IS-304: 2020

Tittel: Tyvfiske2020

Emnekode:	IS-304
Emnenavn:	Bacheloroppgave i informasjonssystemer
Emneansvarlig:	Hallgeir Nilsen
Veileder:	Ilias Pappas
Oppdragsgiver:	IBM

Studenter:

Etternavn	**Fornavn**
Jul-Larsen	Kristian
Lindseth	Henrik
Rygg	Jostein Våga
Schjølberg	Joakim Høiland
Tjøstheim	Marius Johnsen

Jeg/vi bekrefter at vi ikke siterer eller på annen måte bruker andres arbeider uten at dette er oppgitt, og at alle referanser er oppgitt i litteraturlisten.	**JA X**	NEI
Kan besvarelsen brukes til undervisningsformål?	**JA X**	NEI
Vi bekrefter at alle i gruppa har bidratt til besvarelsen	**JA X**	NEI

Acknowledgement

We would like to thank IBM and their employees for welcoming us to their positive work environment and for allowing us to take part in your interesting project.

Special thanks to Morten Kamphaug for administering the project, choosing our group for this project, and selecting our IBM team while mentoring them to ensure that our counselors are of great expertise.

Special thanks to Bjørnar Kjenaas Mælum for honest and helpful feedback, as well as for providing great professional guidance throughout the project, and for being our contact at IBM for this project. We appreciate your guidance greatly.

Special thanks to Helene Kirstine Wekre for taking part in our project as the persona Karen and contributing with feedback as the customer, as well as other useful input Karen aside.

Special thanks to Bernhard Bakken for technical assistance and guidance in the development of our system.

We would also like to give a special thanks to our supervisor Ilias Pappas for professional feedback during the project.

Summary

In the spring semester of 2020, we came to an agreement with IBM and the University of Agder to complete our bachelor degree through a project provided by IBM. Our project was to develop a problem solving design and prototype which collects sensordata and processes it so that it can be used to analyze ship behaviour. The customer of the project is a third party, and one of IBMs customers in Norway, Forsvarets Forskningsinstitutt (FFI). This was our first project with stakeholders other than UiA, and this was an exciting and educational process.

For this project we have used the Scrum approach. Scrum is an agile approach, which means that our requirements and priorities changed during the development. We chose Scrum since this is a working approach which is familiar to us and is used by IBM. This helped us to learn about project management, both in terms of our experience, but also through the knowledge of IBM. To ensure high quality in our product we specified rules and criterias from the beginning of the project. This included making acceptance criterias for the user stories, testing and involvement of IBM the customer. We used user tests with the customer extensively in our approach to ensure quality of our product.

The project resulted in a Node-based web application that decodes AIS data collected from Kystverket, and visualizes these on a digital map from open source based MapTiler and OpenStreetMap. The system provides different filters and features to help the users to quickly analyze marine behaviour. We implemented all of the user stories that both we and IBM deemed of high priority. We also listened to our customer and focused on making a system that would best please the customer. The end-product is of good quality and fulfills the requirements in the project description.

This report consists of several chapters describing the development process of our bachelor project. The report is a detailed description of the planning, execution, and result of the project. It includes specifications and reflections of the system, and should provide an understanding of how and why the system works. In addition to the technical embodiment you will find reflection on project management and group dynamics.

Table of Contents

Table of Figures

1. Introduction

This report documents the work process of our bachelor project at the University of Agder, with IBM as our employer. This report is a documentation of the system development project, and will act as our final report in the course IS-304 and documentation for the product “Tyvfiske2020”.

The idea behind the project is to incentivise a solution for one of IBM's main customers FFI (Forsvarets Forskningsinstitutt). The product revolved around a digital solution to the Coast Guard work method. It is a map based solution displaying ships along the Norwegian coast line, at specific times. The name of the project is named “Tyvfiske2020”.

The incentives for us with this bachelor project was to learn how system development works at a corporate, and professional level, and how to incorporate different expertise into making a finalized product. With the project we wanted to learn about several different technological aspects regarding system development, both through guidance from IBM, but also in our own agile terms. We wanted to use this bachelor degree as a platform to elevate our knowledge regarding system development, both in terms of technical aspects, but also the processes that ensure a qualitative and structured product following an agile work method.

By starting a cooperation with IBM, International Business Machines, a leading international technology company with almost 400.000 employees spread over more than 170 countries, we were certain that what we wished for in the bachelor would be possible to achieve. IBM is represented in Norway with almost 1.000 employees and also delivers a broad spectrum of technology, products and services with a specific focus on cognitive technology and solutions, and cloud based solutions.

2. Product & Customer

Click the link for a product presentation: Tyvfiske2020

The product is a web based analyzing tool of marine traffic for the Norwegian coastline. The system effectivizes the workflow of analytics in the Norwegian Coast Guard, and simplifies old primitive analyzing processes, and eliminates previous tedious tasks in order to assess marine data efficiently. The product has a security login, but can be used by whomever when granted access. The goal of the system is to assist the Coast Guard in analyzing and detecting poaching in the Norwegian coast.

One of IBMs main customers at the date of the deliverance of this report is FFI. The Norwegian military force science institution. IBMs incentive for this project is to iterate on the project as a part of their innovation project towards FFI, and use it as a contention for FFIs collaboration with IBM. IBM wishes to use this project as a technology demonstration that enables good and innovative dialogue with the customer, in order to allow for creativity and new necessities. IBM wishes to use projects like this to iterate and inspire for real systems delivered to FFI.

In order for this to be possible IBM already created a persona for us, called Karen. A persona is a fictitious person created in order to help system developers to hone in on specific functions regarding work tasks in order to easily develop according to clients and customers. A persona is usually created through information gathering in terms of interviews, and questionnaires. But, by the persona already being delivered, we did not have to do this for the project.

The specifications for Karen was that she is an employee for the Coast Guard, and works with situational awareness for the marine coast line, and is responsible for the analytic of Barents sea. Two of her typical tasks are:

- Preparation of reports regarding the current situation(ad-hoc/daily content last 24 hours).
- Preparation of bigger situational briefing reports concerning concrete matter (comprehensive and detailed perspective, TimeFrame: 6-18 months).

Karen is regarded as the customer throughout the report of the project. Karen was a persona acted out by Helene Kirstine Wekre, an employee at IBM, in order for us to be able to consult and experience customer service when creating the product. This helped greatly in order to prioritize development changes and ensure quality within the project.

3. Key Decisions during the Project

This chapter provides a descriptive process throughout the project, including elaborations and consequences among the decisions made. Here we will go through the key decisions and their effects on the project. When working in a project, selection of tools and languages is central in regards to system development. Therefore this chapter will also include the major breaking points/properties that made us choose the tools that we did for the system development.

3.1 Methodology and project management

In terms of methodology we used an agile approach to execute the project. Because of the flexibility of the system development process we chose to use the Scrum approach. Scrum allows for changes undergoing the project. One upside utilizing this methodology is that we have experience using Scrum for system development from earlier in the bachelor's degree. Since we knew the base dynamics of Scrum, we found it efficient to use this approach. One downside of this was that we missed an opportunity to learn a new approach, however this decision emphasized effectivizing and improving our abilities both technical and methodological. Scrum is a popular approach and *"one of the most popular frameworks for implementing agile"* (Pletkus, 2019). In addition to being a familiar development process to the group, we were told at the kick-off that scrum already was an integral methodology at IBM. It was important for us that we chose an approach that our employer was familiar with so that collaboration with them would be as efficient as possible.

3.1.1 Scrum

The Scrum process is illustrated in figure 3.1. This illustration suggests a 30-day Scrum sprint, although for this project we decided to work in two-week sprints. Having relatively short sprints allow for frequent evaluation, followed by managing tasks at hand and to have many sprint plannings and reviews accordingly. We found two weeks best suited for our sprint plan. The pre-sprint spanned three weeks, due to extensive planning.

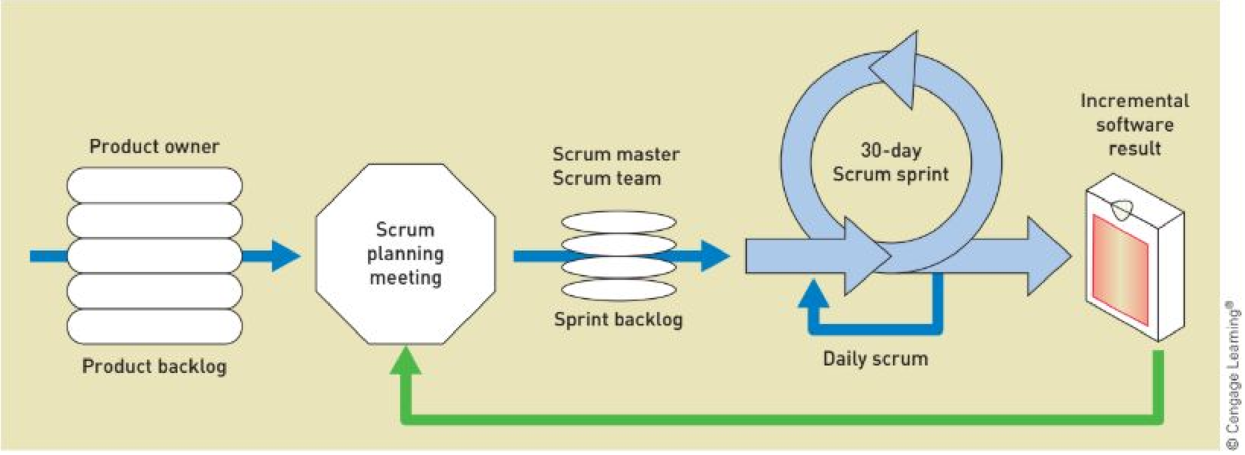


Figure 3.1 - Scrum software development process (Satzinger et al., 2016, p. 317)

At the beginning of each day the Scrum master held a daily standup meeting to account for Daily Scrum. Topics discussed during the standup were: What we had worked on, what to do that day, and challenges that we were experiencing.

At the end of each sprint we arranged a sprint review meeting to administer our Trello board and Gantt diagram to align new prospects. Trello is a web-based tool that let us organize tasks and schedule the project. The Gantt diagram visualized milestones, meetings and timeframe of tasks of the entire project. While we managed our product backlog through our Trello board, and used Gantt to help support and track our Trello Board. The sprint review meetings were directly focused on the specific tasks and milestones for each sprint and the project as a whole, thereby updating Trello and Gantt accordingly, our Gantt diagram can be found in Appendix A.

Another part of the review meetings included a retrospect where we reflected upon how the sprint had worked. Topics for discussion were often connected to the workflow, collaboration and communication. By reviewing our own work we got useful insights on the process and learned both from our mistakes and succeedings. With an adaptive mindset we were able to make changes to both work routines and scope and still keep the same pace and workflow.

When dealing with system development you have to regard that changes and improvements can occur due to technological advancements, therefore it is important to prepare for this with a retrospect that if there are changes to be made in the future, there has to be a set structure which enables for future development with the customer in mind.

3.1.2 Changes and challenges

Working as a group of five proves for challenges in terms of how to manage each group member regarding workload in an efficient way. Throughout the project we have had a semi-fluent workflow with each respective member having certain areas to fill in terms of work that had to be done. Throughout the project we have followed the Scrum methodology and adapted to that accordingly with how we worked. The biggest changes throughout the project has been that we all started out working on the product of the project, mainly front-end at the beginning, and back-end in around sprint 3. From there we changed the dynamic in which half of the group mainly worked on the product, while the other half conducted the documentation for the project as well as front-end solutions. An example of this approach was at the end of the project, when two of the group members focused on unit testing, the design oriented group member made the video presentation of the product, and the remaining two members focused on the documentation and the report. By managing tasks in this manner, we got more work completed, rather than having the entire group focus on each respective task at the same time.

During the project we encountered some technical difficulties. One of the roadblocks which stalled further implementation was to get the sensordata aforementioned by the project description into a database. IBM wanted us to integrate IBM cloud in the product, and we ended up using Cloudant; a database explained in chapter *3.2 Tools and Languages*. IBM provided the database in addition to related guidance in order to set up Cloudant and how to use it properly. We found it difficult to work as intended, and some technical assistance was necessary before we got a structured database with a connection to the map. This was a crucial part of the back-end to front-end part of the project, which we solved by guidance and guidelines provided by IBM.

The 11th of March University of Agder announced that all lectures would be postponed and all students were encouraged to stay away from campus. This was due to the ongoing COVID-19 pandemic which resulted in lock-down of the university, forcing us to make some changes in our routines. The announcement rolled out at the beginning of sprint 3 (figure 5.1), and in the middle of our project. In addition to closing the campus, UiA told their students not to visit their clients and customers during this period. Therefore the group met at campus each day of the sprint until the day of the pandemic announcement.

However our work with IBM did not change as a result of the aforementioned announcement. With IBM being located in Oslo, and the students located in Kristiansand, besides the initial kick-off event all of our communication has been performed online. This made working from home a smooth transition in regards to communicating with IBM. As mentioned previously, all the meetings with IBM were over Webex. This meant that the meetings continued as previously scheduled. The only change in regards to the pandemic was the environment around these meetings on IBM's side.

3.1.3 Risk assessment

Risk management is a part of the initial planning of a project. Risk management is useful to better handle possible conflicts and risks in the project, and ensure that recovery is possible if or when the risks occur. There are three major steps in risk management: Identification, assessment and control. Initially all the risks related to the project must be predicted and located. Next the different information assets and prospects in the project must be evaluated to what extent they are exposed to a present risk. Finally controls must be implemented into the project to avoid the risks and reduce the damage of the risks (Whitman & Mattord, 2016, p. 231). System components to consider while handling the risk management process are people, procedures, data, software and hardware. (Whitman & Mattord, 2016, p. 237). This was our focus point when assessing risks.

We used Google Sheets to create tables where we could include all the different assets, and get colour coded risk implemented to the table. In the table the assets are identified, assessed, and controlled. All group members attended the risk assessment and the probability and effect were discussed and categorized in unison. A risk team was formed for executive responsibility in setting controls to reduce the likelihood and damage of all risks the group discussed together. The risk team was made up of three out of five students. The risk team focuses on making different levels of controls to the risk assessment; managerial controls, operational controls, and technical controls (Whitman & Mattord, 2016, p. 204-205).

Table 3.1 illustrates how the group solved the risk management through the case of health issues. The probability and effect are ranked 1-5. Risk is the result of multiplying probability by effect, in this case resulting in the value 12. The orange colour indicates a moderate risk. The risk management team identified controls to help avoid the occurrence of data loss, and measures to reduce the damage upon incident. The measurement to avoid risk is an example of an operational control. The measurement to reduce damage includes both technical and operational controls, with a back-up solution implemented to physically secure the data on more than one platform (Whitman & Mattord, 2016, p. 205).

Cause	Probability	Effect	RISK	Measurement to avoid Risk	Measurement to reduce damage
Absence due to health issues	4	3	12	Eat healthy. Avoid contact from other sick people. If sick, do not go to school by risking others health.	Work from home. If possible.

Table 3.1 - Risk management clipart

The example shows 1 out of 19 assets we included in our initial risk matrix. The full risk management for the project is documented in Appendix B.

In order to get a better understanding of how well we handled our first risk matrix we decided to look back at the initial analysis, and re-evaluate all the assets at the end of the project span. This was arranged by the scrum master for the team to evaluate how well they calculated both probability and effect of a cause. The goal of this evaluation was to enlighten possible differentials in the analysis, and thereafter have the possibility to learn from the differences, and be better prepared for our next project.

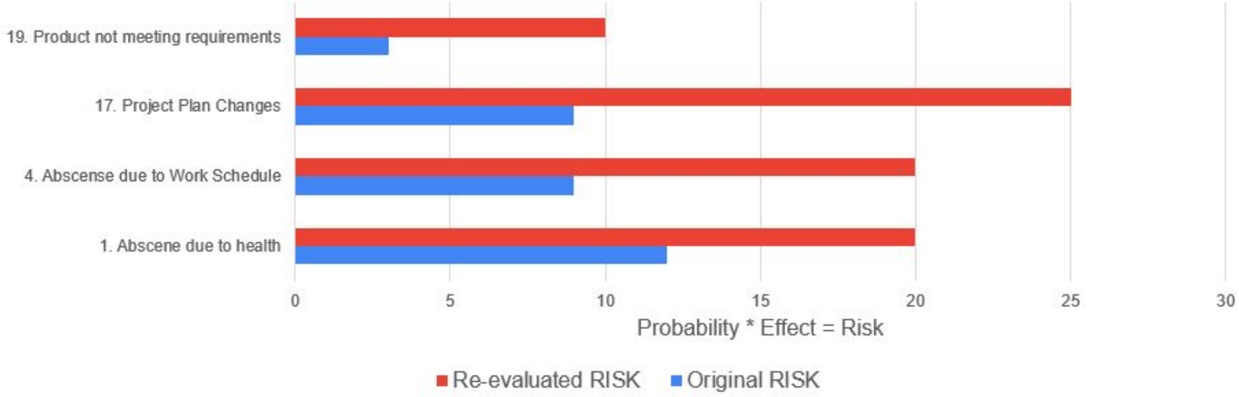


Figure 3.2 - Risk matrix comparison

Figure 3.2 illustrates the different risks we calculated in our original risk matrix versus the re-evaluated risk matrix we formed at the end of the project. Both of these are attached in Appendix B & C In the re-evaluation we went through the original risk matrix and made new calculations on the probability and effect of the different causes. The complete comparison is attached in Appendix C. The blue columns illustrate the original risk matrix, while the red columns represent how the re-evaluated risks compare to the original risks.

The red columns are noticeably higher than its associated blue column, meaning that we felt these risks were more present during the project than originally presumed. This is true for both cause 1 *"Absence* *due to health issues", and* cause 4 *"Absence due to work schedule"*. Throughout the project it is noticeable that for a longer part of the project we struggled with illness. All logged hours can be found in Appendix D. Absence due to both health issues, and work schedule, were more relevant than firstly assumed, especially combined. The reason behind increasing the risk is that the cause was more influential to the project than the original matrix suggests. This is also true for cause 17 *"Project plan changes"*.

Towards the end of the project, during a user test with our customer, we received several new wishes for functions to be implemented into the system. This caused a scope creep late in the project, and is the reason behind the red column being much higher, as we originally did not consider scope creeps well enough. That again is the reason behind cause 19 *"Product not meeting requirements"* being a little higher. It would be hard to fulfill all the customer's wishes with the right amount of quality, which meant we had to prioritize. That resulted in prioritizing and new consulting with the customer. All of these experiences are helpful for becoming more aware and being better prepared for future projects.

3.2 Tools and Languages

When developing the product we have been utilizing several frameworks. The advantage of using a framework is that it enables for a good structure, and easy deployment. It also makes for accessibility in terms of scalability at a later date. Frameworks are also popular within the development community, which makes for several sources to collect resources.

3.2.1 Leaflet

Leaflet provides all sorts of development utilities considering map based solutions. Leaflet has provided us with marker customization, data connection and has made it possible for us to connect AIS data to markers on a map, and also enabled us to customize the map in as many ways as the standard library provides, but also community made plugins. As stated by the library:

"*Leaflet is the leading open-source JavaScript library for mobile-friendly interactive maps. Weighing just about 38 KB of JS, it has all the mapping features most developers ever need. Leaflet is designed with simplicity, performance and usability in mind. It works efficiently across all major desktop and mobile platforms, can be extended with lots of plugins, has a beautiful, easy to use and well-documented API and a simple, readable source code that is a joy to contribute to.*" (2010-2019, Vladimir Agafonkin).

Leaflet has been our primary library when developing the product for the project. It has been our main way of representing data through map based solutions which this library enables for. It is written in Javascript, but because of its popularity is supported by many other languages therefore making it very adaptable, and agile in terms of what we have been able to achieve.

The documentation for Leaflet was easy, accessible and available. The documentation was provided by the Leaflet development team, in addition to a development community. Through the documentation we could find simple and complex solutions for our problems. This ranged from how to create a map, custom markers, and adding plugins to our markers and map.

3.2.2 OpenStreetMap

OpenStreetMap is the open source mapping solution which we chose for the project. We went for this solution because it is open source, which makes it possible for scalability, but also enables for other libraries to support it, such as our main denominator Leaflet. OpenStreetMap enables us to present map solutions through several different frameworks. It provides a visual representation of the world with accurate tile sets, and layouts which are customizable and scalable.

3.2.3 MapTiler

Maptiler is a map-hosting service which allows users to create and host maps for websites and applications as their service. After creating free accounts on MapTiler you can create keys that allow for hosting privileges to the hosting service with the different maps that MapTiler provides. IBM are some of the customers of MapTiler alongside NASA and the U.S Air force. This emphasizes the security and reliability of the service and was the reason we chose this service, as well as its user friendliness. Throughout the project our map has been stable and available.

3.2.4 Node.js

Node.js is an open-source server-side execution and rendering tool for JavaScript. It allows for easy implementation of other open-source libraries. It acts as the main web-server for the system, with connections to databases and APIs, and hosts its own APIs.

3.2.5 Bootstrap

By using bootstrap we were able to *"Build responsive, mobile-first projects on the web with the world's most popular front-end component library. Bootstrap is an open source toolkit for developing with HTML, CSS, and JavaScript."* (Bootstrap, 2020) It contains CSS- and JavaScript-based design templates for buttons, typography, forms, navigation, and other interface components. Bootstrap enabled us to style our buttons and icons and was a good choice since it has been used through popular websites and apps such as Twitter. Thus well documented and easy to use.

3.2.6 Cloudant

Cloudant is a database solution which is provided within the IBM cloud. *"IBM Cloudant is a document-oriented database as a service (DBaaS) that stores data as documents in JSON format. It is built with adaptability, high availability, and durability in mind, as well as a wide variety of indexing options that include MapReduce, IBM Cloudant Query, full-text indexing, and geospatial indexing. The replication capabilities make it easy to keep data synchronized between database clusters, desktop PCs, and mobile devices"* (IBM, 2020). Our system required fast storage and querying, and Cloudant could provide that with the right resources from IBM.

3.2.7 JQuery

JQuery is an open-source JavaScript library that is built for client-side operations. It provides many different types of functionality, from computing to UI-elements. It is also relatively light-weight and has good browser support, which means that it fits client-side deployment.

3.2.8 Jest

Jest is an open source testing framework for JavaScript. It provides us with good automatic reports containing test results and line coverage tracking. It also allows us to run tests on many asynchronous functions and assert that variables are correct through the whole process. Jest has good documentation and is a trusted framework for testing.

3.3 Architecture

Having an established system architecture allows the team to develop different co-dependent modules at the same time. The architecture diagram displays which modules that communicate with each other, and should give both developers and stakeholders a basic understanding of the information flow of the system.

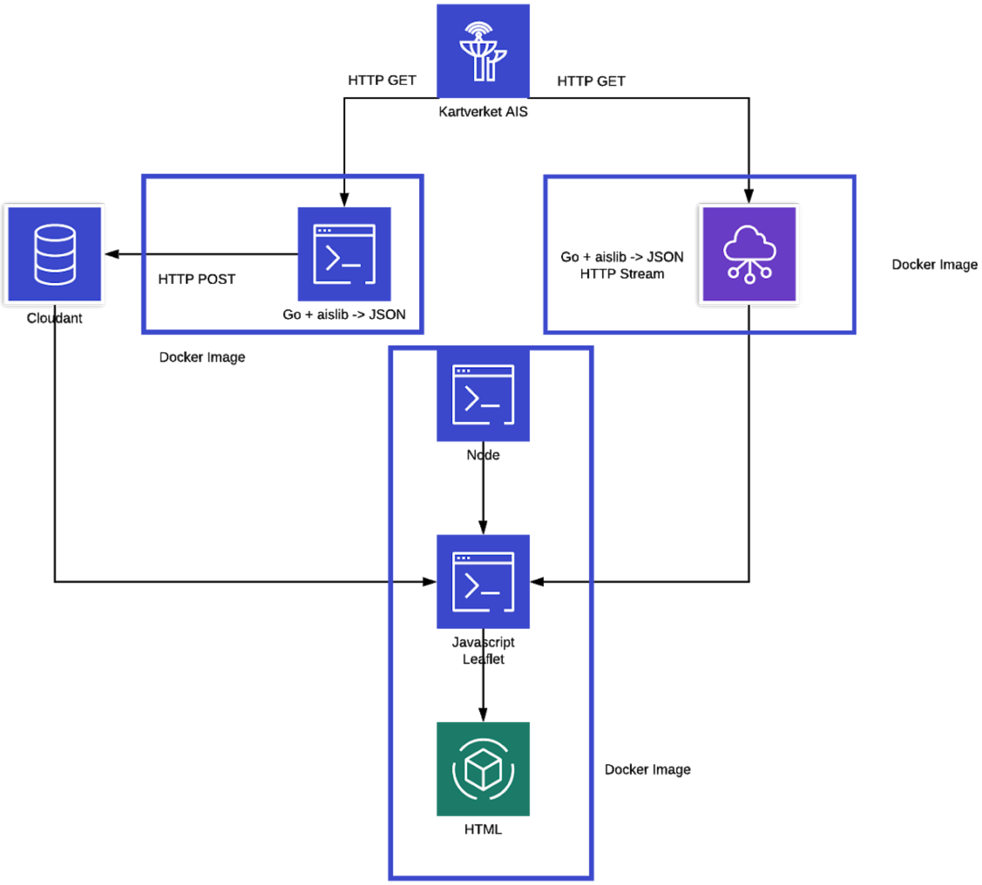


Figure 3.3 - Micro service oriented architecture

The system is built in a modular way, so that different functions can be written in the language that best suits them. This also allows for switching out different parts of the system without rewriting dependencies as long as the returned value is the same.
The system is built using three Docker-images

- One fetching AIS-data, decoding it, and publishing it as an HTTP API.
- One fetching AIS-data, decoding it, and pushing it to the database.
- One running the web-framework, with both back-end and front-end.

Using this setup means that things can be easily rebuilt and restarted without breaking dependencies. The architectural structure of the system is micro service oriented.

4. Quality of Project

The quality of the product has been a high priority throughout the project. This chapter explains some of the measures implemented to ensure high quality in the end-product.

4.1 Involvement of IBM

Bjørnar Kjenaas Mælum has been our contact at IBM, and he made it clear from the start that it was our responsibility to keep him updated on the process and envole IBM when needed. In our first supervision meeting we agreed to make a Slack channel for easy access with other employees at IBM with interest in our project. That made the involvement of IBM more efficient. We also used a file sharing tool called Box where we could easily share documentation and progress with IBM. We were encouraged to keep the Box library updated at all times. Box has version control and worked well as a common library. Mælum also granted our supervisor from UiA access to the Box.

Central decisions, such as choice of databases, user stories, and project progress in the project have been reflected and discussed with IBM, in the bi-weekly status meetings, on Slack, and the steering committee meetings. It was important for us and IBM to include them heavily in the project. With IBM being stationed in Oslo, the project management with IBM happened online. We feel like it would have been an advantage if we could visit them frequently in their office, but managed to have a fluent dialogue through Slack and mail. We are certain that the decision of heavy involvement of IBM has helped the quality of the end-product.

4.2 Acceptance Criterias & User Story completion

Acceptance criteria indicate the features that must be implemented in order for the user to be satisfied. Acceptance criteria focuses on functionality, rather than user-interface design (Satzinger et al., 2014, p. 72). Acceptance criteria are used to test the implemented features to see if it functions as intended. The acceptance criterias must be met in the testing for the user stories to be completed. We used acceptance criterias to make sure all functionality worked as intended in the user stories. As a developer it is helpful to have some criterias to work with when developing the system, so that we know what features to focus on. It also ensures quality in the end product, if no functionality is implemented without passing the acceptance criterias set by the development team.

Must have	Should have	Could have	Won’t have
1. Map ☑	5. Status ☑	6. Irregular behaviour alert □	10. Filter ☑
2. Zoom ☑		7. Destination ☑	11. Expected route □
3. Information boxes ☑		8. Ship type ☑	12. Time-frame ☑
4. Past movement ☑		9. Ship dimensions ☑	13. Calendar ☑
			14. Detect fleet □

Table 4.1 - MoSCoW table

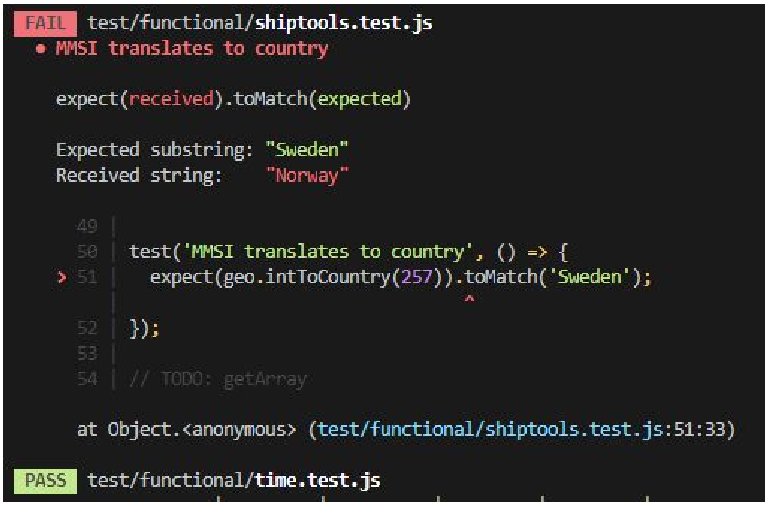
Table 4.1 illustrates the user stories in which acceptance criterias have been met. The completion of a user story is marked in the ballots, and includes all of the must haves and could haves. The user stories, including acceptance criterias can be found in Appendix E.

4.3 Testing

To ensure high quality of the product we made several user tests ourselves and tested the product with the customer, Karen. We got useful information on both user interface and the functionality of the system. Through the user test we found out how the user managed to navigate in the system, and how well the user executed the different functions and purpose of the system. We also found specific bugs and how well the system ran on the customers operating system. One example of a bug was that the system did not correspond well with the Safari Browser on the users Mac OSX. Even though we have several Macs internally in the development team, we had not yet tested the system on Safari, and it turned out that the markers did not appear on the map using Safari. This caused some confusion during the user testing before we localized the error.

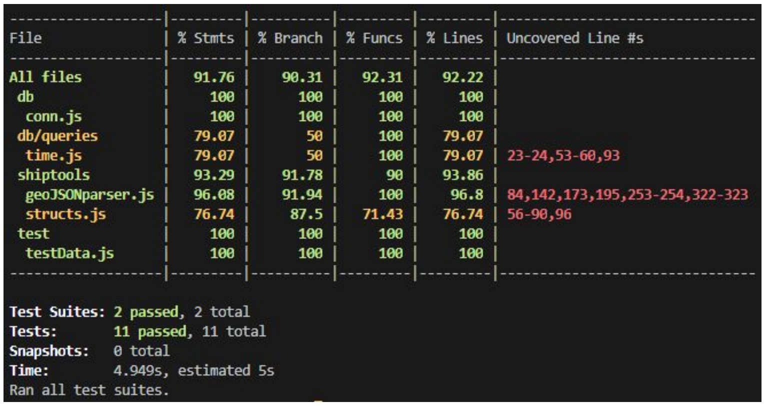
The user tests were important for us as a development team to better understand how the customer would use the system, and what adjustments needed to be done in order to make the system best suited for the customers needs. We asked ourselves what we could do to best help our customer, and the user tests answered these questions. In this project the most critical aspect was to make a functional and user-friendly system for the customer, and the success of that goal could be measured by the arrangement of the user tests. Therefore, we considered the user tests a mandatory asset to measure the quality of the project.

In addition to the user tests, we wrote unit tests as a part of quality testing for our system. The purpose of implementation of tests was to ensure that the system functioned properly. The software was tested by ensuring that specific inputs give the right outputs in different modules of the system. We used Jest as a testing framework, in order to get information about where tests failed or got broken, and coverage statistics, as illustrated in figure 4.1 and 4.2. We did not write tests in the first sprints, because we had to prioritise functionality, and did not know whether or not functions would remain in the system. We continuously tried to break the system during development with intentional wrong inputs, and tried to make the system safe by constraint in the front-end. We did this by simple replacement for testing in early versions. We have used a mix of unit and integration tests, with the biggest focus on integration tests, as some of the big functions are essential for the data flow of the system.



```
 FAIL  test/functional/shiptools.test.js
  ● MMSI translates to country

    expect(received).toMatch(expected)

    Expected substring: "Sweden"
    Received string:    "Norway"

      49 |
      50 | test('MMSI translates to country', () => {
    > 51 |   expect(geo.intToCountry(257)).toMatch('Sweden');
         |                                 ^
      52 | });
      53 |
      54 | // TODO: getArray

      at Object.<anonymous> (test/functional/shiptools.test.js:51:33)

 PASS  test/functional/time.test.js
```

Figure 4.1 - Example failed test



```
-------------------|---------|----------|---------|---------|-------------------------------
File               | % Stmts | % Branch | % Funcs | % Lines | Uncovered Line #s
-------------------|---------|----------|---------|---------|-------------------------------
All files          |   91.76 |    90.31 |   92.31 |   92.22 |
 db                |     100 |      100 |     100 |     100 |
  conn.js          |     100 |      100 |     100 |     100 |
 db/queries        |   79.07 |       50 |     100 |   79.07 |
  time.js          |   79.07 |       50 |     100 |   79.07 | 23-24,53-60,93
 shiptools         |   93.29 |    91.78 |      90 |   93.86 |
  geoJSONparser.js |   96.08 |    91.94 |     100 |    96.8 | 84,142,173,195,253-254,322-323
  structs.js       |   76.74 |     87.5 |   71.43 |   76.74 | 56-90,96
 test              |     100 |      100 |     100 |     100 |
  testData.js      |     100 |      100 |     100 |     100 |
-------------------|---------|----------|---------|---------|-------------------------------

Test Suites: 2 passed, 2 total
Tests:       11 passed, 11 total
Snapshots:   0 total
Time:        4.949s, estimated 5s
Ran all test suites.
```

Figure 4.2 - Test result

5. Project Execution

When working on a project of this size, collaboration is key in order to achieve the goals and wishes of our employer and customer. In order to achieve this we had a detailed plan and a culture within the group that promotes good discussion and work ethic.

5.1 Planning

During the pre-sprint we sat down to plan and discuss the workload and time management of the project. The most important task during this period was the kick-off, where we sat down with our employer and discussed their views and wishes for the project. During this meeting our employer and we set our expectations for the project. As mentioned previously, our project was to be split up into different sprints. The start and duration of these sprints were decided on in the pre-sprint, and can be seen in figure 5.1.

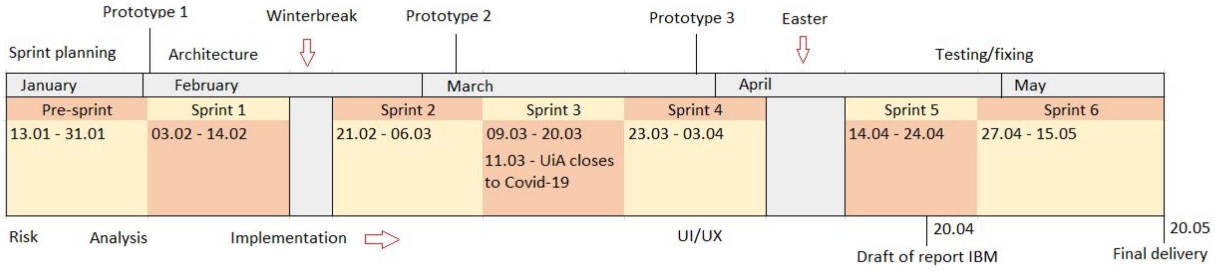


Figure 5.1 - Timeline

5.2 Analysis

When gathering information about how we were to develop the product we conducted interviews concerning the employees at the Coast Guard, namely our customer Karen. This was an efficient form of information gathering, because it allowed for discussions between the interviewer and interviewee. Furthermore it allowed for the interviewee to provide well thought out and fulfilling answers and wishes. The goal of conducting these interviews was to collect information about how the workflow of the interviewee was at that current time, and how our system could improve that workflow significantly. We wanted to collect as much information as possible on what specific areas would be most beneficial for the interviewee.

We wanted to figure out what works well and what is a complication with the current workflow, thereby asking what functionalities that would enhance, and efficiently increase the quality of workflow for the interviewee. Some examples of this was:

- Is there anything in the current system that you feel is missing?
- What do you think is positive about the current system?
- What do you find most negative with the system currently?

After collecting the information gathered from the interview we analyzed the answers in order to structure them into user stories. We used the prioritization method MoSCoW to structure the importance of the user stories. We shared the user stories with our employers at IBM, and had discussions with them on what the smartest prioritization would be. The user stories are attached in Appendix E.

Throughout the project there have been occurrences where we had to evaluate how important or unimportant some of the user stories where. For instance, in a meeting with IBM they issued that there were a small amount of Must have user stories and that the prioritization should be increased in quantity. During the project we have had meetings with the customer discussing how the customer wants the system to be. With these meetings we have had scenarios where we have encountered implementation which did not demand user stories. This was because most of the user stories, at the later stages, were mostly completed. Proving that our user stories performed as great building blocks for the system, enabling for agile implementation of functionalities.

5.3 Design

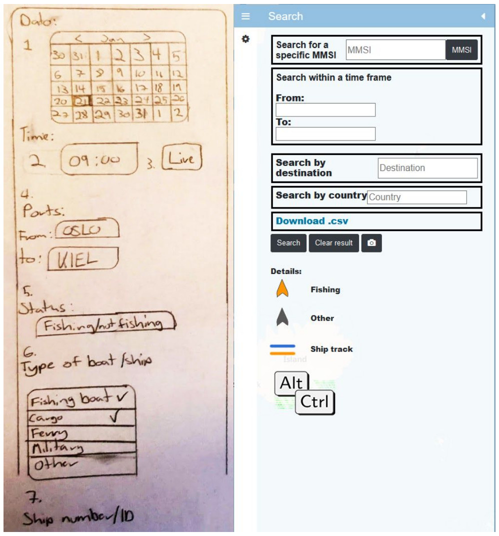
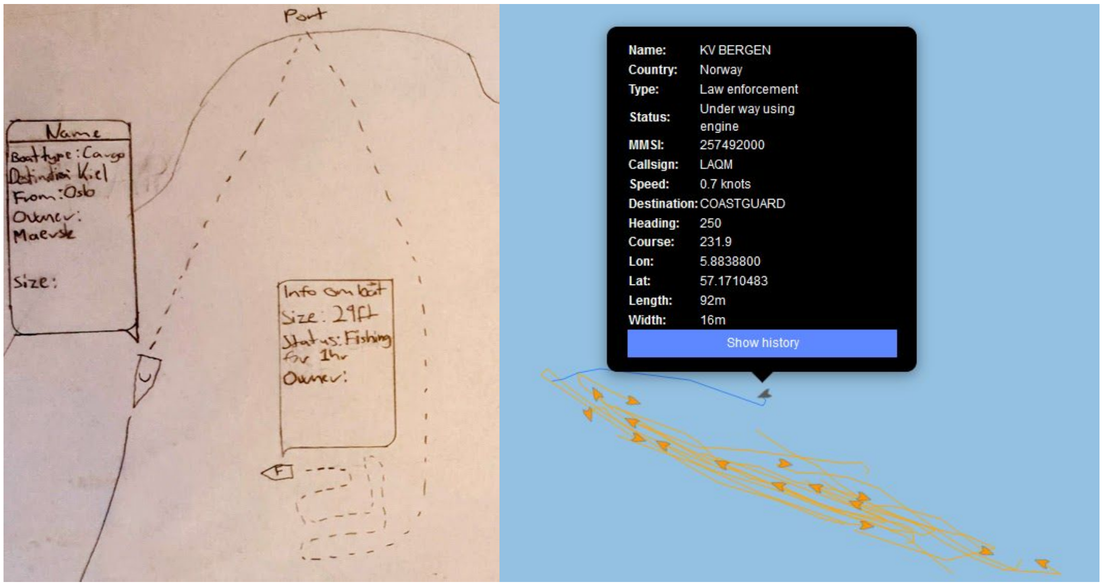


Figure 5.2 - Wireframe vs. menu design

When it comes to the design, we started out by making simple sketched wireframes. These wireframes were based on both the group's thought and what we learned from our initial meeting with our employer. These wireframes served as an initial design that laid the groundwork for our prototype development. The prototypes were presented to our customer. This was very beneficial as our customer could give us constructive feedback, and further wishes.

The figures on this page shows the wireframes and how the final product corresponds to the wireframes. The wireframes were thoughtfully designed to match how we realistically envisioned the end product. The wireframes include functions as they appear in the user stories, as well as being guidelines on how the front-end could be done. We designed the product to be user friendly as well as pleasing to look at, and collected feedback from the customer in design oriented user meetings. The result is a product with a high amount of detail and smart solutions.



Name: KV BERGEN
Country: Norway
Type: Law enforcement
Status: Under way using engine
MMSI: 257492000
Callsign: LAQM
Speed: 0.7 knots
Destination: COASTGUARD
Heading: 250
Course: 231.9
Lon: 5.8838800
Lat: 57.1710483
Length: 92m
Width: 16m

Figure 5.3 - Wireframe vs. map design

5.4 Implementation

The development of the system has been a central part of the bachelor project. We chose to work with Golang and JavaScript as we have worked with these languages before and they have good documentation and support. We chose Golang for simple back-end tasks as it is lightweight and runs fast compared to industry standards like Java.

As soon as we knew what data we could get from the AIS, we agreed on a common format for the entire system, so that different members could work on different parts of the system at the same time. This allowed everyone to work with the languages and functions they felt most efficient with and allowed us to complete components which rely on each other to function.

When we first started implementing, we did not want to commit to any frameworks because that could hinder us in the future. Instead, we implemented a lot of features as stand-alone and adapted them to the frameworks when we were sure it fit the project's needs. None of us were experts at JavaScript, but some of us had used it before, and we were familiar with the syntax, so the whole team was able to implement features using the wide variety of official and unofficial documentation.

5.5 Time Management

Considering that this is a bachelor’s degree, hence a project which does not consist of paid hours, time management was an important factor nonetheless. This is because the project is determined by time in terms of a start and end date. That the project is unpaid work does not diminish the importance of time management, but rather eliminates the focus on cost management. Cost is a part of the project management triangle (Figure 5.4) and usually a major factor within projects, but a side to the triangle not relative to this bachelor project. Time on the other hand was highly relevant, as the project was determined by an end date in terms of a final deadline.

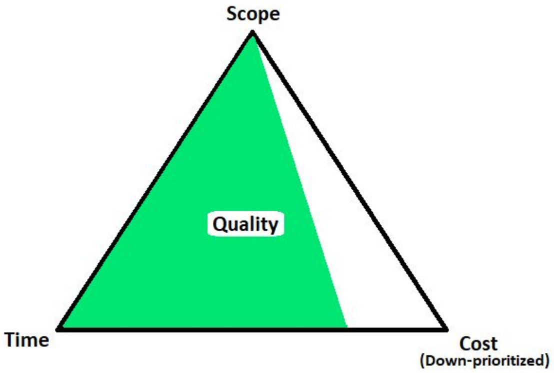


Figure 5.4 - Project management triangle

This for us meant that it was essential to have a way of managing how much each individual was working each day, and for how long we were supposed to work each day. This was in order to meet the requirements before the deadline and handover. We wanted a way to visualize this, to easily manage how we worked, as well as to best share the time management with IBM. By sharing the time management we would be able to ensure our working efforts towards the project despite IBM not being localised in Kristiansand.

When laying the groundwork for how many hours we should be working during the project we calculated the expected amount of working hours in the course, and made this the suggested target value for each sprint spread across the entire project. To keep track of our working hours we used an excel sheet solution which displayed the working hours in a Burndown chart. They are attached in Appendix D.

6. Reflection

When working with an agile approach, iteration is essential. This means going over functionalities already implemented several times, and making sure that they suit the desired end goal for functionality, availability and accessibility.

6.1 Roles

Scrum has three main roles; product owner, development team, and scrum master (West, 2020). In this project we make up the development team. In scrum the development team is the group of people who do the work. Scrum is not categorizing any further than these three roles or defining roles within the development team. It is up to each individual project team to create and organize a balanced and appropriate team for the task. A developer in scrum terms *"means a team member who has the right skills, as part of the team to do the work"* (West, 2020). As a development team within scrum is self-organized it was our job to structure how we would work. We did not define the roles within the team as we wanted to be flexible in regards to task management. Although there are specialties within the group of students, and especially one student would have his main focus in the back-end, and one student would be responsible for UX and design due to his photoshop and design background. Figure 6.1 illustrate that a development team in Scrum shares a wide range of tasks.

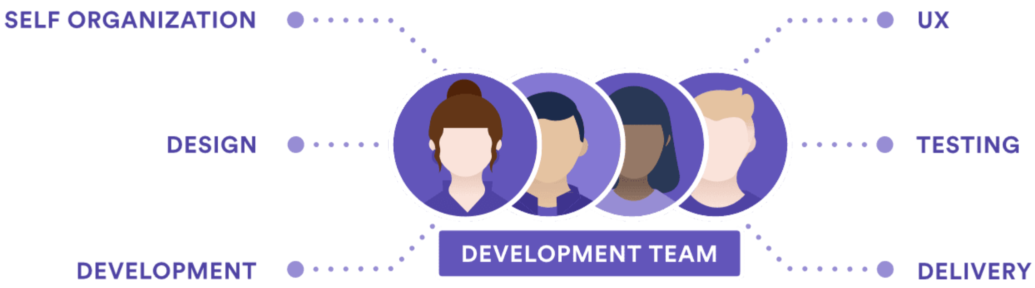


Figure 6.1 - Development team (https://www.atlassian.com/agile/scrum/roles)

6.2 Management

When following Scrum methodology the team is obliged to fill a member with the role of Scrum master. A Scrum master as explained by West *"...is a servant leader which not only describes a supportive style of leadership but describes what they do on a day-to-day basis"* (West 2020). The scrum master must ensure that the scrum approach is being well followed. To fulfill this role, helping the product owner to define values, and to support the development team in following the values is important. To serve the development team in helping them understand what scrum is, and create an environment that supports scrum, the scrum master has a focus on delivering scrum values within the team (West, 2020).

We appointed Marius J. Tjøstheim as scrum master in unison at the beginning of the pre-sprint. Arguments for having Tjøstheim as scrum master was that he was the group member with the most knowledge within programming and understanding system architecture. We found it helpful that the student with the most technical skills was appointed. This changed in sprint 2, as Tjøstheim felt that he did not work as scrum master in terms of the requirements for the role. The group agreed on his reflection, and decided that Henrik Lindseth would continue as scrum master from sprint 2 and forwards.

As part of working as a Scrum master is a responsibility to organize the project, in for example managing the backlog, helping the team in self-organizing and supporting work planning, communicating values, and helping the team at large. Managing these responsibilities consume time and effort, and Tjøstheim felt that he could better help the team with technical effort not being Scrum master. This led to a shift where Lindseth took over the responsibility as Scrum master. The group came to this agreement because he was the one that was pushing the Scrum approach on the rest of the group and in terms was acting as the Scrum master from the beginning. Therefore, not much changed in regards to team dynamics as of this decision.

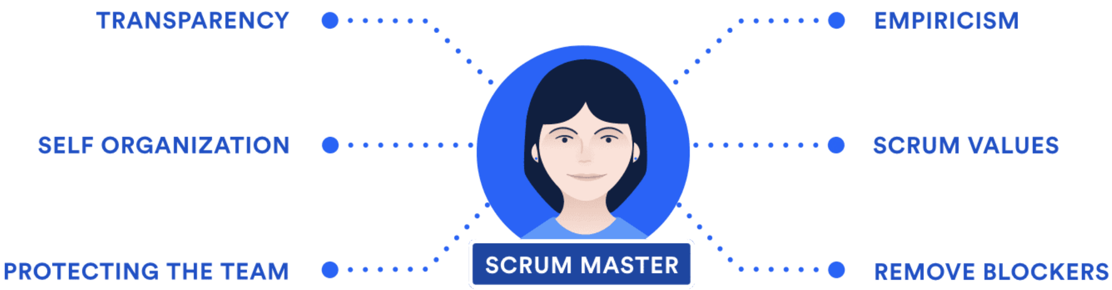


Figure 6.2 - Scrum master (https://www.atlassian.com/agile/scrum/roles)

To follow Scrum policy the Scrum master was not working as the executive leader, but rather fulfilled the role of a servant leader with an overall responsibility. Group democracy was present during the project, and decisions were voted for, with all opinions counting equally. For this process the Scrum master voted last, thus determining the outcome of the result if the votes of the other group members were even, thereby maintaining the role of an overall leader. Managing the project was relatively simple, as there were few disagreements within the group. With a working environment where to share our opinions and open discussions is encouraged, the project management becomes clear and simple. Figure 6.2 illustrates central values and tasks for the Scrum master.

6.3 Home office

In advance to the COVID-19 outbreak we organized meetings at UiA each day and booked large group rooms as our preferred work environment. Explained in chapter *3.1.2 Changes and challenges,* we were forced to work from home after restrictions from the norwegian government due to COVID-19. Because of this we lost the work ethic we previously had achieved. Right after the change of scene, we noticeably lost some of our work ethic. The main reason behind this decrease in effectiveness we believe was directly connected to the loss of a common office, in addition to a new way of communication and collaborating together. The workflow was harder to keep up, because of the situation with COVID-19 and lockdown. The lockdown served as a demotivator, and a stagnation to the project, even though we kept daily stand-up meetings at the usual time. During our set workdays we were always available on Discord, but the dynamic between the group members was not similar to the pre-pandemic dynamic. It was less hands-on and a bit more individualized. It was harder to collaborate directly with each other, because of the inability to spontaneously, and directly show/explain issues at tasks right then and there.

In addition, being home or at the same location for such a long time can temper your mood and thereafter spoil the work capacity, resulting in a sort of cabin fever. We profit off a vivid work culture, and naturally it is harder to stay positive and focused on the end target during such circumstances. After noticing this trend of changing working behaviour we were able to adjust, and after getting used to the new workflow we noticed tremendous progress and efficiency.

The collaboration with our employer IBM did not change however, in regards to the COVID-19 outbreak. With IBM being based out of Oslo, and the students based out of Kristiansand, besides the initial kick-off event all of our communication has been maintained online, and did not change the dynamic between those parties. This made working from home a smooth transition in regards to communicating with IBM.

With the restrictions to no longer being able to meet physically, it forced us to have online meetings instead. We utilized Discord, a communication tool that enabled for text, voice and video communication, as the replacement for the physical meetings, acting as a virtual office.

By using Discord we could easily share screens and contact each other, so that in essence the internal collaboration dynamics remained as usual. In order to keep the same meeting routines as earlier, we also decided to retain our usual work hours, 09-15. The scrum master argued for the group to keep the same work routines as before the COVID-19 outbreak, because a drastic change in routines could affect the progress and collaboration in the project. The development team also thought it would be best for the group to continue as before. Daily stand-ups therefore remained at 09.00 to ensure that the scrum approach would remain, and the communication would continue as pre-pandemic.

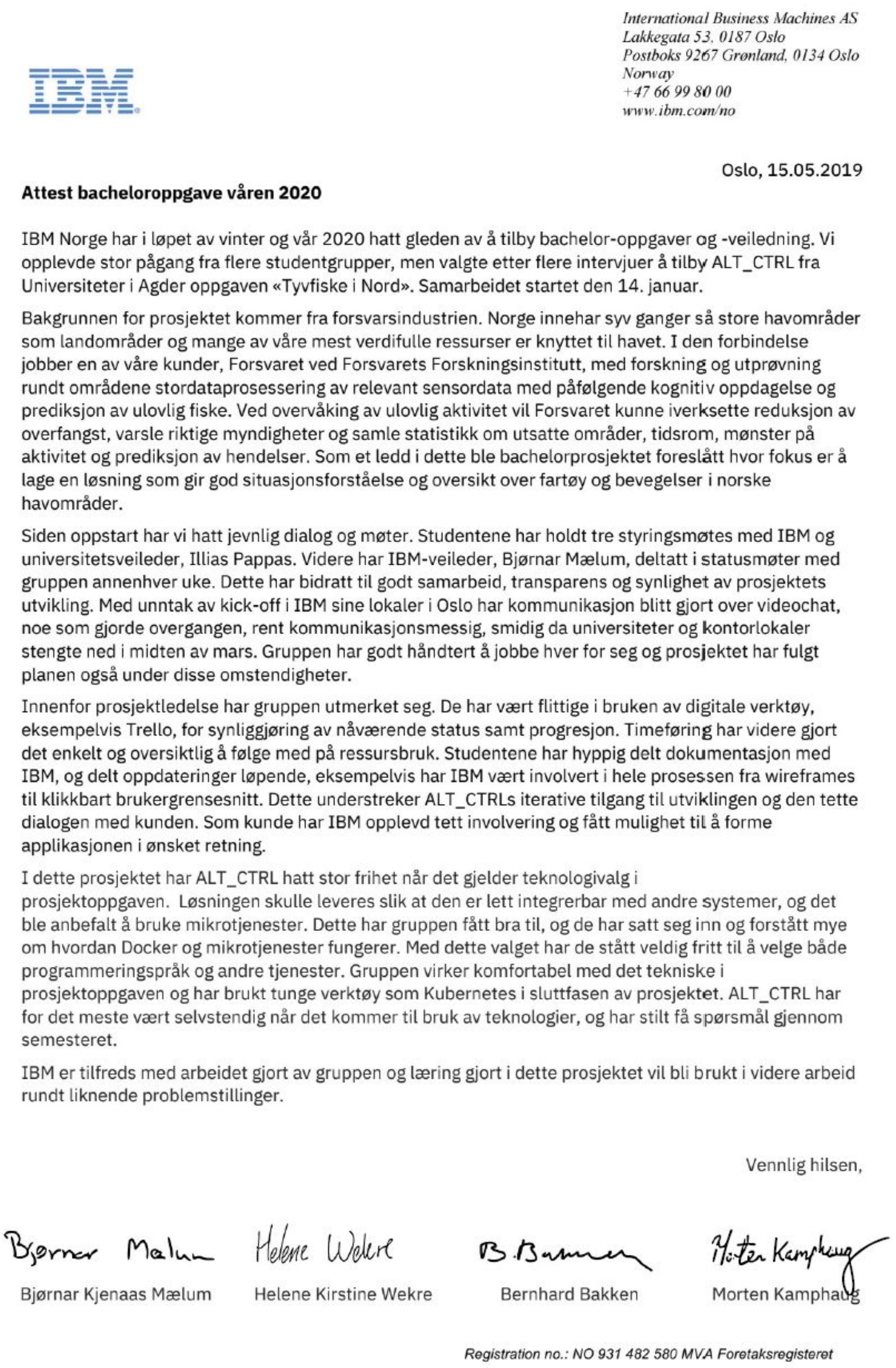
7. Result

In conclusion a project success and execution depends on several different aspects. In terms of our project we have satisfied all parties involved in the project being IBM, UiA, and the individuals in the team. The last steering committee meeting confirmed that IBM was satisfied by our efforts and wanted us to present the product for the customer. We have managed to gather and decode all data essential in order to identify marine activity in the Norwegian coastline.

Thereby we have satisfied the requirements from the customer to help them track certain types of ships, and to coordinate through their historical data, by the mapping service. A marine analytic would now be able to recognize sailing patterns, and segregate these ships that could be practising poaching from those who do not. We have been able to challenge ourselves by learning about system development management and planning. In addition to creating a system with new technologies. This helped us achieve what we planned for in terms of scope, and solving the scope creep. We learned that this could be achieved by incorporating heavy consulting with the customer, to satisfy both parties' needs within the set Scrum methodology time frame.

For further development it could be useful to implement a solution to automatically detect and alert irregular behaviour. However manual detection is something which was desired by the customer, and therefore focus for the system.

8. Statement from IBM



International Business Machines AS
Lakkegata 53, 0187 Oslo
Postboks 9267 Grønland, 0134 Oslo
Norway
+47 66 99 80 00
www.ibm.com/no

Oslo, 15.05.2019

Attest bacheloroppgave våren 2020

IBM Norge har i løpet av vinter og vår 2020 hatt gleden av å tilby bachelor-oppgaver og -veiledning. Vi opplevde stor pågang fra flere studentgrupper, men valgte etter flere intervjuer å tilby ALT_CTRL fra Universiteter i Agder oppgaven «Tyvfiske i Nord». Samarbeidet startet den 14. januar.

Bakgrunnen for prosjektet kommer fra forsvarsindustrien. Norge innehar syv ganger så store havområder som landområder og mange av våre mest verdifulle ressurser er knyttet til havet. I den forbindelse jobber en av våre kunder, Forsvaret ved Forsvarets Forskningsinstitutt, med forskning og utprøvning rundt områdene stordataprosessering av relevant sensordata med påfølgende kognitiv oppdagelse og prediksjon av ulovlig fiske. Ved overvåking av ulovlig aktivitet vil Forsvaret kunne iverksette reduksjon av overfangst, varsle riktige myndigheter og samle statistikk om utsatte områder, tidsrom, mønster på aktivitet og prediksjon av hendelser. Som et ledd i dette ble bachelorprosjektet foreslått hvor fokus er å lage en løsning som gir god situasjonsforståelse og oversikt over fartøy og bevegelser i norske havområder.

Siden oppstart har vi hatt jevnlig dialog og møter. Studentene har holdt tre styringsmøtes med IBM og universitetsveileder, Illias Pappas. Videre har IBM-veileder, Bjørnar Mælum, deltatt i statusmøter med gruppen annenhver uke. Dette har bidratt til godt samarbeid, transparens og synlighet av prosjektets utvikling. Med unntak av kick-off i IBM sine lokaler i Oslo har kommunikasjon blitt gjort over videochat, noe som gjorde overgangen, rent kommunikasjonsmessig, smidig da universiteter og kontorlokaler stengte ned i midten av mars. Gruppen har godt håndtert å jobbe hver for seg og prosjektet har fulgt planen også under disse omstendigheter.

Innenfor prosjektledelse har gruppen utmerket seg. De har vært flittige i bruken av digitale verktøy, eksempelvis Trello, for synliggjøring av nåværende status samt progresjon. Timeføring har videre gjort det enkelt og oversiktlig å følge med på ressursbruk. Studentene har hyppig delt dokumentasjon med IBM, og delt oppdateringer løpende, eksempelvis har IBM vært involvert i hele prosessen fra wireframes til klikkbart brukergrensesnitt. Dette understreker ALT_CTRLs iterative tilgang til utviklingen og den tette dialogen med kunden. Som kunde har IBM opplevd tett involvering og fått mulighet til å forme applikasjonen i ønsket retning.

I dette prosjektet har ALT_CTRL hatt stor frihet når det gjelder teknologivalg i prosjektoppgaven. Løsningen skulle leveres slik at den er lett integrerbar med andre systemer, og det ble anbefalt å bruke mikrotjenester. Dette har gruppen fått bra til, og de har satt seg inn og forstått mye om hvordan Docker og mikrotjenester fungerer. Med dette valget har de stått veldig fritt til å velge både programmeringspråk og andre tjenester. Gruppen virker komfortabel med det tekniske i prosjektoppgaven og har brukt tunge verktøy som Kubernetes i sluttfasen av prosjektet. ALT_CTRL har for det meste vært selvstendig når det kommer til bruk av teknologier, og har stilt få spørsmål gjennom semesteret.

IBM er tilfreds med arbeidet gjort av gruppen og læring gjort i dette prosjektet vil bli brukt i videre arbeid rundt liknende problemstillinger.

Vennlig hilsen,

Bjørnar Mælum Helene Wekre Morten Kamphaug

Bjørnar Kjenaas Mælum Helene Kirstine Wekre Bernhard Bakken Morten Kamphaug

Registration no.: NO 931 482 580 MVA Foretaksregisteret

9. Self-assessment

All group members have contributed to the project in their own way, and we are all pleased with how we have executed the project. The following statements from each group member gives a reflection upon their own work.

Henrik Lindseth
For most of this project I have worked as the scrum master, and in return have kept a major focus on the scrum methodology as a whole. As scrum master I have worked as the executive leader for the project, and ensured that the scrum approach has been well followed. In addition to administrative tasks I have contributed to development where my focus area was front-end, which I find rewarding as you see the results of your progress consecutively. When the university closed mid-semester due to the coronavirus, it somewhat affected the work ethic within the group when we were forced to work from home. Even so I kept daily stand-ups at the same time throughout the project, and I am very proud that we reversed the trend and kept our spirit up in a demanding time. I have learned valuable lessons on group mechanics and development methodologies that I will benefit from later in my career.

Jostein Våga Rygg
During the Bachelors project I have been working as a swiss army knife. I've done some administrative tasks, regarding emails to both IBM and UiA. I have been working with some of the technical aspects regarding the project, mostly front-end, namely Leafet.js(Javascript), testing out frameworks that work with Leaflet Library and understanding it thoroughly. I have also contributed to the documentation for the bachelors project. During the bachelors project I have learned the value and challenges working with several individuals concerning the same goal. It has been a valuable lesson which I hope will strengthen my capabilities in my career.

Marius Johnsen Tjøstheim
As the person with the most technical knowledge I took responsibility for the most complicated implementations, and assisted the others in different areas where they needed it. I suggested - and often decided what frameworks and languages we should use in the project. It was important for both me and the others that I tried to teach and include the others in the work I did, and I think this led to a good way of using each other's strengths. In addition to all the technical experience I gained, I also gained valuable experience in communicating with different actors in a project, ranging from end-users, team members, and product owners.

Kristian Jul-Larsen
During this project I mainly focused on front-end and some back end giving me some experience with Node, JavaScript, Leaflet, HTML, CSS, Cloudant and IBM Cloud. I've learned how to work on a bigger project with multiple people. I'm very thankful for being a part of and working with this group. They are very skilled, good work flow which make them enjoyable to work with, both the group and IBM. This experience working on a bigger project will come in handy in my future with other projects. The biggest experience was how we handled expected and unexpected challenges.

Joakim Schjølberg

I have mostly focused on documentation and some administrative work. During the project I learned how important teamwork is when working on a project for this scale. I think the experience I have gained during this project will be beneficial later on my career. I also gained experience in how to deal with different stakeholders in a project. With a project of this size, and with so many different stakeholders communication is key. I quickly learned that in order for a project of this size to succeed, all the stakeholders need to be kept in the loop. At the start of the project our communication with the stakeholders was not up to par, however throughout the project we adjusted and saw the benefits of constant and good communication.

10. References

Bootstrap (2020, 8. May) Bootstrap. Retrieved from: https://getbootstrap.com/

IBM (2020, 30. March) *Overview Cloudant*. Retrieved from: https://cloud.ibm.com/docs/Cloudant?topic=cloudant-overview

Pletkus, L. (2019, 9. July). 5 Important Project Management Methodologies for Software Development (and How to Choose one) [Blog]. Retrieved from: https://scale3c.com/blog/important-project-management-methodologies-for-software-development-choose-one/

Satzinger, J.W., Jackson, J.B. & Burd, S.D. (2014) *Systems Analysis and Design in a Changing World* Boston: Cengage Learning

Vladimir Agafonkin (2019, 17. November) Leaflet.
Retrieved from: https://leafletjs.com/

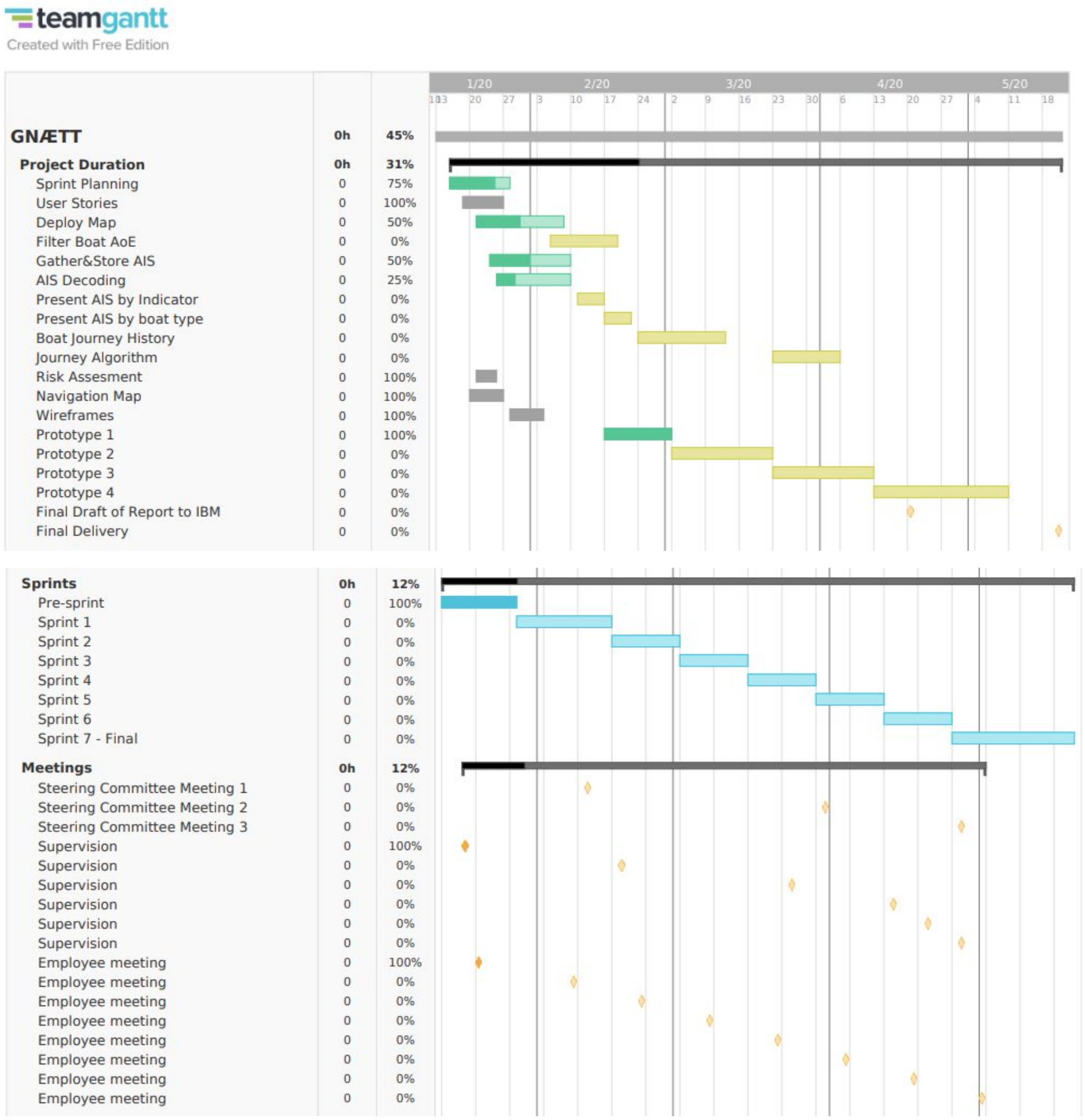
West, D. (2020). Scrum roles and the truth about job titles in scrum. Retrieved from: https://www.atlassian.com/agile/scrum/roles

Whitman, M.E. & Mattord, H.J. (2016). *Principles of Information Security* (5th ed.). Boston: Cengage Learning

11. Appendix

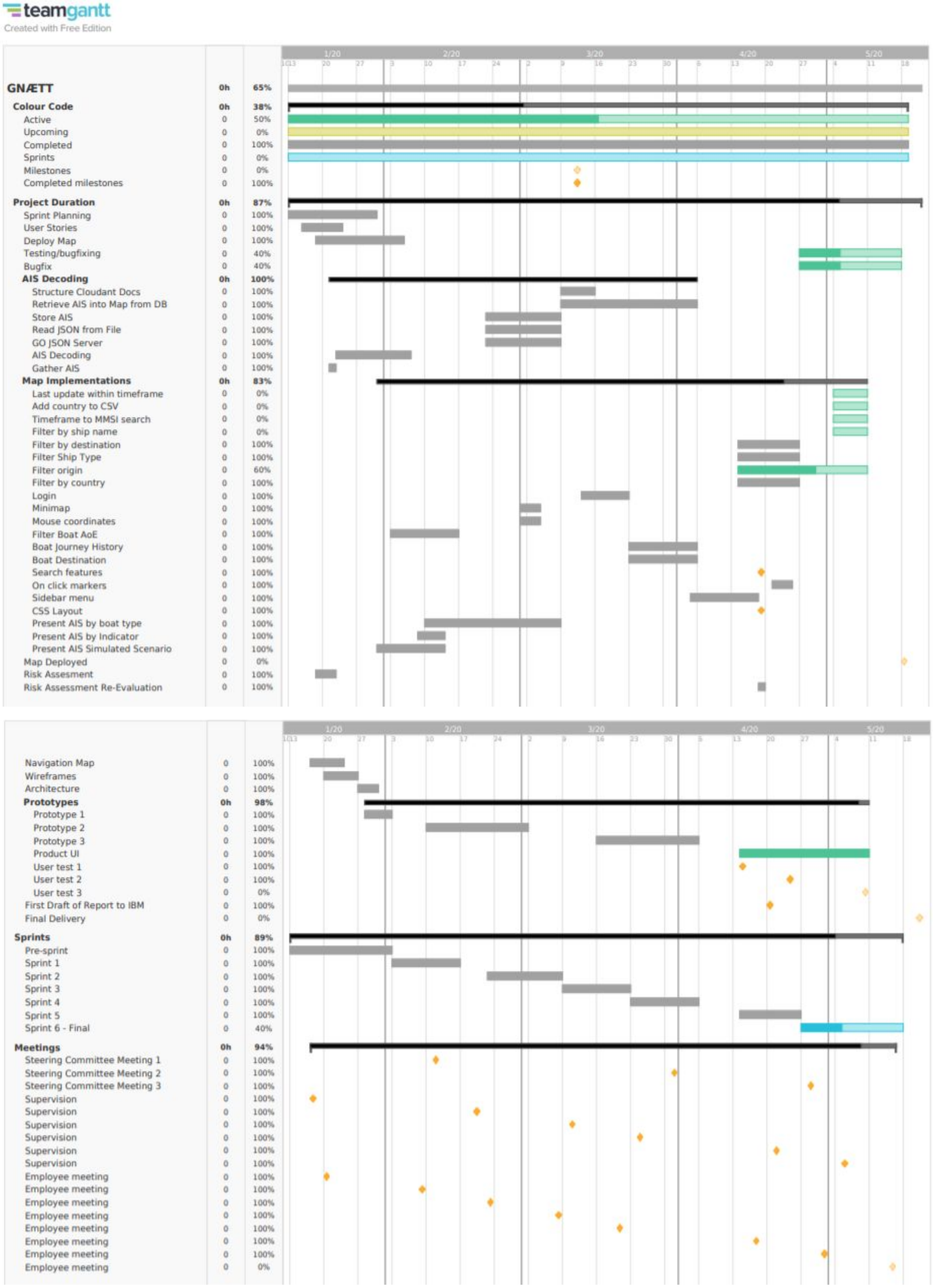
Appendix A

Gantt diagram v1 - pre-sprint



GNÆTT	**0h**	**45%**
Project Duration	**0h**	**31%**
Sprint Planning	0	75%
User Stories	0	100%
Deploy Map	0	50%
Filter Boat AoE	0	0%
Gather&Store AIS	0	50%
AIS Decoding	0	25%
Present AIS by Indicator	0	0%
Present AIS by boat type	0	0%
Boat Journey History	0	0%
Journey Algorithm	0	0%
Risk Assesment	0	100%
Navigation Map	0	100%
Wireframes	0	100%
Prototype 1	0	100%
Prototype 2	0	0%
Prototype 3	0	0%
Prototype 4	0	0%
Final Draft of Report to IBM	0	0%
Final Delivery	0	0%
Sprints	**0h**	**12%**
Pre-sprint	0	100%
Sprint 1	0	0%
Sprint 2	0	0%
Sprint 3	0	0%
Sprint 4	0	0%
Sprint 5	0	0%
Sprint 6	0	0%
Sprint 7 - Final	0	0%
Meetings	**0h**	**12%**
Steering Committee Meeting 1	0	0%
Steering Committee Meeting 2	0	0%
Steering Committee Meeting 3	0	0%
Supervision	0	100%
Supervision	0	0%
Supervision	0	0%
Supervision	0	0%
Supervision	0	0%
Supervision	0	0%
Employee meeting	0	100%
Employee meeting	0	0%
Employee meeting	0	0%
Employee meeting	0	0%
Employee meeting	0	0%
Employee meeting	0	0%
Employee meeting	0	0%
Employee meeting	0	0%

Gantt Diagram v2 - as of sprint 6



GNÆTT	**0h**	**65%**
Colour Code	**0h**	**38%**
Active	0	50%
Upcoming	0	0%
Completed	0	100%
Sprints	0	0%
Milestones	0	0%
Completed milestones	0	100%
Project Duration	**0h**	**87%**
Sprint Planning	0	100%
User Stories	0	100%
Deploy Map	0	100%
Testing/bugfixing	0	40%
Bugfix	0	40%
AIS Decoding	**0h**	**100%**
Structure Cloudant Docs	0	100%
Retrieve AIS into Map from DB	0	100%
Store AIS	0	100%
Read JSON from File	0	100%
GO JSON Server	0	100%
AIS Decoding	0	100%
Gather AIS	0	100%
Map Implementations	**0h**	**83%**
Last update within timeframe	0	0%
Add country to CSV	0	0%
Timeframe to MMSI search	0	0%
Filter by ship name	0	0%
Filter by destination	0	100%
Filter Ship Type	0	100%
Filter origin	0	60%
Filter by country	0	100%
Login	0	100%
Minimap	0	100%
Mouse coordinates	0	100%
Filter Boat AoE	0	100%
Boat Journey History	0	100%
Boat Destination	0	100%
Search features	0	100%
On click markers	0	100%
Sidebar menu	0	100%
CSS Layout	0	100%
Present AIS by boat type	0	100%
Present AIS by Indicator	0	100%
Present AIS Simulated Scenario	0	100%
Map Deployed	0	0%
Risk Assesment	0	100%
Risk Assessment Re-Evaluation	0	100%
Navigation Map	0	100%
Wireframes	0	100%
Architecture	0	100%
Prototypes	**0h**	**98%**
Prototype 1	0	100%
Prototype 2	0	100%
Prototype 3	0	100%
Product UI	0	100%
User test 1	0	100%
User test 2	0	100%
User test 3	0	0%
First Draft of Report to IBM	0	100%
Final Delivery	0	0%
Sprints	**0h**	**89%**
Pre-sprint	0	100%
Sprint 1	0	100%
Sprint 2	0	100%
Sprint 3	0	100%
Sprint 4	0	100%
Sprint 5	0	100%
Sprint 6 - Final	0	40%
Meetings	**0h**	**94%**
Steering Committee Meeting 1	0	100%
Steering Committee Meeting 2	0	100%
Steering Committee Meeting 3	0	100%
Supervision	0	100%
Supervision	0	100%
Supervision	0	100%
Supervision	0	100%
Supervision	0	100%
Supervision	0	100%
Employee meeting	0	100%
Employee meeting	0	100%
Employee meeting	0	100%
Employee meeting	0	100%
Employee meeting	0	100%
Employee meeting	0	100%
Employee meeting	0	100%
Employee meeting	0	0%

Appendix B

Risk matrix

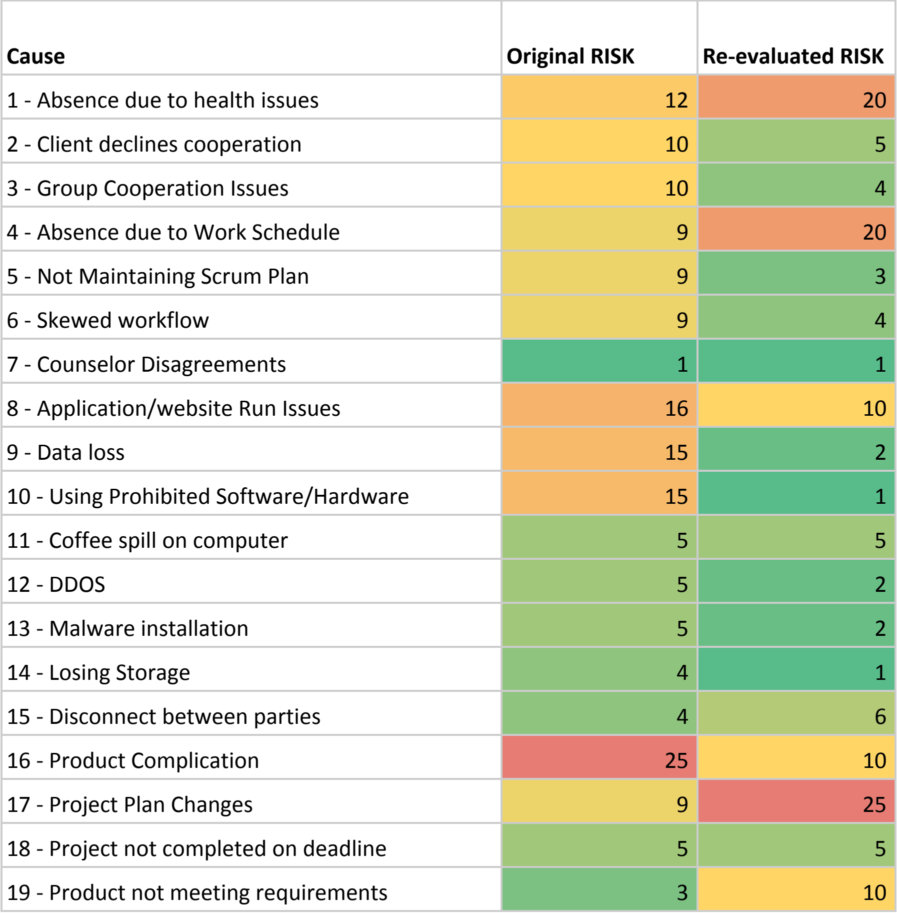
Problematic risks					
Cause	**Probability**	**Effect**	**RISK**	**Measurement to avoid Risk**	**Measurement to reduce damage**
Absence due to health issues	4	3	12	Eat healthy. Avoid contact from other sick people. If sick, do not go to school by risking others health.	Work from home. If possible.
Client declines cooperation	2	5	10	Keep the client happy. Have a good dialogue with the client and meet their requirements.	Backtrack to where the issue with cooperation started
Group Cooperation Issues	2	5	10	Identify and make sure members are aware of roles/workload needed to be done. Group Contract. Honesty/Open-mindedness	Check Group Contract. Evaluate the situation. Come to an agreement. Expel in case of severe violations.
Absence due to Work Schedule	3	3	9	Know your work schedule, and try to change your work hours to not conflict with the project work.	If absolutely needed at work, tell the group in advance. Work with the project in advance or catch up on the project missed due other work.
Not Maintaining Scrum Plan	3	3	9	Have daily scrum meetings.	High work ethic and independent group members.
Skewed workflow	3	3	9	Follow scrum plan, arrange tasks.	Superior scrum master that controls all group members has a task to do. Also an independent team that finds new objectives when old ones are completed.
Counselor Disagreements	1	1	1	Continuous open dialogue with both client and counselor. When contradictions are met, discuss them with both counselor and client. Keep a steady course.	Listen to all parties opinions, and in the end - the client knows best. Try to meet the clients expectations and wishes for the project and cooperation.
Technical risks					
Cause	**Probability**	**Effect**	**RISK**	**Measurement to avoid Risk**	**Measurement to reduce damage**
Application/website Run Issues	4	4	16	Using a trusted and familiar framework.	Follow IBM Coding Standards

Data loss	3	5	15	Save progress often. Rules when merging repositories. Use cloud	Back-up in cloud or hard drive. Try to recover the lost data, and if not possible evaluate the importance of the data.
Using Prohibited Software/Hardware	3	5	15	Agreements with IBM, and maintain open-dialogue about prohibitions	Backtrack process or Meeting with IBM regarding prohibitions
Coffee spill on computer	1	5	5	Close your coffee mug while your beverage and computer is stored in the same compartment.	Dry computer, get fixed ASAP if broken
DDOS	1	5	5	Deploy anti-DDoS hardware and software modules. Deploy a DDoS protection appliance. Protect DNS servers.	Block the attackers. Systems checklist/response plan.
Malware installation	1	5	5	Keep your malware protection programs such as malwarebytes updated and updated antivirus programs.	If infectious malware has entered the system/pc, then scan the pc for malware. Worst case, retrieve data from a backup done earlier where there is no infectious malware.
Losing Storage	2	2	4	Keep backuping your work on an external hard drive or backup in the cloud.	Try to extract the storage lost. Get expert help.
Disconnect between parties	2	2	4	Keep software updated at all times.	Wait for the connection to get back, or try to contact the other part.
Risks toward the user					
Cause	**Probability**	**Effect**	**RISK**	**Measurement to avoid Risk**	**Measurement to reduce damage**
Product Complication	5	5	25	Project Scope Management. Compartmentalize Tasks needed to be done.	Get whatever is slightly finished working.
Project Plan Changes	3	1	9	Scope management.	When a change in project plan is necessary, arrange a meeting with the client and discuss the new plan. Deliver your thoughts and concerns for what must be done differently.
Project not completed on deadline	1	5	5	Keep working through the entire semester and use our counseling to our benefit. Ask our counselor both at our client and at the University for help when stuck.	Ask for reasonings behind the decision

Product not meeting requirements	3	1	3	Scope management. Evaluate the importance of said requirement (MoSCoW), and cut the requirements not necessary for base functionality.	Make use of the valuable learning goals of working on a project for a client. Evaluate how the group handled the situation, and what could have been a more beneficial way of tackling the task

Appendix C

Risk comparison - original vs new



Cause	Original RISK	Re-evaluated RISK
1 - Absence due to health issues	12	20
2 - Client declines cooperation	10	5
3 - Group Cooperation Issues	10	4
4 - Absence due to Work Schedule	9	20
5 - Not Maintaining Scrum Plan	9	3
6 - Skewed workflow	9	4
7 - Counselor Disagreements	1	1
8 - Application/website Run Issues	16	10
9 - Data loss	15	2
10 - Using Prohibited Software/Hardware	15	1
11 - Coffee spill on computer	5	5
12 - DDOS	5	2
13 - Malware installation	5	2
14 - Losing Storage	4	1
15 - Disconnect between parties	4	6
16 - Product Complication	25	10
17 - Project Plan Changes	9	25
18 - Project not completed on deadline	5	5
19 - Product not meeting requirements	3	10

Appendix D

Burn down charts

Sprint 1:

Day of Sprint	Kristian	Henrik	Jostein	Joakim	Marius	Total Effort
1	6	6	6	6	5	29
2	5	5	5	0	1	16
3	0	4	4	0	6	14
4	6	6	6	5	1	24
5	3	2	3	3	3	14
6	6	0	6	5	6	23
7	6,5	6,5	6	6	6,5	31,5
8	7,5	6	3	3	4	23,5
9	6	6	6	0	1	19
10	5	6	5	0	5,5	21,5
	51	47,5	50	28	39	215,5

Sprint 2:

Day of Sprint	Kristian	Henrik	Jostein	Joakim	Marius	Total Effort
1	5	4	4	0	4	17
2	6	7	6	6	6	31
3	0	7	4	1	0	12
4	0	8	5	6	2	21
5	3,5	6	6	6	2	23,5
6	8	5	5,5	0	5,5	24
7	7	7	6,5	7	6	33,5
8	6	6	6	6	6	30
9	6	5,5	7	6	6	30,5
10	6	0	6	5	6	23
11	1	0	1	1	1	4
	48,5	55,5	57	44	44,5	249,5

Sprint 3:

Day of Sprint	Kristian	Henrik	Jostein	Joakim	Marius	Total Effort
1	5	5	5	0	1	16
2	3	6	6	6	6	27
3	4	3	0	0	1	8
4	6	6	0	3	0	15
5	4	2	0	0	5	11
6	7,5	6	4	4	6	27,5
7	6,5	6,5	6	6	6	31
8	5	6	5	6	6	28
9	6	5,5	2	2	5	20,5
10	7	3	4	2	6	22
	54	49	32	29	42	206

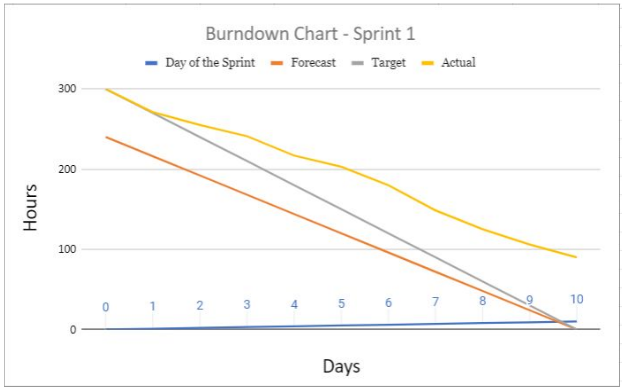
Sprint 4:

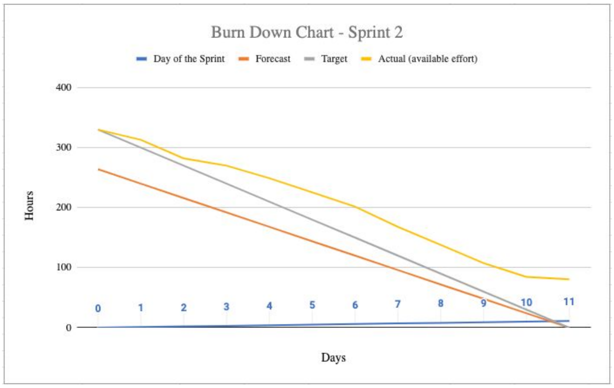
Day of Sprint	Kristian	Henrik	Jostein	Joakim	Marius	Total Effort
1	6	6	6,5	6	8,5	33
2	6,5	6	6	6	7	31,5
3	6	5	6	5,5	7,5	30
4	6	4	3	6	6	25
5	6,5	6	5	0	6	23,5
6	6,5	7	6	6	9	34,5
7	6	4,5	6	6	6	28,5
8	6	6	4	6	7	29
9	5	4,5	6	2	0	17,5
10	6	6	4	4	5	25
	60,5	55	52,5	47,5	62	277,5

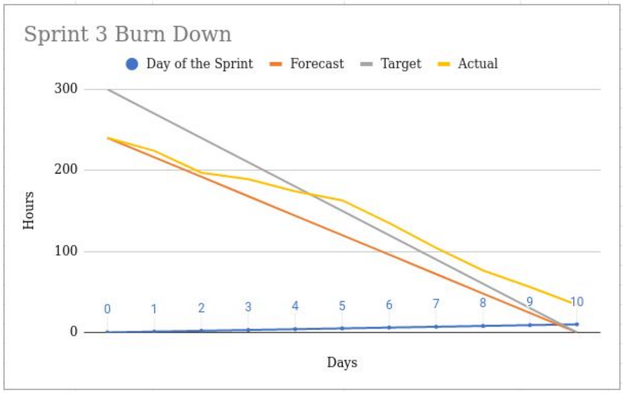
Sprint 5:

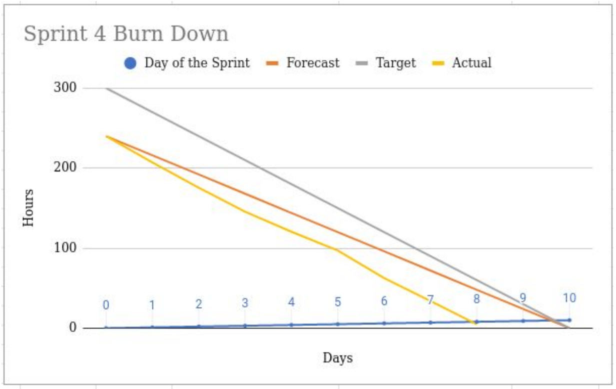
Day of Sprint	Kristian	Henrik	Jostein	Joakim	Marius	Total Effort
1	6	6	6	5	6,5	29,5
2	5	5	6	3	6	25
3	5,5	6	5	4	5	25,5
4	6,5	6	5	6	6,5	30
5	0	5	6	5	6	22
6	6	2	6	6,5	5,5	26
7	7,5	1	6	6	7	27,5
8	6,5	6	7,5	6	7	33
9	6,5	2,5	4	0	3	16
	49,5	39,5	51,5	41,5	52,5	234,5

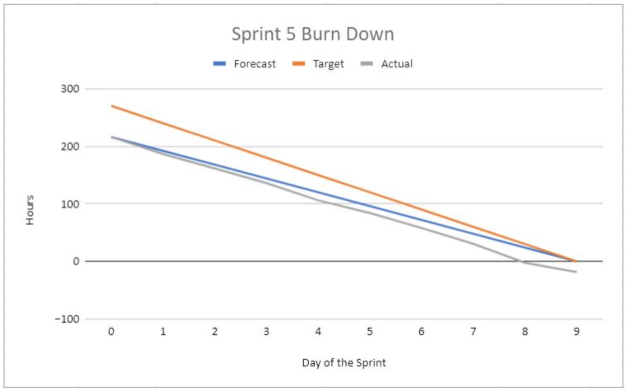
Charts sprint 1-5:











Appendix E

User stories

1 / 14

To better locate the activity in the Norwegian coast,
As a Coast Guard analyst,
I want a visual representation of ship activity

MoSCoW: Must have

Criteria: The criteria for this user story, we set to be the ability to navigate through a map with updated locations of ships though automatic identification system (AIS). We view this core functionality as a necessity for the system. A map with accurate and updated locations of ships we deem as a top priority for the system and Coast Guard.

Comment: This live map will provide useful information and enhance the coastal overview. Naturally, we believe that the map should be interactive and AIS data should be accessible through navigation.

Acceptance criteria:

1. The map must be interactive.
2. The AIS data must be live and available in the mapserver.
3. The functionality of the mapserver must not be weakened by the addition of AIS data.

2 / 14

To view a specific area,
as a Coast Guard analyst,
I want to have a zoom function

MoSCoW: Must have

Criteria: Zooming on a specific area gives the user a clearer and narrower overview on specific areas. If not being able to zoom, all ships will be a cluster on the map and ships will be indistinguishable, rendering the map useless.

Comment: Zooming is an important function to implement. Instead of viewing the whole coast of Norway, the operator can zoom in i.e. Lofoten if the user wants to view the activity in Lofoten specifically.

Acceptance criteria:

1. Must be able to zoom both in and out.
2. Zooming where the mouse pointer is with mouse wheel/finger stretch.
3. Zooming should be responsive and precise.
4. Zooming function should have buttons

3 / 14

I want information boxes connected to specific ships,
As a Coast Guard analyst,
So that I can easily access information of ships.

MoSCoW: Must have

Criteria: The information boxes should not be visible in the map overview, but instantly reveal data at the request of chosen ship.

Comment: When selecting a ship a box with information regarding the ships dimension, size, cargo, destination etc..

Acceptance criteria:

1. Every ship should have its own information box, accessible from clicking on the ship
2. The information box should contain valid and updated information from the ship's own crew.

4 / 14

As a Coast Guard analyst,
I want to see the past movement of a ship,
So that I can analyze their history

MoSCoW: Must have

Criteria: The criteria for this user story is that the AIS data can be stored and traceable (history). As the first user story revolves around live updates of ship traffic, the purpose of this story is to demonstrate the importance of storing the AIS data. The customer benefits from tracking the history of a ship, to scan for irregular behavior. To achieve this in our system we would need a database.

Comment: To fulfil the criteria of this user story we need a database that tolerates the amount of data and implement a solid structure. A core functionality of presenting the data must be implemented into the map itself, for the AIS data to be easily accessible.

Acceptance criteria:

1. History of specific ships must be implemented as a function in the map service.
2. The movement data of ships must be stored.
3. The database must tolerate big data sets.
4. The ships movement history should be easily available from the human interaction of the map.

5 / 14

To locate commercialized fishing,
As a Coast Guard analyst,
I want an identifier for ships that have reported fishing.

MoSCoW: Should have

Criteria: To fulfil this user story we would likely have to implement a function that uses AIS to distinguish ships that are currently fishing. The goal is that the map automatically updates when new information enters the server, and a specific identifier which is associated with fishing.

Comment: This would make it so that the users will have a clear overview of which ships should be fishing at any given moment.

Acceptance criteria:

1. A distinguishable identifier for fishing activities.
2. Identifier for specific ship
3. Identifier should not be active on ships who have not reported fishing
4. Identifier is active on ships that are fishing

6 / 14

As a Coast Guard analyst,
I want an alert of irregular activities,
So that I know what ships to focus on.

MoSCoW: Could have

Criteria: To fulfill this user story we have to implement an algorithm that can detect if a ship is fishing. An algorithm can detect a pattern if a ship is fishing. This algorithm needs to know different pattern of ships that are at sea fishing. The algorithm will run on ships that has the pattern that identifies as fishing and has not set their status to fishing.

Comment: Detecting different and unusual patterns on how a ships travel. A ship sailing from point A to point B will be travelling in a straight line, a ship fishing will go in specific or random patterns in the same area. It is statutory to turn on their AIS transmitter and set their status that they are fishing when they are fishing, and turn the status off when they are not fishing. Their pattern when fishing can be totally random, circular, sailing i.e. south for a certain distance and then sail back north or where it came from.

Acceptance criteria:

1. The result calculated by the algorithm must be presented to the user of the system as an alert.
2. The algorithm must only alert the user of suspicious behavior through irregular patterns.
3. Tests must be created, ran and passed in order to implement the algorithm.
4. Know different patterns which results in fishing.
5. Identifies ships that actually are poaching and not private individuals sailing.

7 / 14

I want to see the destination of the ship,
As a Coast Guard analyst,
So I know where the ship is supposed to travel.

MoSCoW: Could have

Criteria: All ships in the system should have information easily accessible from interacting with the map, in addition to the ship's exact destination. Where did the ship sail from, and which port will it dock next? This information should help the user get an understanding of the ship's route.

Comment: The functionality in the user story will supply the map with useful information on where ships are headed.

Acceptance criteria:

1. The destination of every ship should be easily accessible from interacting with the map.
2. The destination is shown by start and end point.

8 / 14

As a Coast Guard,
I want to distinguish ships by type,
To get an idea of the ships purpose

MoSCoW: Could have

Criteria: It is very useful to distinguish between different types of ships. The user gets to assume its purpose. The type of the ship will define what kind of activity it can or will do. If it's a cargo ship you know its route and that is often to point A to point B. This will help the user to know what ships to look for.

Comment: This story enables the user to distinguish ships. It is useful to know what type of ship we are looking at. This information will show when selecting and requesting a specific ship on the map.

Acceptance criteria:

1. Every ship must have information on what their ship type is.
2. It should be possible to distinguish between different types of ships.

9 / 14

**As a Coast Guard analyst,
I want to know a ships dimensions,
So I can detect irregular maneuvers**

MoSCoW: Could have

Criteria: The system should have information on the ships size. It is important to know the ships dimensions to know how big it is, and how much fish/freight the ship can hold. To be able to view ship size we need a database with information about the ships on our map which is provided by AIS data.

Comment: This information is useful for the Coast Guard to know which ship to look for if they decide to control the ship.

Acceptance criteria:

1. Size dimensions should be accessible.
2. Shown in the information box.

10 / 14

**As a Coast Guard analyst,
I want to filter ships,
So that I can view a specific type of ship.**

MoSCoW: Won't have

Criteria: A filter that can sort ships by type. Selecting/deselecting multiple types of ship to be viewed on the map.

Comment: A filter is useful to filter out the ships you do not need to inspect. A filter that categorizes ships to reduce clutter.

Acceptance criteria:

1. The user can select which types of ships to see on the map.
2. Able to select/deselect multiple types of ships.

11 / 14

**I want to know expected route,
As a Coast Guard analyst,
So that I can predict trajectory**

MoSCoW: Won't have

Criteria: A line (preferably dotted line) from point Y to Z to represent the expected route for a voyage from Y to Z. The line should either be in sync with what is the normal voyage for this trip, and/or in sync with the standard procedures of the shipowners.

Comment: The meaning behind this user story is that the expected route could illustrate if the ship is where it supposedly should be, or is unaligned with the predicted prejectory. The function could be a tool to help make the job handier/quicker for the analyst, although we believe an analyst would be able to interpret this without such a function, if the information of the destination is available. We find it hard to calculate the scale of functionality, as well as the difficulty.

Acceptance criteria:

1. A predicted line appearing on the map.
2. A method to implement the line on the map.
3. Showing voyage from port Y to Z.

12 / 14

As a Coast Guard analyst,
I want to view a chosen time-frame of a specific ship,
So that I can view where the ship has sailed at any given moment

MoSCoW: Won't have

Criteria: Selecting two different dates and time to view the voyage the specific ship has had between that time-frame.

Comment: This function is important for the algorithms to work. Selecting a time-frame makes it possible for the operator/user to focus on a specific timeline to see the activity of the ship.

Acceptance criteria:

1. Select the dates where the specific time-frame is wanted.
2. If the ship has sailed during the time-frame, a line will appear where it has been.

13 / 14

As a Coast Guard analyst,
I want a calendar,
So that I can choose which day I want to view.

MoSCoW: Won't have

Criteria: To fulfill this user story we have to implement a calendar where you can select different days to see the history of AIS data. This calendar should have all seven days of the week and all 12 months of the week. The calendar should be able to go back in time as far as possible, depending on when we decide to delete the AIS data from the database.

Comment: A calendar is easy and familiar to use for everyone. This makes it easy to choose which date, day and month the operator want to view where a specific ship has been in recent days or if the operator just want to see the activity history.

Acceptance criteria:

1. Show accurate dates and months.
2. Interactive calendar where the user can switch between days and months
3. Use a Calendar API or make a calendar on your own

14 / 14

As a Coast Guard analyst,
I want to detect fleets,
So that I know when ships cooperating.

MoSCoW: Won't have

Criteria: A function that shows if the ships are travelling together in a fleet. This will be a group of two or more ships that are sailing together. Detecting if the ships have their AIS sender on to see if they are fishing together.

Comment: Some ships may be fishing and working together. This can be two or more ships within the same company, or friends. There are a lot of factors making some ships fish together and travel in a fleet.

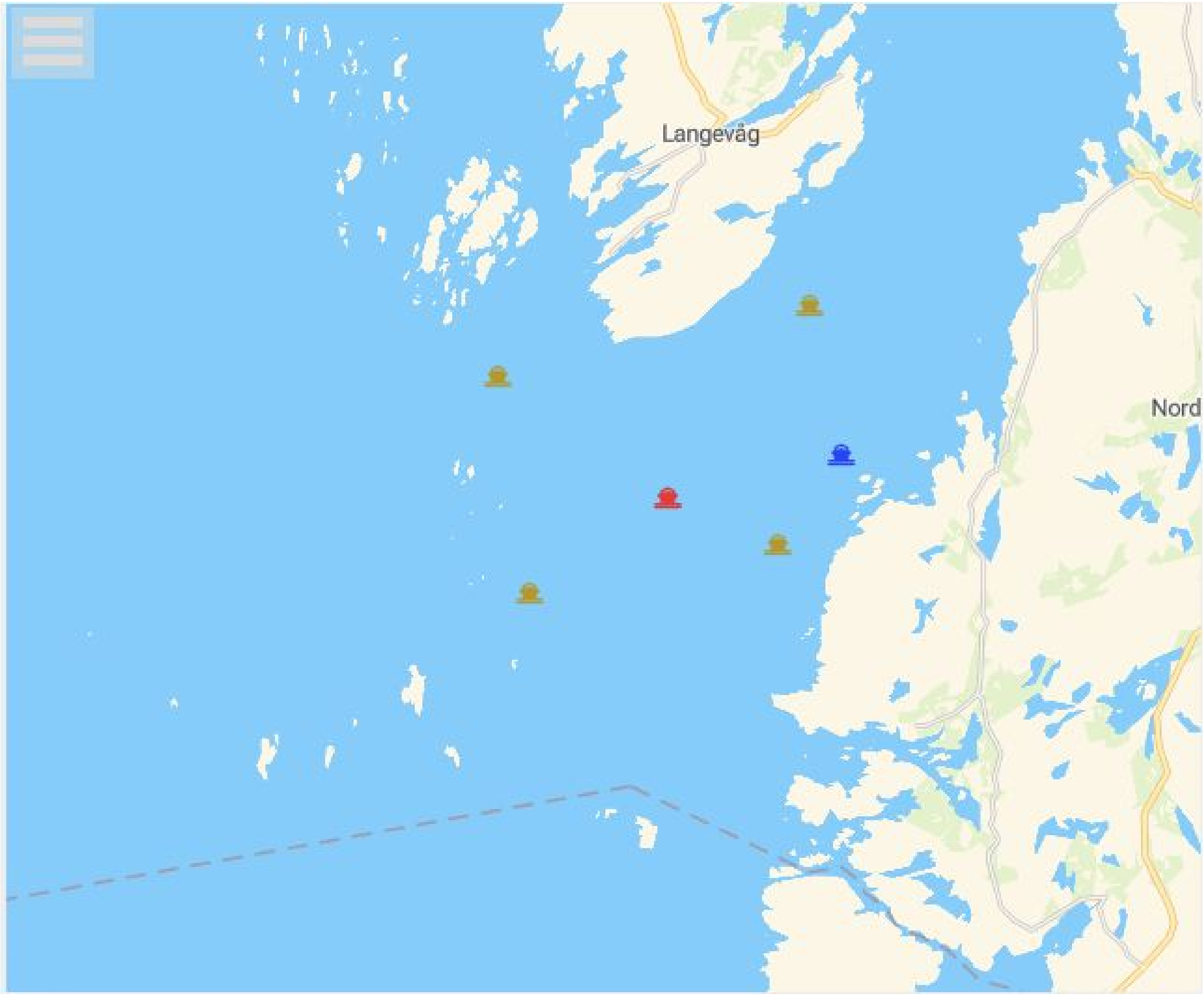
Acceptance criteria:

1. Have some sort of "links" between them, or the same colour, to see that they are in a fleet together.

Appendix F

UXPin Prototype

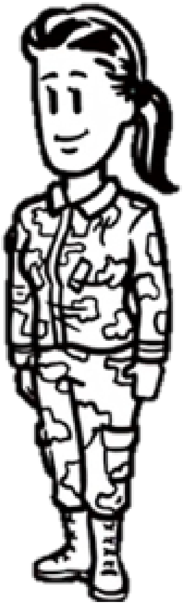
Direct link to the interactive Prototype:
https://preview.uxpin.com/1ef858069e00b66bbc9b1154142125d310a20046#/pages/122770036



Appendix G

Webex intervju med Karen

14.02.2020, Kristiansand



Karen jobber i Kystvakten. Hun jobber med situasjonsforståelse og er ansvarlig for analyse av Barentshavet. Hun representerer systemets målgruppe, og formålet med dette intervjuet er at vi som utviklere skal bli bedre kjent med kunden sitt behov og ønskelige funksjoner for det endelige systemet. Vårt ønske er å finne ut så mye som mulig av hva slags utfordringer kunden møter med det gamle systemet i hverdagen, og finne ut hva det nye systemet trenger for å bedre løse disse utfordringene. Vi vil med vårt system utvikle en løsning som kunden kan bruke for å få en enklere arbeidshverdag.

Vedlegg:
Wire frames, Gantt diagram, Prototype

Spørsmål:

ARBEIDSRUTINER

1. Hva er viktig for deg at et system inneholder dersom det skal hjelpe deg med dine arbeidsrutiner og utfordringer?

Svar: Dagens system - manuell input i fysisk kart. Får inn data, koordinater, status i csv fil for så å putte i excel. Bruker mye tid på daglige statusrapport. Utfordringer: lite visuelt, lese av tall. Ønsker å se kartet digitalt for å visualisere dataene. Effektivisere prosessen.

2. Er der noe du føler mangler i dagens system?

Svar: Får data om skip, model, navn, last, status, type skip, koordinater, historikk i separate csv filer - må importeres til Excel.

3. Hva synes du er positivt med dagens system?

Svar: Det gode er at det er Karen som setter premissene for det som anses som uregelmessigheter. Selv utfører analyse. Egen innvirkning.

4. Hva synes du er negativt med systemet du bruker i dag?

Svar: Tidkrevende

5. Er det en funksjon som du savner i dagens system?

Svar: Helautomatisert, med innvirking på analyse selv. Regler for analyse i systemet. Ønsker å se prosessen.

TEKNISK

1. Skip sender ut mye informasjon gjennom AIS data, hva av denne informasjonen er interessant for deg? ***Viktigste er huket av✔:***

a) Navn	c) Koordinater ✔	e) Destinasjon ✔	g) Mmsi	i) Annet… *Ruten, uoverensstemmelser i*

				(Analysere selv ut ifra) lokasjon, destinasjon og status (type båt)
b) Tilhørighet	d) Nav status ✔	f) Hastighet	h) Størrelse	j) **Alt!**

2. Dersom systemet mistenker tyvfiske, hvordan ønsker du å bli varslet?
    a. Notifikasjon?
    b. Lyd?
    c. Banner?

Det viktigste er å bli varslet raskt!

3. Vil du ha informasjon raskt eller nøyaktig?

Svar: Raskt og nøyaktig!

4. Hvor ofte vil du at punktene/kartet skal oppdateres?

Svar: Minimum på daglig basis. Det er slik det utføres i dag. Om mulig, enda raskere oppdateringer for å forbedre fra slik det gjøres i dag.

5. Er det viktig for deg å tilpasse systemet fortløpende? Eks: algoritmer for sannsynlighet, mengde bevegelsespunkter

Svar: Ikke slik som Netflix, men systemet skal fungere. Oppdatere hvis noe ikke fungerer, men systemet bør se likt ut sikt at hun ikke må sette seg inn i mye nytt hele tiden.

6. Vil du se hvor båten(e) har vært(historikken), og hvor lang tid tilbake vil du se båten?

Svar: Juridisk skal hun se helt fra skipet kom inn i Norsk grense. Noen ganger er hun avhengig av å se dette.

DESIGN

1. Foretrekker du en app-basert eller web-basert kartløsning?

Svar: Hun synes det er fint med web-basert siden de har pcer på jobb og ikke brukt mobil på jobb før.

2. For at systemet skal være lett å ta i bruk, hvordan kunne det sett ut? Typ. informasjonsboks?
    a. Forskjellig farge på båttyper? ✔ *høres fornuftig ut*
    b. Status for båter som fisker? ✔
    c. Filter? *Det tror hun kan hjelpe med å skrive rapporter hvis hun kan sortere ut båter.*
    d. Farge på kartet? Nattmodus? *Det er nais, man bare luksus ting*
    e. Skalert størrelse på ikonene mtp båtstørrelse?

Lage noen flere eksempler (demoer), sånn at hun kan velge den hun liker best. UI/UX, design messig.
Kommentarer til UXPin prototype: fargen på fiskebåtene likner på fargen på veiene. Bør stikke mere ut. Destinasjon og rute til skipene er nais, men hvordan kommer vi frem til rute? Er det med tanke på foretrukket rute eller antagelser? På bakgrunn av info registrert i AIS. HAUK: historisk rute? Kanskje være klarere på dette, siden Karen ikke så med en gang om det var tidligere rute eller antatt rute.
Scenario prototype: ZOOOOM! :D hvordan er den sammenlignet med UXPin? Pins istedenfor iconer :(
Hun liker idéen.