Østfold University College

BACHELOR'S THESIS

Development of Makerspace Management System
Group BO17-G14

Nicolai Naglestad
Thomas Begby
Espen Ottar Skjeggestad
Simon Chen Dybvik.

20/04/2017

Computer Science / Digital Media Production / Information Systems
Faculty of Computer Sciences

HØGSKOLEN I ØSTFOLD

Avdeling for Informasjonsteknologi
Remmen
1757 Halden
Telefon: 69 21 50 00
URL: www.hiof.no

BACHELOROPPGAVE

Prosjektkategori: Bacheloroppgave	X	Fritt tilgjengelig
Omfgang i studiepoeng: 20	(30/12 2029)	Fritt tilgjengelig etter
Fagområde: Informasjonsteknologi	(X)	Tilgjengelig etter avtale med oppdragsgiver

Tittel: Utvikling av MakerSpace Management System	Dato: March 21, 2017
Forfatterere: Nicolai Naglestad, Thomas Bergby, Espen Ottar Skjeggestad, Simon Chen Dybvik	Veileder: Børre Stenseth
Avdeling / Program: Avdeling for Informasjonsteknologi (alle programmer)	Gruppenummer: BO17-G14
Oppdragsgiver: HiØ/IT (MakerSpace)	Kontaktperson hos oppdragsgiver: Espen Teigen

Ekstrakt:
Det har vært en økende vektlegging på dokumentasjonen i bacheloroppgavene ved HiØ, slik at hoveddokumentet nå er grunnlaget for karaktersettingen. Formålet med dette prosjektet er å gjøre det enklere for studentene å produsere dokumentasjon med hensiktsmessig innhold, tradisjonell struktur, og profesjonell utforming. Rapporten starter med å redegjøre for generelle krav til vitenskapelige og tekniske rapporter. Det blir lagt spesielt vekt på kravene som stilles ved HiØ. Det gies en kort oversikt over hvordan man produserer og vedlikeholder dokumenter, både analoge og digitale. Deretter blir det utformet en mal som angir struktur og innhold i hoveddokumentet. Etter en ha utviklet en sett med minimumskrav til programvarene som skal brukes, blir det klart at kun to verktøy er aktuelle: LaTeX og *OpenOffice Writer*. En selvforklarende mal blir implementert i dokumentverktøyet LaTeX (en mer eller mindre identisk mal for OpenOffice er beskrevet i prosjektet *OpenOffice mal for bacheloroppgaven*).

3 emneord:

REST API / NOSQL
Inventar
Maker Movement

Abstract

Coming soon. . . !

Acknowledgements

We would like to thank the following people. . .

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 The Group

The group consist of 4 third year students. 1 informatics 2 digital media production and 1 information-systems. Most of the group have worked with each other on multiple occasions. Be it in student organisation or school projects. 3 member of the group also share apartment. All member have a high interest in the project where they themselves have spent much time.

1.1.1 Thomas Magelssen Bergby

A student who has been interested in everything regarding IT and technology since he was a kid. Thomas has been a leader for “Lær Kidsa Koding” (A group of students who teach coding for kids) and a leader for student assistants in web-development and JavaScript courses.

Thomas studies Informatics, and has gained skills within JavaScript, Java, PHP, CSS, Linux and Python. During his studies, he has taken courses like

- Algorithms and data-structures
- Software Engineering
- Object Oriented Programming
- Android Programming

He also enjoys traveling, and hopes to eventually get a job in the United Kingdom or USA.

1.1.2 Nicolai Naglestad

Studied International Baccalaureate at Skagerak International School in Sandefjord. Nicolai has an above average interest in technology and is always looking for something new to learn. Beside his studies he works as a student assistant in the subjects introduction to programming, web development, object-oriented programming and lastly he works at the schools MakerSpace where the latter is a position where he helps students get started on projects and with the use of the 3D printers. Nicolai has great interest with most aspects the are to be found inside the MakerSpace and you will find him there mosty of the time.

Nicolai studies digital media production at Østfold University College, but has taken subjects such as OOP, Software Engineering and .NET. He also enjoys learning new systems and languages.

1.1.3 Espen Ottar Skjeggestad

He has a broad field of interest, but the main one is IT and Biology. He is an active person that likes jogging, training, diving and trips. He is also politically active in the student politics and has roles as elected representative for the class, member of the student counsel and member of the executive committee for the student-democracy. He is currently a student vara-member of the University College Board. In work roles he was a student assistant for GRIT and is now working at the school library.

Espen studies Information Systems with focus on IT and code, but also includes business leadership and classes about economy.

1.1.4 Simon Chen Dybvik

Simon has been interested in technology his whole life. As a curious child, he often disassembled products to see what's inside and how it worked. He is over average interested in Apple and their products.

During his studies at Østfold University College he has exchanged a semester abroad to California State University, Monterey Bay, where he focused on web development using CMS, JavaScript, jQuery, HTML and CSS, and graphic design. He is former vice-president of NITO Studentene Halden. NITO is a union for engineers and technologists.

Simon studies Information Systems with emphasis in web development. He has taken courses like project management, marketing, business economics and graphic design.

1.2 Employer

The employer for this project is MakerSpace (MS) which is a room located in Østfold University College (HiØ). The MakerSpace is a playroom for creating all types of technology, everything from electronics and robotics, to programming and 3D-printing. The room is currently funded and managed by the IT department.

Here, students and lecturers can use the rooms equipment to experiment with technology to further educate themselves within topics that they find interesting, and that are not necessarily related to any ongoing subject at the university college. MakerSpace is open for all students and staff of the university college, but is mainly used by the IT department.

The employers for this project are Staff Engineer Espen Teigen and University College Teacher Michael Andersen Lundsveen.

1.3 Task

The task of this project is to develop an inventory- and loan-system for Østfold University College's MakerSpace. The purpose of this is to make it easier for employees at MakerSpace to keep track of inventory at all times. A full inventory-system will help both students and staff to find equipment when a student assistant or department Engineer is not available. The system should preferably be able to know where equipment is in MakerSpace at any time. Simultaneously the employers

of this project want to have a system for users of the MakerSpace to be able to loan out the equipment in the MakerSpace.

1.3.1 Purpose

The purpose of this system is to make maintaining the MakerSpace easier for all parties, but mainly for the employees of MakerSpace. This means less time is used for maintaining inventory and helping to find different equipment. This benefits the school in saving money, as the student assistants do not need to be consulted as often. They currently help with mundane tasks like finding equipment and counting inventory, and decide what needs to be ordered to fill up stocks.

1.3.2 Project delivery / Prototype

This group aims to supply the employer a website (front-end) and server (back-end) that is both user tested, and to the employers and users specification.

The website will support the following features:

- View all items (Name, Location, Description, Amount in stock)
- Create/Update/Delete items (CRUD)
- Register/Modify/Delete/View users
- User registration either via custom system or via OAuth 2.0
- The ability to loan set items defined by admins (list editable)
- See currently loaned items (all items or based on user)

Additionally to this there will be a REST API based on Node.js and MongoDB to provide a system for storing the information for the website and possible future applications or other systems.

Documentation

Each separate prototype/system will also include full documentation on how the system is to be used and in the case of the REST API, how it can be used in other systems. This documentation will be hosted on

the same location as where the code is stored (GitHub). As with our main project page the document will be a web page generated by Jekyll hosted by GitHub Pages.

1.3.3 Method

The group will be using the incremental method for development of the system. This method focuses on development piece by piece, and works well for modular systems. It also works for quantitative and qualitative testing of the parts that are done. These parts can also be used, and delivered to the employer.

Incremental method is that you work on one piece of the system at a time. E.g you make the database-system first, and finish it. You can then move on to the next part.

This method has a lower risk of total failure and no delivery, because of the fact that is made up by working pieces.

1.4 Report structure

The report will be structured in the following way

1. Introduction
2. Analysis
3. Design
4. Evaluation
5. Discussion
6. Conclusion

Analysis

This chapter is about what the employer wants, and how they define the task. This is also where the group will address the fact that the project started out big and difficult, to the smaller and more practical project it is today. The project tools and descriptions of these will be discussed here as well.

Design

This is where the design of the project is discussed. The design process, implementations and decisions during the project is addressed here.

Evaluation

TBA

Discussion

TBA

Conclusion

TBA

Chapter 2

List Of Notions

2.1 The task

The task is from a need to get a control over the items that exists in MakerSpace. What they are and an approximately how many there are. It is also a need to get a control on the items student loan. The system will need the student assistants that work on MakerSpace to add and remove items. The boss of MakerSpace will need to have the same right and the possibility to add and remove student assistants. There will not be a need to get an exact cont on items like small leds and screws. But for bigger and more expensive equpment like raspberry pi or drones will need a count.

2.2 MakerSpace

MakerSpace is manly manned with by student assistants Around 3 people. It has a large room with usually many visiting through the day. Many of the student visiting make do by them self but occasionally need student assistants to find something or get advice on how to do a project. Manly the student assistants work with making sure the MakerSpace room is in order. They also have courses in relevant discipline for MakerSpace like 3D printing and drones.

2.3 Program tools

2.3.1 MongoDB

We chose to use MongoDB as a database system because we mainly wanted to learn to use a NOSQL database system. MongoDB type databases, have some strengths that make it easier for us to use when developing a system. MongoDB themselves have a good overview showing the differences of SQL and NOSQL [1] where we can see that this can both help us build the database easier and we should learn and gain experience with this type of system as this might become very relevant for business in the future.

2.3.2 Node.js

Like with our database system it is mostly chosen due to our want to learn and experiment with technologies that we have not used yet in subjects we have had previously. We also looked around at what other had been using to make a REST API and why they used that specific architecture.

We quickly found that most recommended to use Node.js. This was because that it is well documented and is very easy to setup and test (requiring only the server itself and a database server to connect to) other architectures require specific software from the server to run. (As an example a REST API based on Spring/Java require a Gradle or Apache Maven server to run)

2.3.3 Git and GitHub

The group uses Git and GitHub for version control, both for the report and development.

The repositories on GitHub are public for everyone to see and contribute to, since this is a MakerSpace project. All the development is open source, and GitHub is a good platform for this exact purpose.

2.3.4 Latex

The group chose to use LaTeXfor writing the report, and the documentation. The Minutes are also written in LaTeX.

The group chose to use LaTeXover other text editors, because of the usability. When it comes to large projects and reports, LaTeXis superior to other editors.

It is proven [2] that LaTeXis easier to use and more manageable on larger reports and projects, over e.g. Word. When a document becomes complex, it is a lot easier to use LaTeX. See graph (figure 2.1).

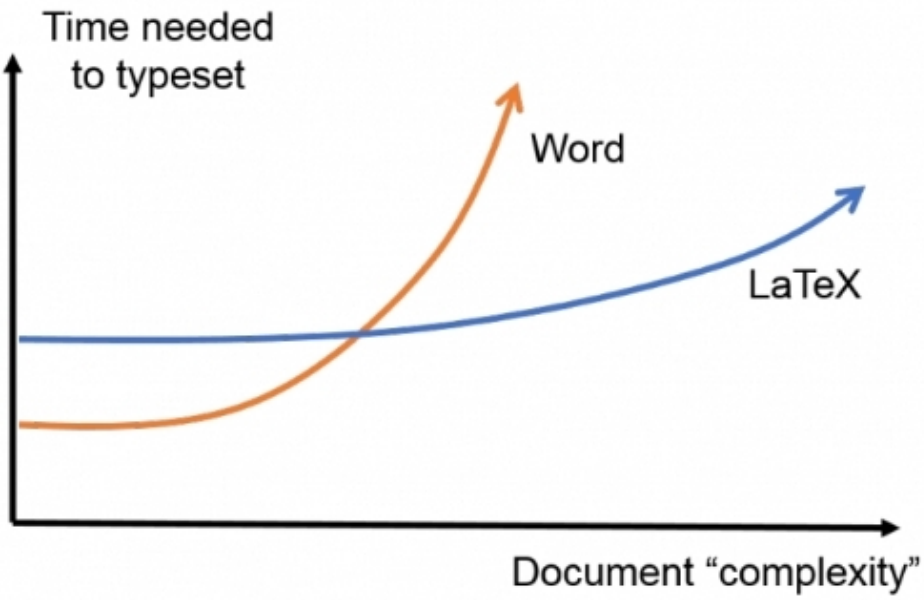

Figure 2.1: Graph about document complexity.

Chapter 3

Analysis

3.1 The task

The task is from a need to get a control over the items that exists in MakerSpace. What they are and an approximately how many there are. It is also a need to get a control on the items student loan. The system will need the student assistants that work on MakerSpace to add and remove items. The boss of MakerSpace will need to have the same right and the possibility to add and remove student assistants. There will not be a need to get an exact cont on items like small leds and screws. But for bigger and more expensive equipment like raspberry pi or drones will need a count.

3.2 User Friendly design

3.3 What is already out there?

The different MakerSpaces around the world use different systems for inventory. After researching online, it seems like it is only Dallas MakerSpace that have any information about it online. They have an inventory system, but it is no longer under development. This shows how an universal inventory system will be useful, not only for the MakerSpace at HiØ but also MakerSpaces around the world. The system is versatile and open source, so it is easy for other MakerSpaces to implement and run the same system.

Even the different MakerSpaces in Norway uses standard shelves

and paper notes on what they have in inventory, and does not have any system for inventory. Since MakerSpace stands for technology and development, a system to keep track of the inventory should be mandatory.

3.4 Open Source

It is more common to develop community software like an MakerSpace inventory system open source, than make everything closed and away from the public. There are many reason why software should be open source, and what makes open source better. MakerSpace Management System is licensed under the MIT License.

3.4.1 MIT License

The MIT License is a license for free software. It is an permissive license, and puts only a few restrictions on reuse of the software. The license originated from Massachusetts Institute of Technology (MIT).

3.4.2 MakerSpace Management System MIT License

The MIT License (MIT)

Copyright (c) 2017- MakerManagement https://makermanagement.github.io/

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED

TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

3.4.3 Security

Software developed open source, tend to have a higher level of security than closed software. Since open source software are viewed by everyone who are interested, errors and flaws of the software will be found more easily and fixed quicker than if the software were closed off.

"Security through obscurity" is a popular statement used by companies and people who wants their software closed off and restricted from the public. Open source is the opposite from this, and encourages that others help and contribute toward the software. A good example for when open source was a good thing , was when the Linux kernel had an bug which caused people to exploit the software[3]. The bug got fixed quickly, because of the Linux community. This would not be the case in bigger, closed off companies. It takes weeks, even months for a patch to e.g Windows.

Using existing systems (HTTP Auth, OAuth2, Etc.)

There are already countless authentication and security systems where they have differ in features and how complex their encryption method is. The benefit of using a an existing system is that it is maintained by as larger group of people and is well tested so when it is fully set up you know it will work well and as described in the documentation, which is also another positive aspect of using an existing system.

A negative aspect of using a existing system is that there is a higher possibility that the system can be hacked/cracked, therefore it will be essential to make sure the system is always up to date. And as many of these existing system allow multiple authentication methods to be used at the same time. (In an example using OAuth2, you can allow login via Google, Facebook and Feide. Here all 3 methods can be attacked

and you are trusting these separate systems that they will not grant a unwanted user access.)

3.4.4 Custom system

Making a custom system can be both more secure and easier to gain access to. This is because as it is a custom system, it can be better tailored to our specific needs. Whereas a existing system will have loads of functionality that we do not need, and these extra function could have vulnerabilities that we are unaware of as we don't use that exact function.

However a custom system requires more upkeep as issues and vulnerabilities could be found after it is created and implemented. In our system we thankfully do not store sensitive information other than a users password, our system would have to make sure that passwords are stored as secure as possible.

Internet vs Intranet

Our employer has requested that the website should only be accessible while at the university, as our system is a listing over equipment in the MakerSpace and could give potential thieves a incentive to steal its equipment. We agree with this request as well support the added security is adds if we move the website to Intranet.

As our system is to be installed at the Østfold University College there are already existing systems that monitor the traffic and log suspect activity as well as all WiFi users being authenticated. In this situation we would just have to make sure that our server logs all activity so if there is a attempt on hacking the system we can use our logs and the universities systems to find and block the hacker.

Only issue here is that we would have to use a local custom authentication system that does not rely on external authenticators, as they all require a connection the the server that a user is trying to authenticate to, so to make sure that the user is actually trying to authenticate with that specific server.

3.4.5 Cost

The cost of open source software will in most cases be notably lower than closed off software. Since software developed open source often uses other open source technologies, it will be practically no cost to run the software.

3.4.6 Support and maintainability

To find support and help around most open source software the user can just search for the problem online. There are large groups of people helping each other in the online community, and it is normally free.

3.5 The Technology

The technology used in developing MakerSpace Management System is chosen out of a couple of different factors.

Since the system is open source and a part of MakerSpace, it is important to use languages and frameworks who are open source and well documented. Another important factor is that the technologies are well supported and easy to maintain.

3.5.1 Front-end

JavaScript / jQuery

JavaScript will be used for the main part of the system. Since the front-end of the system is a webpage, it is only natural to use JavaScript. JavaScript is well documented, well known and fits the needs of the system.

jQuery is a JavaScript framework, but supports more GUI and UI functions which will be useful for the user experience.

Since JavaScript is client side, its less of a load on the server. This means the server can be small and efficient, and can be placed on a Raspberry Pi or similar pocket-size computers.

HTML5 / CSS3

HTML and CSS is the way of showing the information to the user. The whole system is written from scratch, and is following standard conventions from W3C. HTML and CSS are also accessible through all browser and devices, and will always be shown to the user.

PHP7

PHP is a big part of the system and webdesign in general. The system uses PHP for a couple of different reasons, like language support, user authentication, and page control.

One of the big advantages with PHP is the support against API's. The system will utilize PHP for sending and receiving data from the API, and will handle the different HTML forms. PHP also works well with cURL, which is a command line tool which supports HTTP/HTTPS requests[4]. This means the system can ask the API through PHP and cURL, and support full CRUD.

Like JavaScript, PHP is also widely used and well supported.

3.5.2 Back-end

Chapter 4

Design

4.1 Site layout

The site layout is focused on easy recognisability through the use of models based on the business model from sites like komplett.no, Kjell&Company and Thingiverse. This is because the model is well tested and easy to recognize for the user. We choose these kind of sights because they product/item based. The basis is a landing page with featured items with a search bar and usually a menu with categories to the left or on the top of the site. And like most sites a language selection at the top right corner. For each product item/prodct page where there are usually a description, specification, product reviews and if it's in stock and sometimes how many. Out of that information we made our wireframes

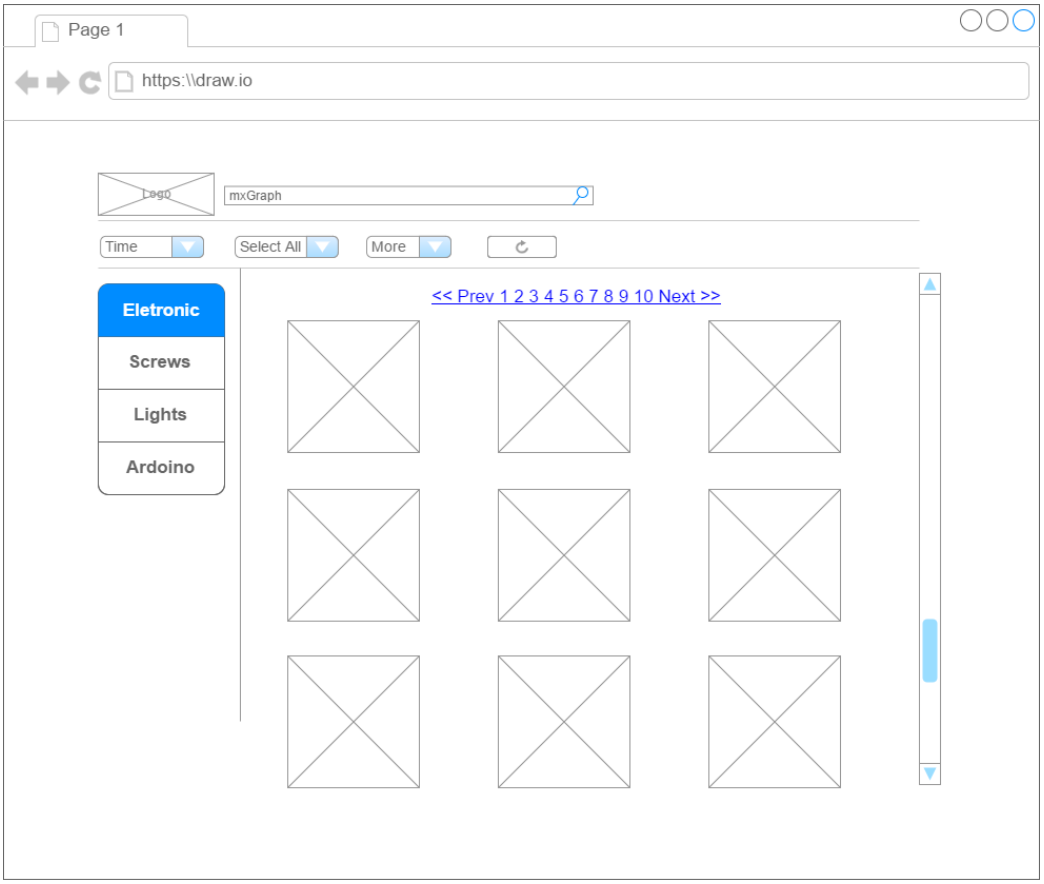

Figure 4.1: Wireframe for the landingpage

The start/landing-page take use of search, categories, and item layout with pictures for each item.

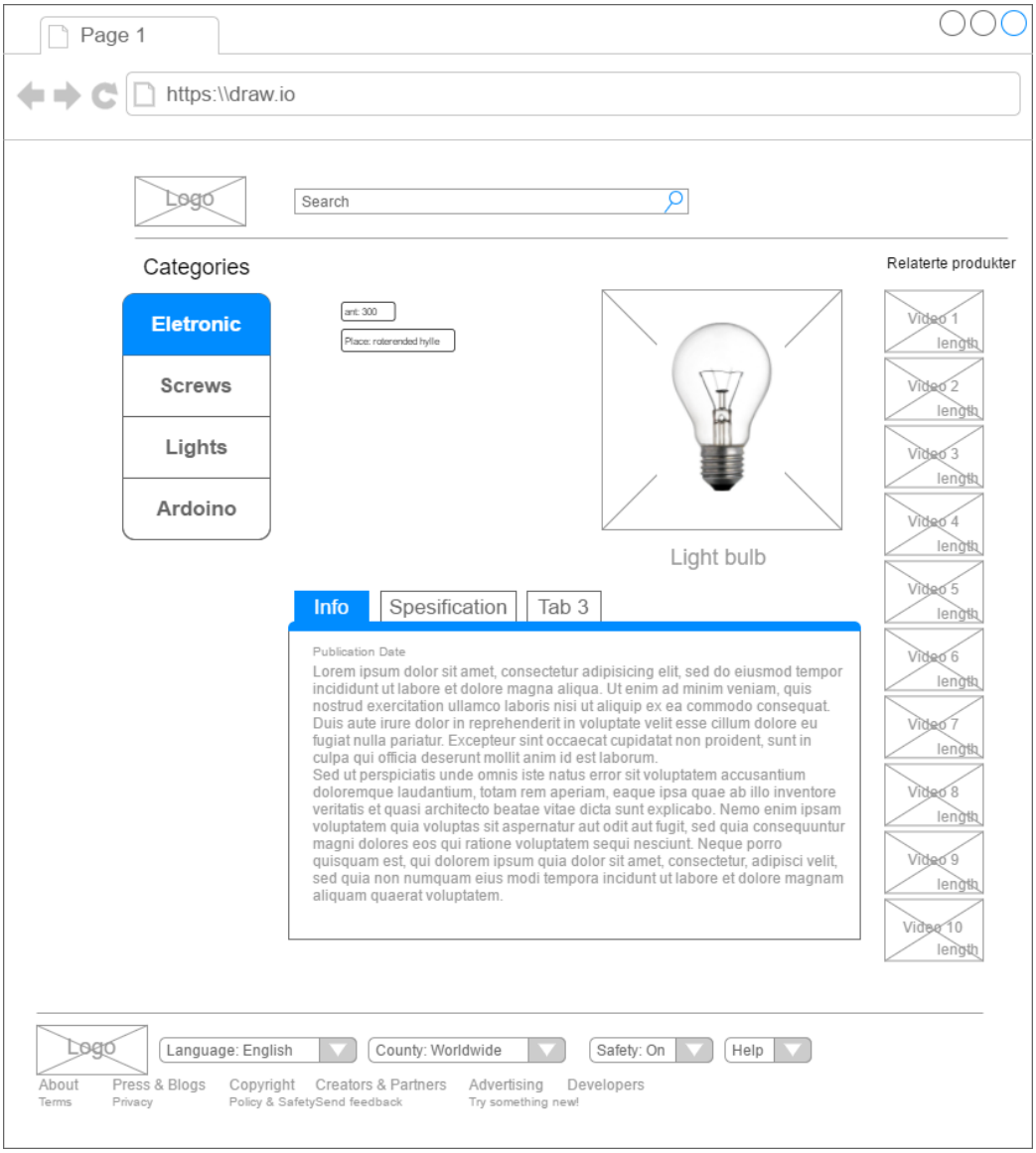

Figure 4.2: Wireframe for the itempage

the itempage make use of a picture, amount, specification box, information box and place for the item.

Originally the group wanted to have a site that had pictures for each item and a lot of information about the items. But after discussion with supervisor about how the workforce of MakerSpace is limited. There are only 2-3 student assistants working at part time. The work that would go inn to making sure each item had a lot of information take a

lot of work. The dangere is if the workload is to big to create items in the page then it might end up that it’s not used at all. The workload to use the site must be so low and mostly effortless so that it’s possible to do on the low work hours that makerspace student assistants have.

4.2 MongoDB

As stated in chapter 3, our database is a NOSQL type database there is no proper way to properly represent the database models. We therefore choose to display them in two ways; the model.js file in the NodeJS server and a output from the REST API in it's default JSON format.

4.2.1 Items

Node.js Model

```
var ItemSchema = new mongoose.Schema(
  {
    item_name: {
      type: String,
      unique: true,
      required: true
    },
    description: {
      en: String,
      no: String
    },
    short_description: {
      en: String,
      no: String
    },
    categories: [
      {
        type: mongoose.Schema.ObjectId,
        ref: 'Category'
      }
    ],
    tags: [
      {
        type: mongoose.Schema.ObjectId,
        ref: 'Tag'
      }
    ],
    locale: {
      type: mongoose.Schema.ObjectId,
      ref: 'Location'
    },
    image_url: String,
    quantity: Number
  }
);
```

JSON Model

```
{
  "_id": "58de638c5edb8f418b9c22d5",
  "quantity": 5,
  "image_url": "http://website.com/image.jpg",
  "item_name": "A Item",
  "__v": 0,
  "tags": [
    {
      "_id": "58b445ecbdc857062a247226",
      "__v": 0,
      "tag": {
        "en": "arduino",
        "no": "arduino"
      }
    }
  ],
  "categories": [
    {
      "_id": "58ac2d03bdc857062a24721a",
      "__v": 0,
      "category": {
        "en": "Arduino",
        "no": "Arduino"
      }
    }
  ],
  "description": {
    "en": "A item.",
    "no": "En ting."
  }
}
```

4.2.2 Category

Node.js Model

```
var CategorySchema = new mongoose.Schema(
  {
    category: {
      en: {type: String, required: true},
      no: String
    }
  }
);
```

JSON Model

```
{
  "_id": "58a3309f170b430a6a835012",
  "__v": 0,
  "category": {
    "en": "Test Category 1",
    "no": "Test Kategori 1"
  }
}
```

4.2.3 Tag

Node.js Model

```
var TagSchema = new mongoose.Schema(
  {
    tag: {
      en: {type: String, required: true},
      no: String
    }
  }
);
```

JSON Model

```
{
  "_id": "58b445ecbdc857062a247226",
  "__v": 0,
  "tag": {
    "en": "arduino",
    "no": "arduino"
  }
}
```

4.3 User Stories

A user story is different scenario of tasks a user, admin or student assistant can do. This will help to understand the functionality of the system.

- As a User
  I want to find a LED bulb
  So I can check the availability
- As a Student Assistant
  I want to reply to messages
  So I can provide service
- As a Admin
  I want to check the availability of a product
  So I can order more if needed

Chapter 5

Implementation

5.1 First iteration

5.1.1 Website (Front-end)

First website version did not call the API (used file based json)

5.1.2 Node.js API (Back-end)

The first iterations of the API only had GET and POST calls of items and GET calls for categories. It would either return all objects or a single object based on the URL.

The following endpoints where available in the first iteration of the REST API:

- GET –> `http://api.url/api/items/`
  - Returns all items
- POST –> `http://api.url/api/items/`
  - Adds a item and returns the new item
- GET –> `http://api.url/api/items/[ITEM-ID]`
  - Returns single item that matches ITEM-ID
- GET –> `http://api.url/api/categories/`
  - Returns all categories

5.2 Utredning

5.3 Mediaproduksjon

5.4 Flowchart

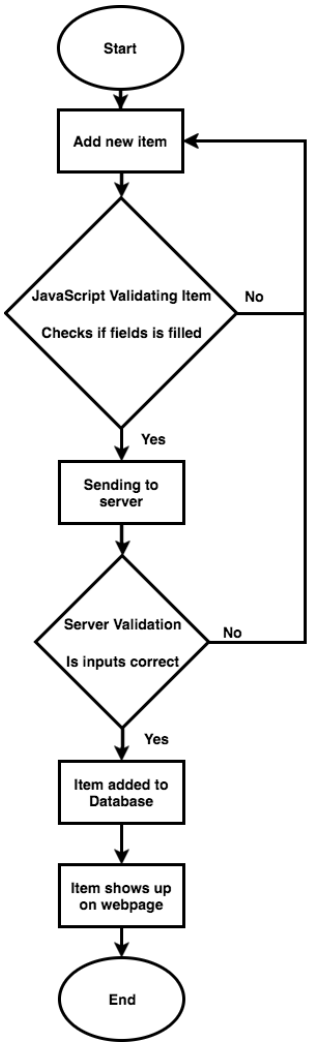

Figure 5.1: Flowchart for adding an item

Chapter 6

Testing

6.1 Why do a user test

A user test is to get the insight needed to make a site user friendly.

The

- Simulates a close to real situation.
- Has concrete tasks for the user.
- Is i test to observe the user.
- Evaluates the user-friendliness of the site.

4.1

6.2 Test goal

Test the site's usability. This include finding item

To ensure the system do what it is supposed to do, we need to verify this through a usability test. The group need to ensure that the new functions work as intended after implementation. As already existing functions can get affected after rolling out the new implementation.

Testing verifies that the system meets requirements and verifications to ensure that the system is built right. Testing helps to validate that the system is being developed for what the user need and expect.

Test the site's usability. This include finding item's, Using categories and adding items.

6.3 Methods

6.3.1 Five Second Test

The Five Second Test is a test to measure the clarity of the site. By showing the test subject the design for only five seconds, and then asking what they recall after viewing it.

6.3.2 Usability Testing

An qualitative method for testing the usability of an product. This be given a user a set of task to do on the product and evaluating the results. It's important not note that it's not the user that is tested, but the product.

6.4 Target Audience

The target audience will be students and employees that do not have as much knowledge about MakerSpace and it's items. This is to make sure the system is optimal to understand too new users.

The target audience will be students and employees at HIOF, that do not have as much knowledge about MakerSpace and it's items. This is to make sure the system is optimal to understand too new users.

The test group will consist of 4 students, where 2 male and 2 female, And 2 employees, 1 representative from each sex.

6.4.1 User Types

Student

Student is the main bulk of the user group. They will be using the system mainly to searches for items, look for information on items and notifying when items are not there or low on stock.

Student Assistant

Student Assistant will be maintainting the inventory. This by adding, editing and deleting - items, categorise and locations.

Employee / Admin of MakerSpace

The admin of MakerSpace will mainly want an overview of the inventory and adding, editing and deleting users. This both be other admin's and student assistants.

6.5 Test Execution

6.5.1 Roles

User

The user will go through different instructions given by the test leader for the site. The reactions and problems the user will encounter will be valuable test data to improve the site.

Test Leader

The test leader will give instructions to the user and note all activity.

6.5.2 Setup

User will in setup be in the same room as the test leader. This is because there are no good test environment available to have the user and the test leader in separate rooms a normal for a normal usertest.

The user will sit at the desk in a environment with as few distraction as possible. The test leader will sit behind the user to not distract and "guide" the user by body language or other non verbal way.

6.5.3 Tasks

1. Find any kind of a led light that a red color.

   The goal of this question is to se if the user will prefer to find the light by using category og use the searcher

2. Find Cosmo and tell us what it can be used for.

   The goal of this question is to se how the user will find an item they don't know how is spelled and if the searchbar will help the user.

3. Change language of the site from English to Norwegian.

   The goal here is to se if the user can with ease find where they change the language of the site, or if this should be emphasised more.

4. Add a random item that the test leader provide to the site.

   The goal is to se if its easy and understandable for the user to add an item to the site.

5. Find Arduino Uno without using search box.

   The goal for this test is to check what the user will use. Category list or the inventory list.

6.5.4 After interview?

After testing the different tasks the test person got some follow up questions regarding the MakerSpace site.

1. Did you get a overview over what kind of items there are in MakerSpace?
2. Did you feel the site, interface and design was easy to understand and figure out?
3. Do you have any suggestions to features that was missing or any improvements to the site?

6.6 Results

6.6.1 Interview

Did you get a overview over what kind of items there are in MakerSpace?

Three of the six people who was in the test group said it was easy to get an overview over the site and the products, and easy to read information from the product boxes. The rest said it was bad visually represented, too much text in the product boxes and that it took a lot of time to understand what MakerSpace inventory was.

Did you feel the site, interface and design was easy to understand and figure out?

Five said it was easy to understand the site, and easy to get an overview. One person complained that the categories was not clear enough.

Do you have any suggestions to features that was missing or any improvements to the site?

Need more focus on a clean inventory site and more items. Need a site with overview over employees at MakerSpace. The search bar and "Load More"-button need to be more highlighted. The "Load More"-button blended too much with the background. Change name to picture in the product boxes, or have picture and name. It was a problem with red border on the dark background, and a dark background generally don't work in a website. One said that the category bar was too small and bad choices on category name. The category names need to have same type of clarification. E.g. "Tools" is a vague category, where "Drones" is more specific.

Chapter 7

Evaluation

Chapter 8

Discussion

Chapter 9

Conclusion

Bibliography

[1] 2017. [Online]. Available: https://www.mongodb.com/nosql-explained.

[2] J. Blanco, *Word or latex typesetting: Which one is more productive? finally, scientifically assessed | computer science | mapping ignorance*, 2015. [Online]. Available: http://mappingignorance.org/2015/04/06/word-or-latex-typesetting-which-one-is-more-productive-finally-scientifically-assessed/.

[3] E. Teo, *Bug 634457 - cve-2010-3081 kernel: 64-bit compatibility mode stack pointer underflow*, 2010. [Online]. Available: https://bugzilla.redhat.com/show_bug.cgi?id=CVE-2010-3081.

[4] S. Pillai, *Curl command tutorial in linux with example usage*, 2014. [Online]. Available: http://www.slashroot.in/curl-command-tutorial-linux-example-usage.

Appendix A

Meeting notes

A.1 Meeting 24-1-17

BO17-G14 Guidance meeting

Minutes for January 24, 2017

Present: S. Børre (Chair), S. Espen, N. Nicolai B. Thomas

Absent:

Reports

The pre-report is delivered and approved

Last meeting points

1. Create pre-report
2. Create project contract
   The minutes of the previous meeting were approved.

New Business

1. Define tools we are going to use
2. Have perimeter meeting with employer.

Next Meeting: Thursday, January 31, at 10:30

A.2 Meeting 31-1-17

BO17-G14 Guidance meeting

Minutes for February 31, 2017

Present: S. Børre (Chair), S. Espen, N. Nicolai B. Thomas

Absent: B. Thomas *(Travelling)*

Reports

Nothing notably to report.

Last meeting points

1. Create high fidelity wireframes

   The minutes of the previous meeting were approved.

New Business

1. Create wireframes
2. Define work roles
3. Meeting with employer to discuss wireframes

Next Meeting: Thursday, February 07, at 10:30

A.3 Meeting 7-2-17

BO17-G14 Guidance meeting

Minutes for February 07, 2017

Present: S. Børre (Chair), S. Espen, N. Nicolai B. Thomas

Absent:

Reports

Wireframes

Landingpage and Itempage — We have created wireframes of landingpages and itempages. These pages have also been discussed with employer.

Last meeting points

1. Create wireframes
2. Have meeting with employer on what the system should contain.

   The minutes of the previous meeting were approved.

New Business

1. Create web page usable for user testing
2. Define therms

Next Meeting: Thursday, February 14, at 10:30

A.4 Meeting 14-2-17

BO17-G14 Guidance meeting

Minutes for February 14, 2017

Present: S. Børre (Chair), S. Espen, N. Nicolai B. Thomas

Absent:

Reports

Webpage

Database — We now have a server. MongoDB is no created and with some dummy-data. It used Json files.

HTML page — The HTML landing page is now created and show dummy data and has a standard navigation menu that follow on all pages.

Last meeting points

1. Check the webpage
2. Defining therms

New Business

1. Start to fill out main report.
2. continue a prototype webpage so we can start user-testing.
3. Start to create a user-test.

Next Meeting: Thursday, February 21, at 10:30

A.5 Meeting 21-2-17

BO17-G14 Guidance meeting

Minutes for February 14, 2017

Present: S. Børre (Chair), S. Espen, N. Nicolai B. Thomas

Absent:

Reports

Site

HTML page —

- The page now speaks with the API
- Created item page and admin page

Database — Continud working on the API

- Fix database form by empty items.
- Fix models
- tarted working on authentication for API

Main report

Added some parts form pre-report to main-report. Gone through the main structure for the template. Checked existing hall of fame reports to se best practise for the main-report.

Last meeting points

1. Start to fill out main report.
2. continue a prototype webpage so we can start user-testing.
3. Start to create a user-test.

New Business

1. Make a design on the webpage to make it ready for user testing.
2. Create the structure and fill out what we can on the main report. Add discussion of why we chose to focus on a easy to update site rather then a heavy administrated site.

F

Next Meeting: Thursday, February 28, at 10:30

A.6 Meeting 28-2-17

BO17-G14 Guidance meeting

Minutes for October 5, 2011

Present: S. Børre (Chair), N. Nicolai, B. Thomas

Absent: S. Espen *(Travelling)*

Reports

Website

- We now have a detailed item view (not all info included)
- Search implemented (still testing)

Meeting discussions

New group member

We have been asked by a member of another bachelor group if he can join our group. We discuss this matter during the meeting, where our supervisor states that this decision is up to us te members of the group. Between now and next meeting we will make a decision if he will join our group or not.

Website / System

We discussed different aspects of the website, what it still needs and how we will solve different issues.

We discussed the following points that we need to implement on the website:

- Items
  - View (done)
  - Item out of stock / messaging system
  - New items (semi done) / Edit items
  - Tags on item page and in search
- Messaging system
  - Item out of stock
  - Loaned item
  - General messages
  - Need assistance
- Box location

A.7 Meeting 28-2-17

- Users
  - Unauthenticated
    - View items
    - Send messages
  - Authenticated
    - Same as Unauthenticated
    - CRUD items
  - Admin
    - Same as Authenticated
    - CRUD news
    - CRUD users
- Login System
  Here it was discussed if we need a complicated login system as the majority of users on the system are unauthenticated users.

Report

The deadline for the report is March 9, but our supervisor states that this date is not that important as we can review the report every meeting.

Until next meeting

1. Continue work on website, to prepare it for user testing
2. Continued work on the report.

Next Meeting: Tuesday, March 7, 10:30

A.8 Meeting 14-3-17

BO17-G14 Guidance meeting

Minutes for March 14, 2017

Present: S. Børre (Chair), S. Espen, N. Nicolai B. Thomas D. Simon

Reports

New member is now in the group and active in all the tools we are using.

Last meeting points

1. Continue work on website, to prepare it for user testing
2. Continued work on the report. The minutes of the previous meeting were approved.

New Business

1. Work more on report. Especially Analysis and implementation
2. Remove most of template fill that is not needed in the main report.
3. Add part about why we changed the direction of the system from high bureaucracy to low bureaucracy.

Next Meeting: Thursday, March 21, at 10:30

A.9 Meeting 21-3-17

BO17-G14 Guidance meeting

Minutes for March 21, 2017

Present: S. Børre (Chair), S. Espen, N. Nicolai B. Thomas D. Simon

Reports

Low productivity. Nothing note worthy to report.

Last meeting points

1. Work more on report. Especially Analysis and implementation
2. Remove most of template fill that is not needed in the main report.
3. Add part about why we changed the direction of the system from high bureaucracy to low bureaucracy.

New Business

1. Work more on report. Especially Analysis and implementation
2. Make sure that the new report always is ready for Mondays.
3. Make sure that all task has defined owner and time limit.

Next Meeting: Thursday, Marsh 28, at 10:30