Chapter 9 - Conclusion: The conclusion of the project.

Chapter 2 - Terminology

In this chapter we will talk about different definitions and terminology used in this document

2.1 Programming languages and data formats

The application is mainly written in Javascript using the MEAN stack in NodeJS.

JavaScript

JavaScript is an object-oriented programming language and is one of the foundations for most web development solutions, usually used in conjunction with CSS and HTML.

CSS

CSS stands for Cascading Style Sheets, and is primarily used to modify attributes and rules of HTML elements. This changes how a standard HTML page is displayed to the user and is a powerful tool when creating an intuitive and accessible solutions.

HTML

HTML stands for Hypertext Markup Language and is used to describe the layout of a web page. This is the bottom layer of a web page and is accessed by other languages by using the tags associated with the desired block.

JSON

JSON stands for Javascript Object Notation and is a text based, simple way to store javascript objects. The simple syntax is to allow human readability and easy access for javascript objects to be stored, and easily retrieved for later use.

SVG

SVG stands for Scalable Vector Graphics and is a XML-based file format to describe two-dimensional vector graphics. It's a basic part of HTML, but it can't be modified in the same way as other tags using CSS for example.

2.1.1 Libraries

SVG.JS

SVG.JS is a "lightweight library for manipulating and animating SVG." (<http://svgjs.com/>) It offers additional functionality otherwise unavailable in the standard svg format, to allow for easier customization, manipulation and creation of svg elements.

2.2 Framework

The framework used for the application is the MEAN stack in NodeJS, a platform built on Google's V8 javascript engine. It uses a NoSQL database called MongoDB to store data in JSON-like documents. MEAN stands for **M**ongoDB, **E**xpressJS, **A**ngularJS, **N**odeJS.

MongoDB

MongoDB is a open source NoSQL distributed database using JSON-like documents. NoSQL databases were made to deal with limitations in the SQL databases, mainly scalability, multi-structured data, geo distribution and agile development sprints.

ExpressJS

ExpressJS is a lightweight framework built on top of NodeJS to help organize the web app into a MVC architecture. ExpressJS har various features built in to help manage requests, routes, views etc.

AngularJS

AngularJS is an open source javascript framework developed by google. It is "what HTML would have been, had it been designed for applications". The current version and the one we're using, Angular 4, is written in typescript which is a language written on top of javascript. Most importantly what it features is the possibility to declare variables as types, much like in java and other object oriented programming languages.

(<https://docs.angularjs.org/guide/introduction>)

NodeJS

NodeJS is a framework built with the intent to provide an easy way to build scalable web applications. It's based on Google's V8 engine, which allows it to handle thousands of concurrent connections on a single thread.

V8

V8 (or Chrome V8) is a Javascript library written by Google for chromium based browsers. It's an incredibly fast engine as it compiles javascript code directly into native machine code prior to executing it.

Material Light

Material Light is a CSS library that is developed based on the android android material design principles. Material light is developed with a purpose of creating new interactions of user and elements by transforming the behaviour of real world objects and their natural features like shadow and distance into the digital world.

2.3 Design

WCAG 2.0

The web content accessibility guideline sets a standard for web contents accessibility. The guidelines contains twelve points that is a minimum requirement when developing a website that are accessible to different types of users despite their physical and cognitive ability.

2.4 Software Engineering

Scrum

Scrum is a framework for team collaboration and is used for addressing complex problems that can occur in a project. Teams that use scrum as development method develop in iterations to achieve goals.

Design Thinking

Design thinking is a user oriented designing strategy that can be applied when designing a solution aimed to solve a certain problem. The design thinking process contains five-stages; empathise, define, ideate, prototype and test, the team iterate through these stages when discovering and denoting the users pain points.

Unit Testing

Unit testing is facilitate testing of small parts of the application called units.

2.5 Orienteering

Orienteering is a kind of sport that can be performed combining map and compass for navigation. It's easy to learn, but always challenging. The object is to run, walk, ski, or mountain bike to different points displayed in the map, choosing routes, both on and off trail, that will help you find all the required points and get back to the finish in the shortest amount of time.

The points on a course are marked with orange and white flags and pouches or electronic devices, so you can prove you've been there. Each "control" marker is located on a distinct feature, such as a stream junction or the top of a knoll (Orienteering, 2018).

Orienteering. (2018). *About orienteering*. Hentet fra <http://www.orienteering.ca/about/orienteering/>.

Norsk Orientering. (2018). *Østfold*. Hentet fra <http://www.orientering.no/kretser/ostfold/>.

The sport demands the pupils ability to read maps and compass, teamwork and a coordination of physical and mental activity when performing.

In schools the sport is used as a tool for learning use of compass and map reading. The activity helps increasing social activities and enhance interaction between pupils.

Chapter 3 - Project process

This chapter elaborates how the team managed planning, execute the project plans and development process when working toward the goals, it also include the decisions the team took through discussions held with stakeholders, how the team scheduled, delegate responsibilities and organise the project using several practices and methodologies.

3.1 Overall process

Right after the team got the project, we focused on gathering necessary information related to orienteering sport and commence studying the domain, the purpose of the sport and the expectations of Østfold O-krets and World Orienteering Champion organisers (WOC).

The team held a meeting with WOC and Østfold O-krets.

Both parts explained their desire and what should be prioritised when building the application.

The team aimed to deliver a well functioning application that will be used for several years in the future. The project plan was exciting, but not achievable, considering the deadline of the project and estimated workload, the team discussed the matter with Østfold O-krets and came to an agreement to reconsider the complexity of the specification. Based on that, we received the second version of the specification after three weeks.

While waiting for the new specification the group worked on sub functions of the application, specifically the map editor function. The sample was published on frigg(hiof server), it was available for all stakeholders to follow up. The goal of making the sub functions available was to give continuous access to the project and that the stakeholders view, evaluate and give feedbacks on the progress of the application.

The sample application helped the team experiment several development tools, programming languages and select development environment and technologies that suit the application. There are assortment of possible methods, technology and tools that could be used, and making the sample application makes the decision process manageable.

Maintaining effective communication with stakeholders throughout the whole project is vital, the stakeholders involvement in the project was preserved and it was relevant for both the team and Østfold O-krets achieving the goals.

The team had a weekly meeting with our mentor. The responsibility of the mentor was to advice and guide the team. We also had continuous contact with Østfold O-krets representant.

We did a research on technologies, project management methods that are applied in the project. The team discussed which technology and methods that should be used and their practical effects on reaching efficiency and product quality.

As a part of the research the team had interviews that helped us understand users pain points, got an insight on how orienteering sport is applied in the schools, prioritise the

functionalities and support design decisions. The team also conversed with Lise and

The team attempted to work in agile following the scrum framework. The number of team members, the working environment and setting makes it difficult implementing scrum to the letter. After considering those factors, we choose to modify scrum by applying some of elements of scrum that are suitable.

Details on implementation of the above mentioned processes are discussed starting from chapter 3.2 and chapter four.

3.2 Planning and Research

Planning and research are responsible for putting the group on the right path, by creating different plans, such as strategic choices, and making important decisions in terms of sources and knowledge.

Here the group will take important steps. Based on these important phases, the group will gain insight into how they have planned research before making a choice and what kind of knowledge they have used and received before they have made a choice.

3.3 Methodologies

Methodology is a research strategy in which the person applies methods to a field of study.

3.3.1 Design Thinking

Design thinking helps mapping the users need and desire. It helps making complete design solutions of a product. By integrating the principles of design thinking the group attempt to draw out the functionalities required, understand the users need and develop the user interface. Most of the process was design thinking was performed using papers and pen before the start of developing the application and helped making the design of the application.

3.3.2 SCRUM

Scrum is a framework of a development process that can be used to deliver software product. Sprints are results that the project team produces in scrum, they are small fraction of the overall deliveries that are shipped in a one to two weeks of an interval and iterative development process. Scrum developing suits a team of three to nine persons, and the team members are delegated certain task focusing on three categories. These categories are scrum master, product owner and development team. Scrum increases the effectivity of project processes and development activities, but it may cause overlapping of roles in a team with a few members as ours. The team decided to implement the suitable elements of scrum and incorporate with projects tasks and processes.

3.4 Implementation of methodology

Hvordan gikk det, avvik osv.

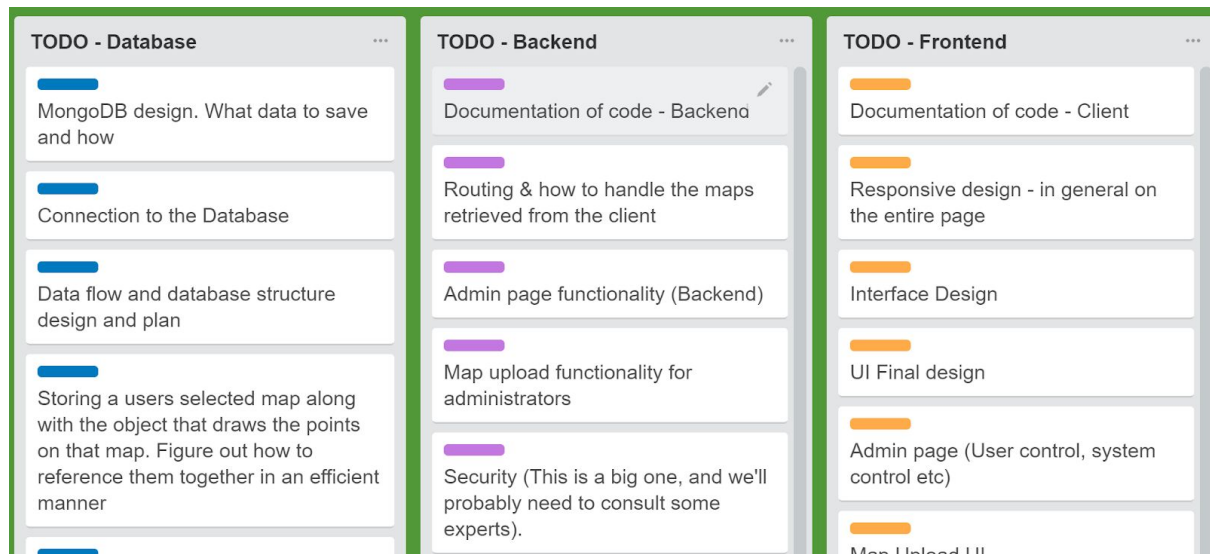
use of trello

backlogs and deliveries, delivery of sprints

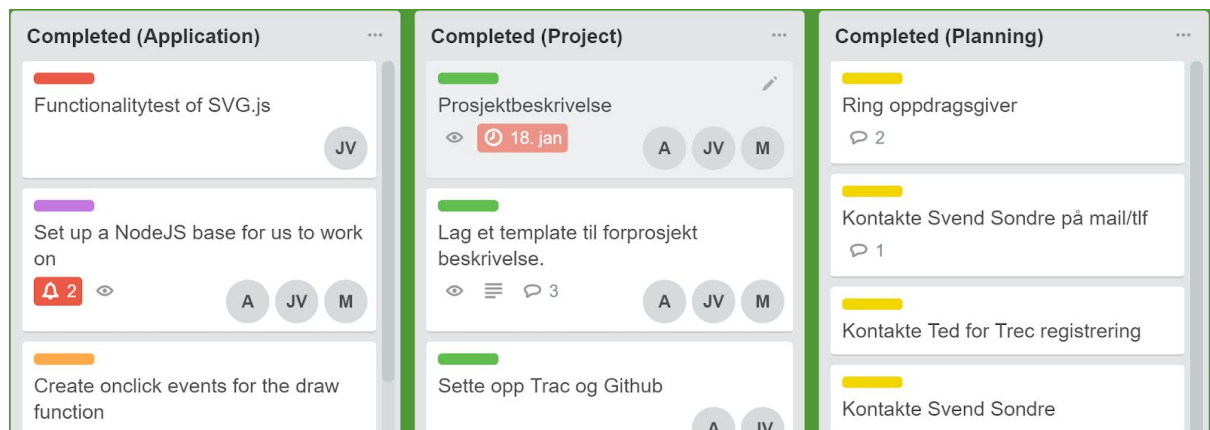
The team used a variety of tools. Some of the tools used are Trello, google doc, skype, facebook and discord for task management, documentation and communication.

Tasks were posted on the backlog board on trello, undergoing tasks and tasks that are completed were marked accordingly. The tasks were grouped based on the application structure categorized as back-end, front-end, administrative tasks and the database.

Any member of the team can select a task he desired, from the backlog on Trello, after the member started working with one task, he was expected to mark that task according to which phase and category that specific task belongs.



image>> Undergoing task examples from trello



The team had a weekly meeting with mentor where we discussed a variety of issues that occurred and resolved the issues, the group also used the meetings with mentor to inform the work in progress and completed tasks.

In addition to the meeting with the mentor the team had continuous meetings that remained until the project completed. The internal meeting lasts maximum fifteen minutes. The meeting was held up to three times a week. In these meetings the group discussed weekly plans, completed tasks, contacts with stakeholders. The meetings were also used to discuss a variety of issues a group member may encounter and to discuss solutions.

Meetings with Østfold O-krets

Chapter 4 - Analysis

In this chapter we will talk about the choices we took, which tools and libraries we decided on and what implications those choices have. There will be a qualitative analysis of the interviews done on the end-users, as well as an analysis of the technology used.

The purpose of this analysis was to further support our choices and decisions when creating the application.

4.1 Interview research

http://eprints.ncrm.ac.uk/3276/1/complete_proofs.pdf

4.1.1 Interview research method

Qualitative interview information

There are three main ways to perform a qualitative interview: Structured, semi-structured and unstructured interview.

As the names implies: A structured interview means having a set of questions, often on a form, for the interviewee to answer, leaving the questioner little room for modifying them. The purpose of a structured interview is to gain the same type of information from a large number of interviewees to compare the data together.

A semi-structured interview has a set of questions very much like a structured interview, but offers flexibility when asking the questions. The interviewer can ask the questions in any order they'd like, and follow up with their own questions to pursue a particular topic of interest. This allows the subject to answer in their own words, but the answers still have some semblance for comparison.

The last one, an unstructured interview, doesn't necessarily have a set of questions, but rather a topic or subject to cover. This allows the interviewees to answer in a way that is comfortable to them, allow them to reflect on their own ideas and are not influenced by the questions you ask. This is probably the hardest interview to hold as you as the interviewer need to keep the conversation going, while trying not to influence the interviewee when asking about a particular subject or topic.

For our purpose, we chose a semi-structured interview with a set of questions. A semi structured interview was more open to discussion than a structured one and we didn't need to worry about derailing the interview. Having a set of questions also allowed us to categorize the data such that we could draw comparisons with the results from other teachers and studies.

4.1.2 Purpose

End-users focused interviews was conducted with a purpose of discovering the users need, find out specific functionalities that users might desire that may not be considered when the specification was written by ØOA and also use the interviews to confirm and support the decisions that are made when designing the specification.

4.1.3 Goal

The goal of the research was to acquire relevant information for prioritising core functionalities of the application.

4.1.4 Interview with stakeholders

Interview with end-users and Østfold O-Krets

The focus group for user research was some of the end-users of the application, the teachers. **By maintaining continuous contact with user groups throughout the development and testing phase of the application modules.** The team was able to gather relevant information regarding user needs, usability of the application and functionalities.

Users were also involved when testing the application while the team build small modules and the final product. Building the contact from the beginning of the development and involve users through the process was vital as the team function following the guidelines of agile methodology.

Methodology

“In individual interviews, an interviewer talks with one user for 30 minutes to an hour. Individual interviews allow you to probe their attitudes, beliefs, desires, and experiences to get a deeper understanding of the users who come to your site.”
<https://www.usability.gov/how-to-and-tools/methods/individual-interviews.html>

By conducting One-to-one interview with end-users we were able to confirm the need for the application functionalities , acquire information around what problems the users are facing and suggestions from the users as to what the application can do to solve the problems they have regarding access to maps when having orienteering activities.

The teachers we contacted were willing to participate when the team test the application, the team aim to maintain a good contact and trust by involving stakeholders in the development process.

4.2 Interview Analysis

4.2.1 Method

A qualitative analysis method will be used to analyse the interviews with the end-users.

“Qualitative data analysis is a process that seeks to reduce and make sense of vast amounts of information, often from different sources, so that impressions that shed light on a research question can emerge.” (6 methods of data collection, page 13).

Since qualitative data is the opinions and interpretations of the subjects, we need a way to systematically group it together in a way that allow us to draw comparisons. One approach is called a “thematic network analysis”. is the process of “coding” all your data. Coding simply means you condense and categorize the data to sort the answers such that they can be compared against each other later (6 methods of data collection, page 14).

For our interviews, we chose to go with the second approach.

6 methods of data collection (source):

<http://www.open.edu/openlearncreate/mod/resource/view.php?id=52658>

4.2.2 Coding and describing data

Before we started coding the data, we had to index it to make it more manageable. Indexing is the process of applying “meaning to raw data by assigning key words or phrases”

(<http://methods.sagepub.com/book/keywords-in-qualitative-methods/n47.xml> source). In other words, picking the important parts of an interview. We then had a basis of comparison between the interviews, and could settle on categories that would describe certain aspects shared between the interviews. The following are the categories we decided on:

1. “How is the activity performed today?”
2. “What is required to perform the activity?”
3. “What works badly with the current implementation?”
4. “How can our product help the pupil?”

With category 1 we wanted to figure out what the current way of teaching orienteering to pupils was. Not necessarily to conclude that it was a bad way of doing it, but rather to draw inspiration from what works well, learn from the mistakes of what didn’t and make sure the changes we made weren’t too drastic. Ultimately it was not a team of engineers that were going to spend time teaching this activity, which means our application had to be easy to use and understand, and not alienate teachers who had a lack of experience with such technology.

With category 2 we want to know if there were any essentials we did not already know of that was part of the activity, and we could solve using the application. For example, since our application was purely a drawing, sharing and information tool, it was still required that the

teachers put physical markers at the spots marked on the maps. However are there any other requirements today that would also need to be handled in the application?

With category 3 we wanted to make sure we did not repeat problems that existed with the current solutions. Were there any ways our application could reduce or completely eliminate said problems?

With category 4 we wanted to know how our solution would contribute to ØOA's goal, that is teaching the pupils and raising awareness about orienteering. For example, giving the teachers an easier time to teach orienteering and collaborating between themselves would also contribute to the pupils learning more about the activity.

4.2.3 Results

With the data categorised, hereunder are the results of the interviews. These are the indexed versions of the answers written in Norwegian because the interviews were held in Norwegian, and translating them would subtract from the authenticity of the answers. The full transcript are available in the attachments:

How is the activity performed today?

Interview 1:

- Har en mappe med gamle kart de må bruke, med mindre dem lager sine egne
- Orienteringsklubben/Skiklubben har vært en stor del av opplæringen pga. utstyr og kunnskaper rundt emne
- Er vant til å tegne kart selv for hånd fordi gamle kart er dårlige, ofte med hjelp av google maps eller andre karttjenester på nett. Veldig tidkrevende.

Interview 2:

- Lærerne har gamle og utdaterte kart og noen poster. Utstyr er ikke det største problemet.
- Noen kart er ferdig markert.

What is required to perform the activity?

Interview 1:

- En bank med alle kart tilgjengelig hadde løst mange problemer med orienterings opplæring
- Trenger at undervisningen er litt "proff"
- Vil ha tilgang til alle områder i nærheten, vil ikke bli begrenset til kun skoleområdet, slik at man kan ta med seg klassen vekk fra skolen
- Lærerne har mye å gjøre, artikler og resources til å gå med kartene hadde lettet byrden dems veldig og gjort mye av arbeidet for dem

Interview 2:

- Enklere at lærerne koordinerer på skolen og at de har orientering samme uken. Blir mer naturlig for dem og bruke mer tid på det.
- Sikkerhet.

- Mer utfordring med tanke på passe vriene løyper og oppsetninger samtidig at det ikke blir for vanskelig for elever som ikke er fysisk sterke eller har funksjonsnedsettelse.

What works badly with the current implementation?

Interview 1:

- Lærere har lite utstyr tilgjengelig
- Lite kontroll på hva andre lærere gjør
- Ikke alle barn er interessert i fotball, derfor må gymtimene være varierte for å få frem talenter
- Har lite penger å bruke på utstyr

Interview 2:

- Tidkrevende pga mangel på oppdatering og forenkling.
- Lærerne eier 40 år gamle papirkart
- Noen lærere er flinkere enn andre når det kommer til orientering
- Løper samme løype i 5 klasse som i 7 pga. lite variasjon i kart

How can our product help the pupil?

Interview 1:

- Elevene synes det er gøy fordi det er noe dem ikke gjør veldig ofte
- Synes det er lurt av klubbene og satse, veldig mange barn er interesserte men det er dårlig lagt opp for øyeblikket
- Lærere blir mer motiverte av å få opplæring. Blir de flinkere selv er det morsommere og de gir bedre opplæring for barna

Interview 2:

- Alle elever som har orientering på skolen trives med det
- Å gjøre det lettere for lærerne gjør at de kan ha orientering oftere

4.2.4 Interpreting the results

By conducting this interview, we hoped to answer certain questions about current orienteering practices in the schools. The intent was to gain some insight into their current practices. Was there additional functionality desired that we did not already provide, how often were they performing the activity, what challenges were associated with it and could any of them be solved using technology? While most of these questions were already answered by our contractor, we realised the teachers might have different answers to the same questions. Ultimately the teachers are the main users of the application, which we considered important especially when designing the user interface and prioritising functionality.

The challenges surrounding orienteering does not seem to be exclusive to one school in particular, though certain issues are more prevalent in some than others. Both interviewees stated that the maps they had available were usually very old with posts already written on them. This brought some difficulties because the teachers wanted to change things up between the orienteering courses. It wasn't fun for the teachers nor the pupils to keep

running the same course every time, and since the pupils had at least one course each year, they ended up doing the same course in 5th grade as they did in 7th grade. This was one of the issues that contributed to less orienteering courses being held each year.

One interviewee stated that they had the option to find and create their own maps by selecting a location on Google Maps, downloading and drawing on that map manually. However they also said that the other teachers were not experienced enough with computers to do this themselves. After the interview they stated “most teachers here doesn’t even want to touch a computer”.

There was a huge disparity on knowledge about orienteering among the teachers. Both interviewees stated that while a few of the teachers knew a lot about orienteering, most had almost no knowledge at all. As a result there was a great need for additional resources to create these courses, especially when considering pupils with disabilities. They would like resources that explain best practices, rules and guidelines to follow when setting up a course such that it’s suitable for the pupils’ needs. This was another issue that contributed to less orienteering courses each year.

This was an activity the pupil loved in general because it’s not something they got to do very often. In most PE (Physical Exercise) classes the pupils play football and other general gym activities. This was mostly due to teachers not having the time to set up everything a orienteering course required. For this reason the teachers wished the tools were more readily available such that they could be better at planning, learning and teaching orienteering more often.

There are certainly issues the teachers has to deal with when teaching orienteering and while the application won’t remove all, it will at least deal with most of them. Having the maps all available in a digital format and being able to draw new control points on them, solves the issue of old paper based maps. Having resources tied to the different courses and readily available, solves the issue of teachers lacking experience when dealing with those issues. Being able to distribute the maps amongst themselves, allows the teachers to collaborate on the courses. Solving these issues should help raise awareness around orienteering, recruit more children and provide a greater variety in PE classes for the pupils.

4.2.5 Limitations of the interview data

The group was happy to even have a single teacher agree to an interview, though two interviews was not enough and there was a lack of data to analyse to come up with anything conclusive. The data retrieved was still useful to affirmed our decisions designing a user interface, and to verify the functional requirements. Further research was required to cover all aspects of user interaction, any desired functionality not currently implemented and additional resources for the users.

4.3 Current solution

Based on the interviews with the teachers we understood that the teachers faced a variety of challenges when trying to apply the orienteering sport into their PE classes. Among those, a limitation of access to orienteering maps and access to well established activity plans are mentioned. While we couldn't generalise all schools in Østfold based on the results from a few, there were indicators that most of them struggled with the same issues. This idea was further reinforced when talking with our contractor, project leader and other entities affiliated with ØOA, as well as other schools that declined our interviews because they did not teach orienteering.

As mentioned in the task description, ØOK already had digital maps available used for orienteering competitions. These were maps created by professionals using OCAD, which is a 3rd party application used for creating detailed orienteering maps (<https://www.ocad.com/en/functionality-chart/>). They were further edited in a second program called Purple Pen (<http://purplepen.golde.org/#about>) to place control points and control point information on them.

4.4 Technology criterias

A description of what led to us using certain libraries, frameworks etc. Things that led to this would be research done (reading about different ways to implement the solution), going through the requirements specification (ultimately the things we decide on needs to be compatible with the requirements), interviews done with the end users (teachers, clubs etc.), talking to experts on different technologies (lecturers in our school etc.) and continuously iterating on the requirements with the project leader. Community support of the programming languages, libraries and frameworks. Their ability of solving the problem. Time saving. Future popularity of libraries and maintainability. Availability of the technology. Easy to learn in short amount of time.

There are several factors that affected the selection of technologies used in the project, the group discussed and sorted out the relevant features and purpose each technologies can satisfy, to deliver a product that can be used for many years in the future.

Node.js is a JavaScript runtime built on Chrome's V8 JavaScript engine. Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient. Node.js facilitates the use of javascript in front-end and back-end of the application structure and built using c# and JavaScript on its core.

<https://nodejs.org/en/>

Using Node Package Manager(npm) developers can benefit from one of the worlds largest collection of libraries and resources that can be implemented in Node.js applications. There are over 1600 developers contributing to Node.Js on github and millions of application are built using Node.Js.

<https://foundation.nodejs.org/about>

- Scalability

The application built in this project can simply be upgraded by adding several functionalities. Node.Js allows efficient structure of scalable applications. It's very fast when working with JSON files and non CPU consuming applications.

The application is made by taking into consideration new functionalities that can be added in the future like messaging and can be further scaled if there is a need to serve different regions outside of Østfold. The use of MongoDB integrated with Node.js simplifies this process because there is no need for altering the database for storing a different form of data.

- Robustness
- Support

The Node.js and JavaScript community continuous growth guarantees that developers that are working in the technology benefit from the resources and support that can easily be found. There are several articles, blogs, active developers on stack overflow and similar services in addition to the official documentation that can be accessed on the web.

- Resources
- Developers knowledge and access

As mentioned above the Node.js, JavaScript and Angular community grow continuously and there are more developers involved and year after year. After the completion of this project, ØOA may not struggle to find developers that can be hired to further development of the application, when there is a need for update, maintenance and addition of new functionalities.

- Use Case

4.4.1 Programming libraries

Which libraries we chose and why

4.4.2 Database system

The database system chosen by the group is MongoDB. MongoDB is a no-sql database system that uses JSON like documents and schemas.

<https://en.wikipedia.org/wiki/MongoDB> (Ikke bruk wikipedia som kilde...)

The group chose MongoDB because it is often applied with node.js because of their shared asynchronous nature, and because of the ease of use of javascript JSON objects with the JSON based mongoDB document Structure. Other reasons is because of the plugin mongoose for mongoDB. Mongoose makes the development easier to model the application because it makes the process straightforward.

<https://stackoverflow.com/questions/12755961/node-js-with-nosql-or-sql> (igjen, finn kilder, ikke tull.)

Probably more sub-titles here to explain some more choices that doesn't fall into these categories

4.5 Application functionality from a user perspective

Functionality from a users perspective will always differ from our perspective as developers, and the project leaders perspective for what they want (Ref. to the what the * wanted etc. picture for instance)

Chapter 5 - Design

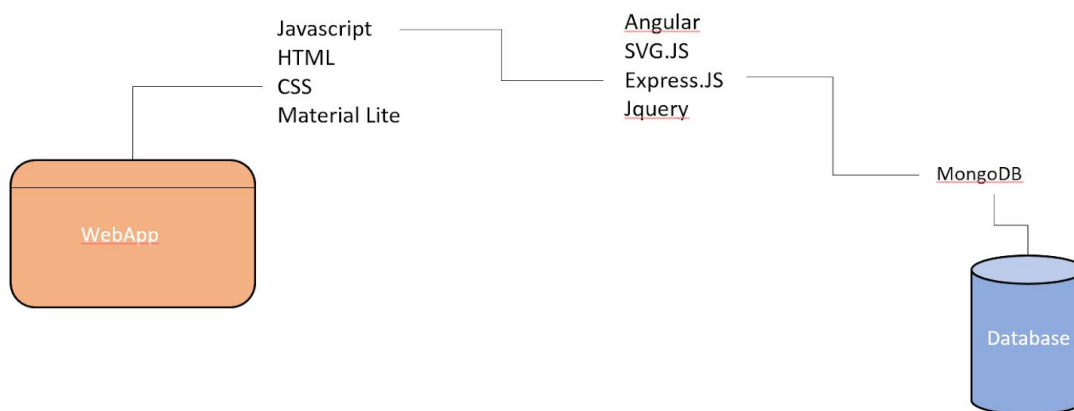


Figure nr : The figure above illustrates the general structure of the system. This section elaborates the design of user interface and interaction design decisions that was taken before and while the team worked on the project. The design went through an iterative process. Decisions were reviewed by the team and ØOK on each iterations.

5.1 Domain description

The application helps teachers, ski clubs and ØOA share and exchange orienteering maps, activities and related information.

ØOA is responsible for administering the application. The admin has access to all functionalities of the application similar to other users, In addition to the application functionalities the admin access registered users (add and delete) users.

Teachers will be able to download and print orienteering maps that are available only in their area e.g teachers that are registered from Halden schools will not access maps from Fredrikstad. This decision regarding limitation of access to maps were made by ØOA as specified in the specification.

Details stated in the table below illustrates different user groups' access to features of the application. The user groups are divided in three, these are ØOA, Ski clubs, and the schools.

Access	Østfold O-Krets	Ski ski center	Schools
--------	-----------------	----------------	---------

Upload maps	*	*	
Download maps	*	*	*
Print maps	*	*	*
Delete users	*		
Add new users	*		
Register		*	*
Draw points	*	*	*
Write and publish articles	*		
Save map courses	*	*	*

5.2 Interface

5.2.1 Design Requirements

User Interface color palette



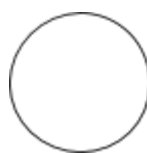
Background

Orange
#FF671F



Text and
Background

Black
#000000



Text and
Backgrounds

White
#FFFFFF



Positive
feedback

Green
#6AA84f








Warning

Red
#F00

Icons

<https://cdn.materialdesignicons.com/1.1.34/>

Used as	Icon	Description
Login		Appears when users are not signed in
Logout		User logout access from drawer
Map		A page where users access orienteering maps
Home		Main page of the application
News		Hyperlink that sends users to news page

z

5.2.2 Use case

Use Case 1 - Admin upload a map:

Goal in Context: Admin/sport clubs upload a map

Scope: Upload Page

Level: Sub function

Preconditions: admin has a map that are created and downloaded from third party applications.

Success End Condition: The map will be stored in database and it will be available to users.

Failed End Condition: The map failed to be stored in database and the user will be prompted with error message.

Trigger: Admin wish to upload new map.

MAIN SUCCESS SCENARIO

1. User clicked on upload button from the hamburger menu or header
2. User will be able to select the file from windows explorer or finder in MAC.
3. User click UPLOAD FILE
4. Admin/clubs will be notified if file is uploaded
5. The map will be available for users

RELATED INFORMATION

Priority: High

Performance Target: -

Frequency: -

Channel to primary actor: Web application

Secondary Actors: windows explorer or finder in IOS operating system

Channel to Secondary Actors: default file explorer of the operating system

EXTENSIONS

- 2a. Failure of operating systems
- 3a. Uploading may fail because of network failure.
- 3b. Uploading may fail if admin try to upload unsupported formats.

OPEN ISSUES

User should be able to get a confirmation of the uploaded file. The system should be able to check if the file is uploaded and confirm the event. Network problem/ error can occur between while uploading.

SCHEDULE

Due Date: Unspecified

Use Case 2 - Edit Map:

Goal in Context: Edit or add an orienteering course on the map

Scope: Edit map page

Level: Main function

Preconditions: A map uploaded in the system available in the users location

Success End Condition: Save/Print the map

Failed End Condition: Map or orienteering course unable to be saved in the database, or unable to be printed.

Trigger: User wants an orienteering track of a location.

MAIN SUCCESS SCENARIO

1. The user wants to draw an orienteering track on a map.
2. User selects the desired map
3. User draws the orienteering track on the map using the functionality present
4. User is finished and wants to print the map
5. User selects "Print Map" and is notified if they want to make the map public
6. Map is saved the system, and printed for the user
7. Map is available for other users to edit

EXTENSIONS

- 2a. The map is unavailable to the user
- 3a. User edits an already existing orienteering track
- 4.a User didn't finish drawing the orienteering track
- 5a. User decides to keep the map private
- 6a. The map failed to save or print
- 7a. The map is set to private and unavailable to other users

RELATED INFORMATION

Priority: High

Performance Target: -

Frequency: -

Channel to primary actor: Web application

Secondary Actors: Web browser's print functionality

Channel to Secondary Actors: The default window.print() function in javascript

OPEN ISSUES

The map a user wants might not be uploaded in the system.

The map a user wants might not be available in their location.

SCHEDULE

Due Date: Unset date

Use Case 3 - Admin Post article:

Goal in Context: Admin post edited articles

Scope: CMS page for admin

Level: Sub function

Preconditions: Articles should be edited using the rich text editor.

Success End Condition: The article will be stored in database and it will be available to all users.

Failed End Condition: The article failed to be stored in database and the user will be prompted with error message.

Trigger: Admin publish edited rich text -article.

MAIN SUCCESS SCENARIO

1. User click on publish article .
2. System gather all texts with their style features (Bold, italic, underline, new line etc..).
3. The article text will be stored into database.
4. User will be notified that the article is published successfully.
5. The article will be available to all users.

EXTENSIONS

- 1a. Storing text may fail because of network failure.
- 3b. Uploading may fail if admin try to insert symbols that are not acceptable by MongoDB.

RELATED INFORMATION

Priority: High

Performance Target: -

Frequency: -

Channel to primary actor: -

Secondary Actors: Server side functions for saving article text.

Channel to Secondary Actors: Functions exporting data from UI to backend.

OPEN ISSUES

Functionality of rich text editor will only be few and for that reason formatting of text, mathematical symbols and accessibility shortcuts using keyboard will be affected. ,

SCHEDULE

Due Date: Unspecified

Use Case 4 - Admin Edit article:

Goal in Context: Edit an article

Scope: Edit article page

Level: Main function

Preconditions: An article posted and saved in the system. The actor is an administrator.

Success End Condition: Article is edited and updated in the system

Failed End Condition: Article fails to save properly

Trigger: User wants to edit an article

MAIN SUCCESS SCENARIO

1. The user wants to edit an article
2. User clicks "Edit" on the desired article
3. User edits the contents of the article
4. User saves the article
5. Article is updated in the database

EXTENSIONS

- 4a. User didn't finish editing the article
- 5.a Article fails to save in the database

RELATED INFORMATION

Priority: High

Performance Target: -

Frequency: -

Channel to primary actor: Web application

OPEN ISSUES

SCHEDULE

Due Date: Unset date

Use Case 5 - User find map

Goal in Context: User search for map

Scope: Internal map search

Level: Sub function

Preconditions: Maps from database should be available to be indexed

Success End Condition: User will be able to view all maps based on the search key user inserted.

Failed End Condition: No search results available or user gets irrelevant results

Trigger: User search maps

MAIN SUCCESS SCENARIO

1. User search map using keyword
2. System checks maps on on keyboard key down
3. System display result
4. User will be able to view maps that contains the search key

EXTENSIONS

- 1a. No result with the inserted keyword

RELATED INFORMATION

Priority: High

Performance Target: -

Frequency: For each search

Channel to primary actor:

OPEN ISSUES

SCHEDULE

Due Date: Unset date

Use Case 6 - Admin edit users info

Goal in Context: Admin wish to change users information

Scope: Users in admin page

Level: Sub function

Preconditions: There should be users registered and displayed in the page

Success End Condition: Admin change user information

Failed End Condition: Admin failed to alter the required information of user

Trigger: Delete user, edit user info

MAIN SUCCESS SCENARIO

1. User search map using keyword
2. System checks maps on on keyboard key down
3. System display result
4. User will be able to view maps that contains the search key

EXTENSIONS

- 1a. No result with the inserted keyword

RELATED INFORMATION

Priority: High

Performance Target: -

Frequency: For each search

Channel to primary actor:

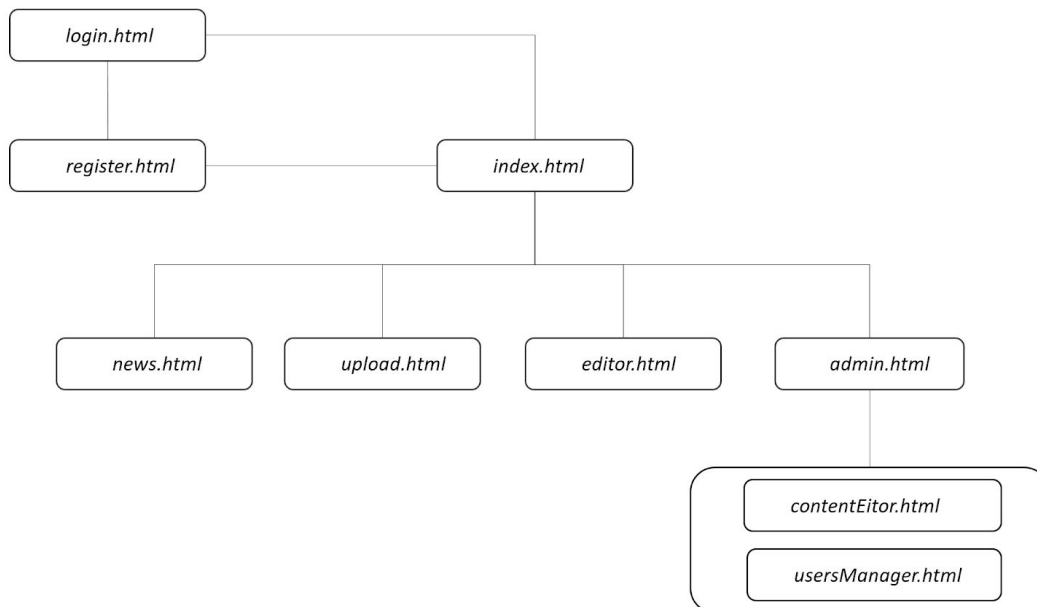
OPEN ISSUES

SCHEDULE

Due Date: Unset date

5.2.3 Statement (hva alt skal gjøre)

5.2.4 Information Architecture



5.2.5 Design iterations

5.2.5.1 First iteration

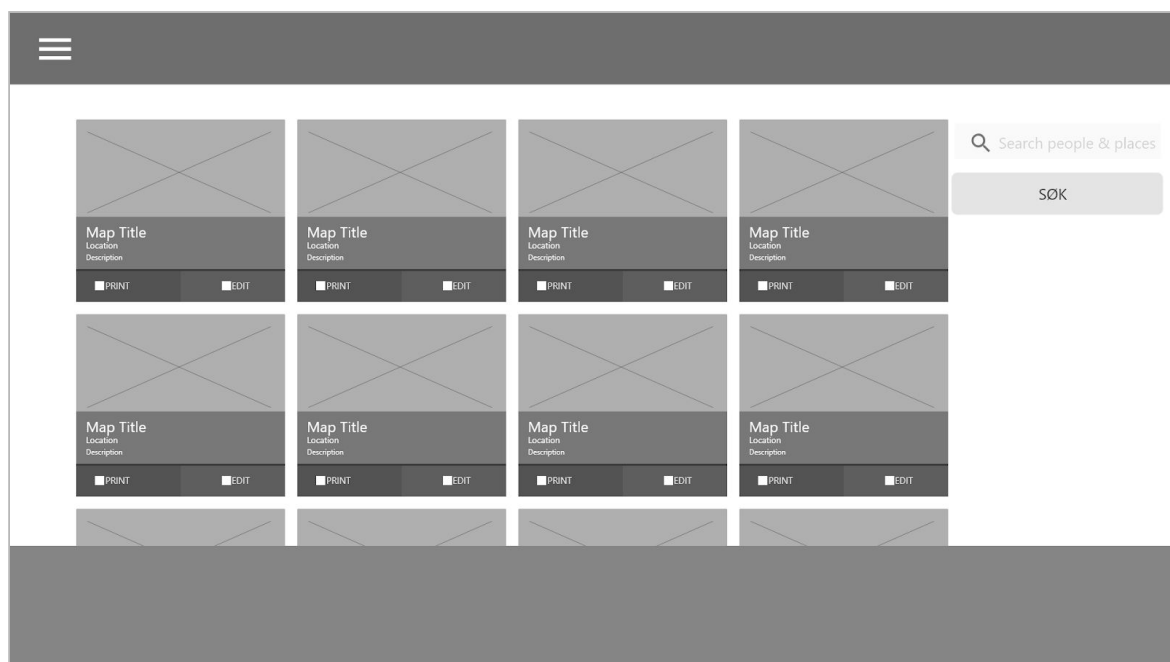


Image nr: Home page that display maps

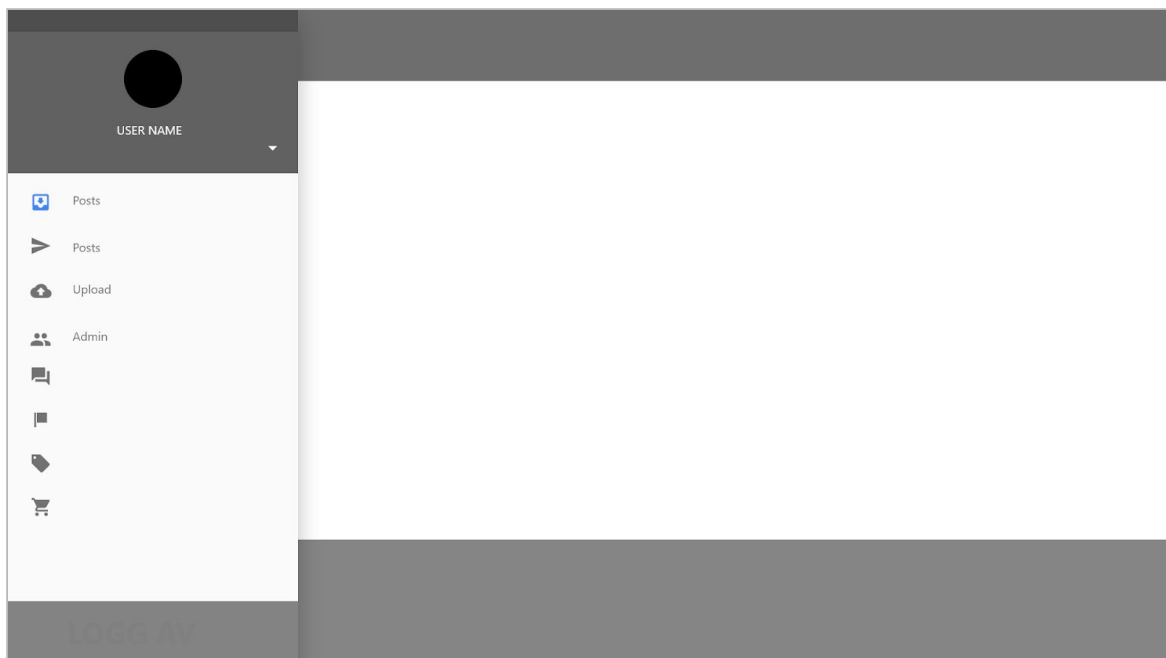


Image nr: Navigation menu design

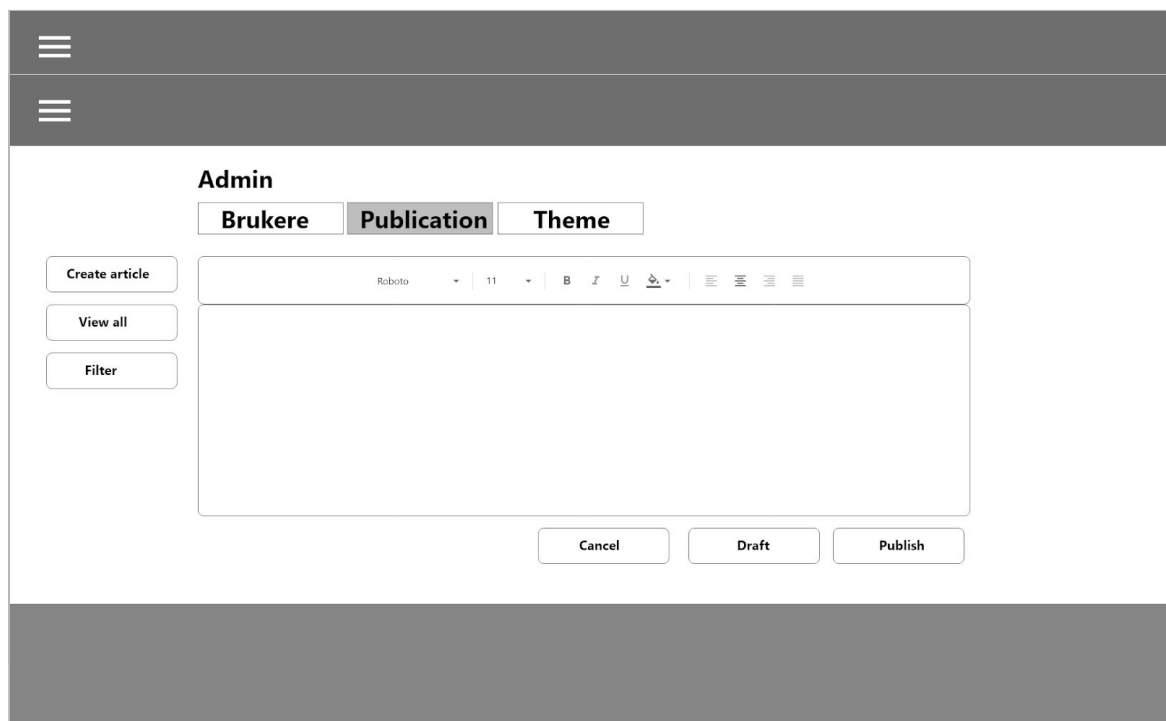


Image nr: administrator page gn

Image nr: Navigation menu design

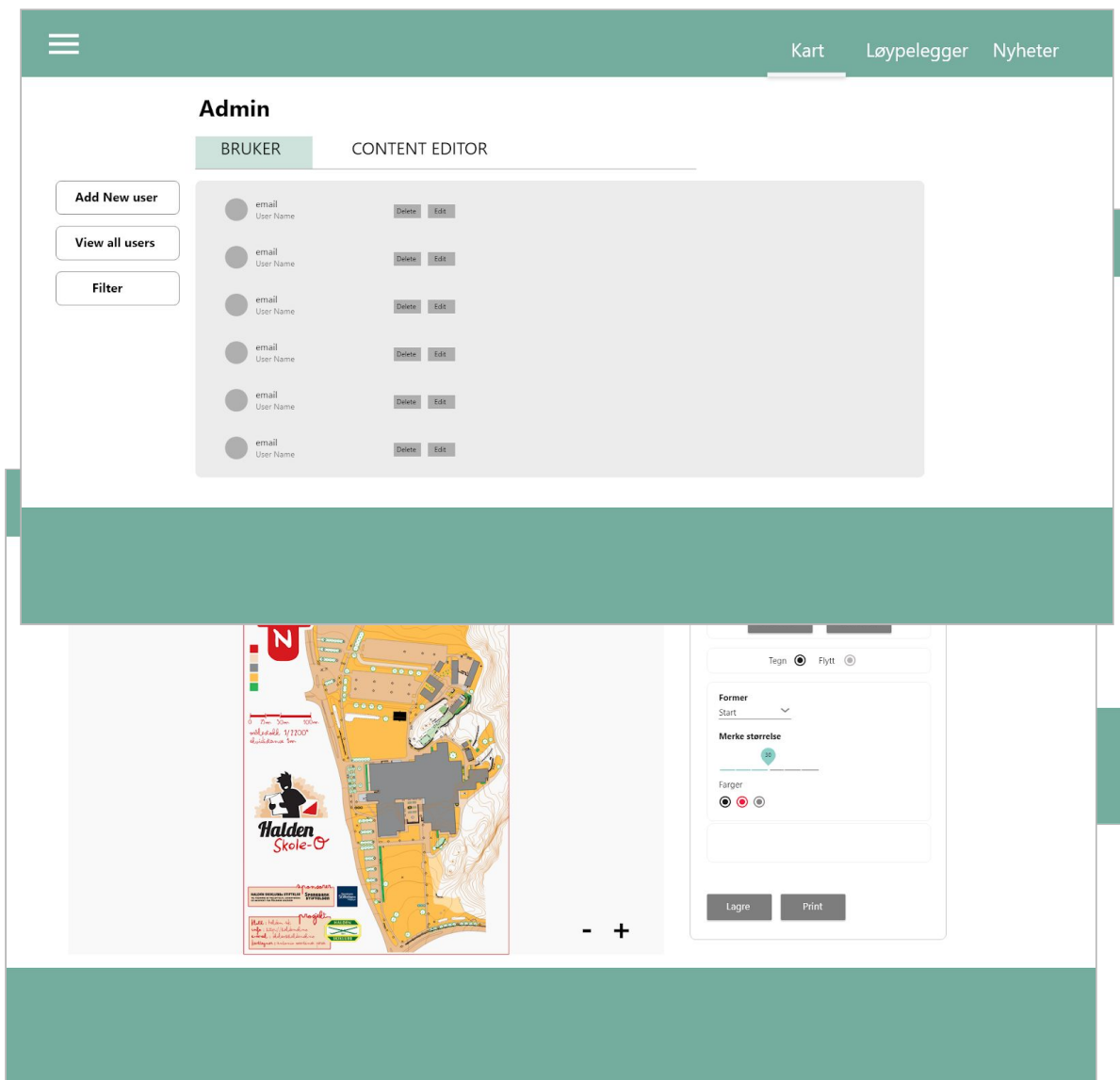
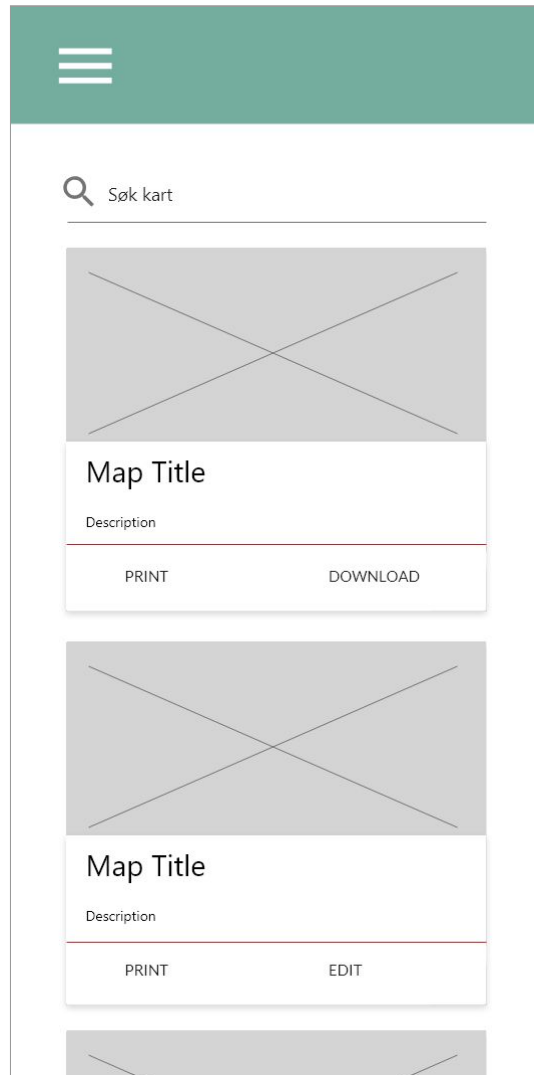
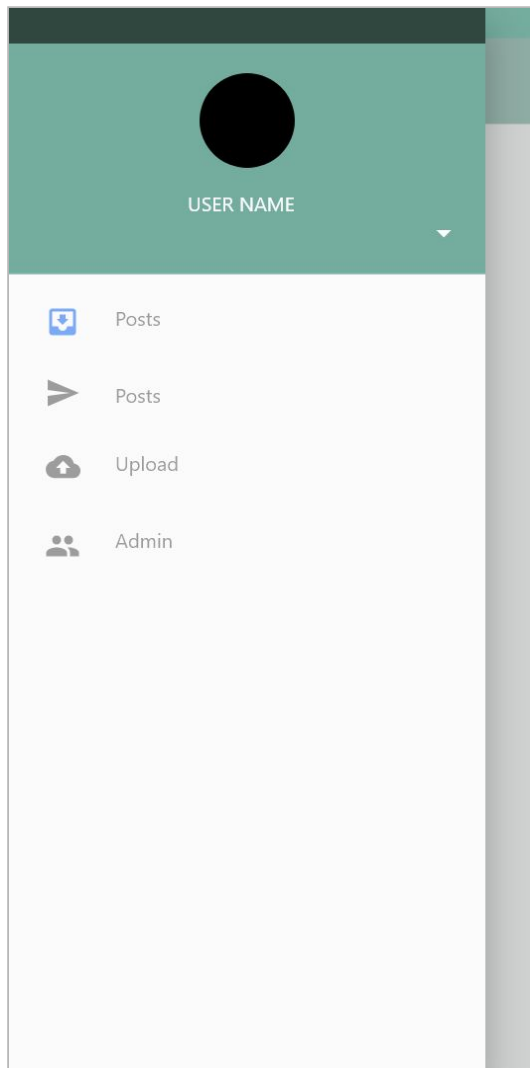


Image nr: Admin page

Mobile UI Designs



5.3 Functional Requirements

The functional requirements for the application are essential criterias and are expected by the group to be followed in the development process.

A functional requirement defines a function in the application. These functional requirements defines specific features the application has to do in order to present results of the requirement specification, or in other words what features it specifically is supposed to achieve in the requirement specification. In this project, the application has to have the opportunity for the user to edit and save maps, register and login and publish news-reports. So in order for this application to achieve what its supposed to achieve, it has to follow the features of the requirement specification.

On the other hand there are also non-functional-requirements which is known as requirements of quality that sets guidelines on design and implementation. Non-functional-requirements can be seen as the opposite of functional requirements and are the technical architecture of the application. (The application is the requirements).

Application

RequirementID	Requirement Statement	Must/Want
001	The application shall contain a list of maps the users can choose from.	MUST
002	The application shall have a search bar that the users can use to search the maps.	MUST
003	The application shall include a map-editor where the users can edit, save and print the maps.	MUST
004	The application shall contain a news publishing editor where the users can publish news and paragraphs.	WANT
005	The application shall have a register and a login function where the users can login as admin og teacher.	MUST
006		

001: The application system shall;

001.1: The application shall have list of maps

001.2: The application shall give the opportunity to the users to pick a map from the list.

001.3: The application shall make the user enter the map when the user clicks on the desired map.

002: The application system shall;

002.1: The application shall have search bar.

002.2: The application shall give the user the opportunity to enter a input in the search bar.

002.3: The application shall search for the users desired input and output the desired input.

003: The application system shall;

003.1: The application shall have a map-editor.

003.2: The application shall give the user the opportunity to edit the map.

003.3: The application shall give the user the opportunity to save the map.

003.4: The application shall give the user the opportunity to print the map.

006: The application system shall;

006.1: The application shall have text-editor.

006.2: The Text-editor shall have the opportunity to create html text.

006.3: The text-editor shall provide users with text-editing tools.

006.4: The text-editor shall give the users to publish/save paragraph or news in a database system.

005: The application system shall;

004.1: The application shall have a register function with forms.

004.2: The application shall have a Login function with forms.

004.3: The application shall give the opportunity to the users to register and login as a admin.

004.4: The application shall give the opportunity to the users to register and login as a teacher.

5.3.2 User stories

Unsure if we're including this yet

5.3.4 Sequence diagram

Unsure if we're including this yet

Chapter 6 - Documentation

This chapter will detail and go through the code most relevant to the applications functionality, for example the svg drawing tool.

6.1 Application Structure

How is the application built up

6.2 Javascript documentation

The most important parts of the self written javascript code in the application.

6.3 Database documentation

How is the database set up, and how are we storing our data

6.4 HTML/CSS documentation

How did we build and style the web pages

Chapter 7 - Testing

A part of design thinking is creating a prototype and testing it. We are combining that with scrum and iterative testing modules of the platform.

7.1 Test plan

Kommer om test plan, hva som skal testes, hva som ikke skal testes om det er tidsbegrensing.

7.2 Test types

Functionality testing

Check if each function of the web application suits the requirement specification intended for it. This is mainly involved as black box testing which is a method for examining the functions without looking into the source-code or the internal structure. The person testing the software is aware of what it is supposed to do but not aware of the mechanism behind it. In other words, the tester is aware that a input returns a output but not aware of the functionality.

Usability and accessibility testing

Usability testing or easy of use falls under the type “functionality testing”. The difference is that the testers are real users and that this type of testing method gives direct input. Its focus is to measure the user's behavior and action and see if it meets the requirement specification.

Accessibility can be tested using automated testers(validations), keyboard and The straw test.

Interface testing

This test requires three areas to be tested. These areas are application, web and database server. Testing the requests sent to the database and see if it outputs a correct display. See if the server is working correctly with coping with the application requests and not denying the service. Last and not least is the database server which is testing if the queries sent outputs expected results.

7.3 Test criterias

What are our criterias for performing the tests listed above? And why they are our criterias.

7.4 Results and measures taken

What we did with the results from testing in the first iteration.

7.5 Second iteration of testing

Same as above, might use under titles or something here. Basically what differs is changes we did to criterias from the first round, what new things did we find out, what issues persisted, what new measures did we take, what didn't work on the first iteration.

Chapter 8 - Discussion

Chapter 9 - Conclusion

Register

LaTeX, 25, 26, 28, 29, 33, 34

Recto, 12, 16

Recursion, *Se* Recursion

Tower of Hanoi, 32

Tillegg A