

Webproject II

Project Report

Isabell Helen Johnsen, Rina Yamada,

Elisabet Therese Bruce Høyér

May 31, 2020

Contents

1	Introduction	2
1.1	Project description	2
1.2	Goals	3
1.3	Report structure	3
2	Designing solutions: Prototyping	3
2.1	Features presented with prototype	4
2.1.1	Students view and choose internships/bachelor projects	4
2.1.2	Student rearranging the order of preference	5
2.1.3	Student application form	6
2.1.4	Application form confirmation	7
2.1.5	Company offer an internship/bachelor project	8
2.1.6	Coordinator/Admin log-in	9
2.1.7	Archive	9
2.1.8	Accept, edit and deny project suggestions	10
2.1.9	Pair students and companies	11
2.1.10	Email and meeting invitations	12
2.1.11	Student List	13
2.2	Universal Design, Accessibility and WCAG 2.0	13
2.3	User Testing	15
3	Implementation	15
3.1	Technologies	16
3.2	Front-end	16
3.2.1	components	17
3.3	back-end	18
3.3.1	The database	18
4	Future work	20
5	Conclusion	22

1 Introduction

Norwegian University of Science and Technology (NTNU) is the largest university in Norway which has 371 study programs and over 40,000 registered students (NTNU, n.d.). With the university's strong authority, they make an effort to build a network between its students and a job market during students' study period. For example, they provide internship opportunities to its students through several courses, and many of bachelor thesis' are done for companies cooperating with the university. However, the process of organising internships and bachelor thesis has been challenging for employers at NTNU due to a manual operation using numerous excel sheets, phone calls, and emails. Several interviews and a presentation was done with Emil Bakke, Terje Stafseng, and Nan Augland to receive insight on the current process.

Product owners are in need of a clearer overview of internships and bachelor projects, companies, and all the students who are applying to and participating in internship or bachelor projects. Furthermore, an automatized process which matches students' applications and companies' favor would be beneficial for the product owners in regard to their work efficiency. It would allow them to spend more time to follow up companies and students with their working progress instead of filling in excel sheets.

1.1 Project description

In this project, we will present a React application where: students can search and apply to internships and bachelor projects, companies can offer internship or projects for bachelor theses, and coordinators at NTNU can have an overview of both students and companies together with students/company history. Coordinators would have passwords and authentications since they will have an access to students' information.

1.2 Goals

The main goal of this assignment is to reduce the workload required from the coordinators related to running internships and bachelor projects. In this project, the goal is to reduce emails and excel sheets by using a database to store information, and application forms for students and companies to use so the coordinators do not need to sort through these applications manually.

1.3 Report structure

The report will start by presenting the prototype and discuss each feature presented within it. This section will also include pictures from the relevant section of the prototype, but the entire prototype can be viewed in the attached document gr1-prototype-imt2671-2020.pdf. This section will also discuss some design decisions made in the prototype, particularly regarding colours and font use.

After this, in section 3, the implementation of the project will be discussed. This includes an introduction to the technologies used both for the front-end and back-end; react.js and php. This section will also include a discussion regarding WCAG 2.0 and how this was kept in mind and followed during the development process.

Section 4 contains a discussion of all future development that should and need to be done for the solution to improve.

Finally, the report will be ended in section 5 with a conclusion regarding the project.

2 Designing solutions: Prototyping

After the presentation from the product owners and the first interview a list of potential features was created. During the creation of this list there was a focus upon reducing all the emails being sent between the project coordinators, the students and the companies hosting bachelor projects or internships. In addition, getting rid of excel sheets to reduce product owners' workload was being considered throughout the development of the application. Particularly the process a student goes through

when applying to a project/internship got extra focus.

2.1 Features presented with prototype

First of all, the application expects 3 types of users: students, companies, and coordinators. As the solution is first being developed for students and companies, the hi-fi prototype only contains the students and company relevant features while the lo-fi prototype also contains the coordinator features. The following features were considered particularly important while making the prototype: a list of internships and bachelor projects, an application form for students, a form for companies to offer internships or bachelor projects, and an overview of companies for the coordinator.

2.1.1 Students view and choose internships/bachelor projects



Figure 1: *an image of the first part of the internship application process for a student, showcasing a list of available internships.*

As seen in figure 1, this page displays a list of internships or bachelor projects on the left hand side of the screen. This list is updated in real time once coordinators accept the offers from companies and the data is added to the database developed

for the application. The list of internships and bachelor projects can be filtered by faculty and/or location.

More detailed information on the currently chosen internship/bachelor project is displayed on the right hand side as a user clicks one of the lists. Students are also able to add the project to the list of preferable projects by clicking the "add to list" button. This button will be changed to a "remove project" button if the project is currently in the students list.

A log-in function for students to view the list of internships / bachelor projects were considered in this process, however it is concluded that the list should be open for students who wishes to read about the project beforehand.

2.1.2 Student rearranging the order of preference

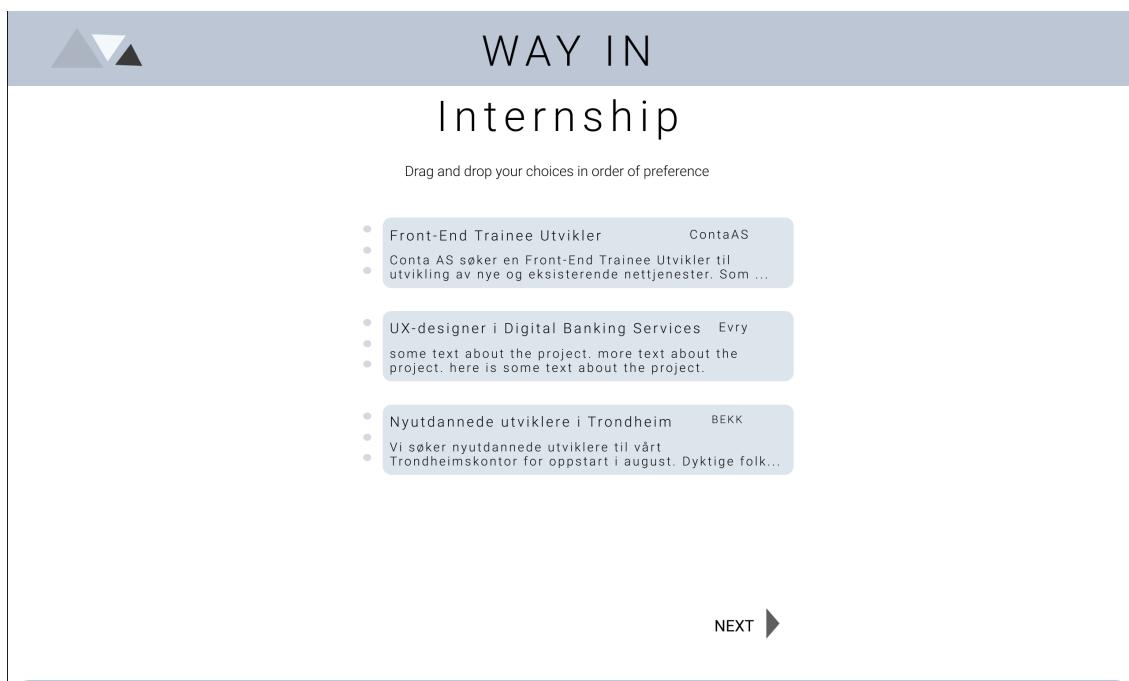


Figure 2: *The second step of the internship / bachelor project application process is to order the chosen projects by preference.*

After students have added 3 projects to their list and press the "next" button as presented in figure 1, they will be led to the page with an overview of chosen projects as it is shown in figure 2. The page makes lets student get a clear overview of their chosen projects in addition to making it possible for the student to change the order

of their chosen projects based on preference. This mimics the way things are done currently, where a student has to write a list of their wanted projects based on preference before sending the application email to the coordinator.

2.1.3 Student application form

- (a) *The application form for students applying to internships* (b) *an application form when applying to bachelor projects*

Figure 3: *The application form for students applying to bachelor projects*

The third step of the application process for students contains an application form. There are 2 different forms depending on what the students' are applying for (internships(figure 3 a) and bachelor projects(figure 3 b)), owing to a bachelor project might involve several students in a group. The internship form only accepts information for one student, while in the bachelor project form it will accept several students. The email input only accepts "@stud.ntnu.no" emails that exist within a database of all students registered to a course that has an internships or bachelor project to prevent misuse.

Once a student has applied to internships/ bachelor projects, this application will be sent to a database, and the coordinator is able to view the information on their log-in pages (presented in 2.1.6-2.1.11). This way the coordinators do not need to send out emails to all the relevant students with the projects nor do they need to sort through their emails to find the applications and move them over to an excel sheet.

2.1.4 Application form confirmation

The screenshot shows a web page titled "WAY IN Internship Application overview". At the top left is a logo consisting of three overlapping triangles. The main title "WAY IN" is in large, bold, black letters, with "Internship" in a larger font below it, and "Application overview" in a smaller font. Below the title, there are two sections: "Chosen projects:" and "Your information:". The "Chosen projects:" section contains three items:

- Front-End Trainee Utvikler ContaAS
Conta AS søker en Front-End Trainee Utvikler til utvikling av nye og eksisterende nettjenester. Som ...
- UX-designer i Digital Banking Services Evry
some text about the project. more text about the project. here is some text about the project.
- Nyutdannede utviklere i Trondheim BEKK
Vi søker nyutdannede utviklere til vårt Trondheimskontor for oppstart i august. Dyktige folk...

The "Your information:" section contains the following fields:

- Name: FirstName MiddleName LastName
- e-mail: StudentName@stud.ntnu.no
- Programme: BPC(Bachelor in programme code)

At the bottom center is a blue button labeled "SEND APPLICATION".

Figure 4: *confirmation of an internship application form before submitting*

The last step of the application process for students is to confirm the contents of the application before pressing the "send application" button. This page reduces the chance of a student sending an application with misinformation or by accident. When the "send application" button is pressed by the user, the page shows "thank you" to assure users that the application form is submitted.

2.1.5 Company offer an internship/bachelor project

The screenshot shows a web-based application form titled "Internship Offer Application". At the top, it says "Please fill in the application form below." Below this, there are several input fields with asterisks indicating required information:

- Company name: [text input field]
- A contact person: [text input field] (first name) [text input field] (middle name) [text input field] (last name)
- Phone number: [text input field] +47 xxx xx xxx
- E-mail address: [text input field] ex) xxx.xx@xxx.xx
- Project name: [text input field]
- Project duration: [radio button] Less than 6 month [radio button] More than 6 month
- Project start: [dropdown menu] Spring 2020
- Project description: [text area]

At the bottom right of the form is a "NEXT" button.

Figure 5: *Companies offering an internship through an application form*

The Application contains a form where companies can offer an internship or a bachelor project, as seen in figure 5. The application form can be only used by companies that are already registered in the database by coordinators beforehand. It limits the use of an application form, however the decision is made based on meetings with product owners, and it is for the reason that they would like to make sure these companies have a right intention and suitable internships/ bachelor projects for students' study programs before they build a partnership. Therefore, companies would be given an company ID by coordinators and will be instructed to use an application form once they reach agreement.

As with the student application process, this feature will reduce the amount of emails and excel sheets needed to be kept by the coordinator once they have an agreement. Here the database will hold all relevant information, and the website presents this to the relevant coordinator in a readable, as well as editable(see 2.1.8) format. Once the projects are accepted by coordinator, students are able to apply to internships / bachelor projects.

2.1.6 Coordinator/Admin log-in

To access the coordinator features, it is necessary to log-in, which is presented in this lo-fi prototype. In the prototype there is not set up an easy way to register a new user, instead this has to be done manually in the back-end.

2.1.7 Archive

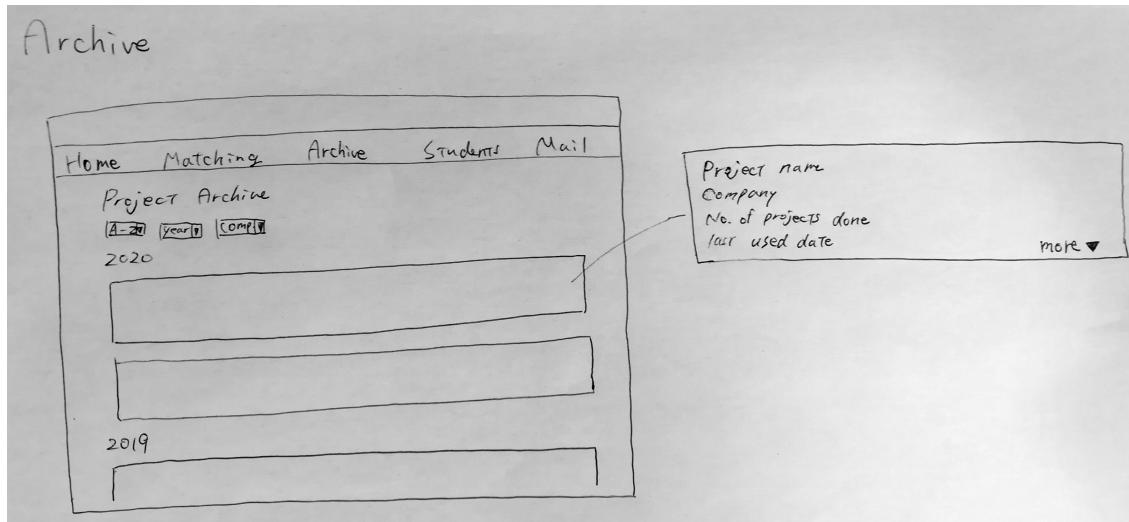


Figure 6: *The lo-fi prototype of an overview/archive page where the coordinator can see an overview of all projects either sorted by year or by company.*

In the archive page, it displays all projects, currently ongoing, finished and not yet started, in an orderly fashion as it is shown in figure 6. Coordinators are able to see which companies have been used a lot recently and which have not been used at all, helping prevent accidental favouritism. By having an overview of the companies together with the date of the last project, the cooperation between NTNU and the companies will be maintained.

2.1.8 Accept, edit and deny project suggestions

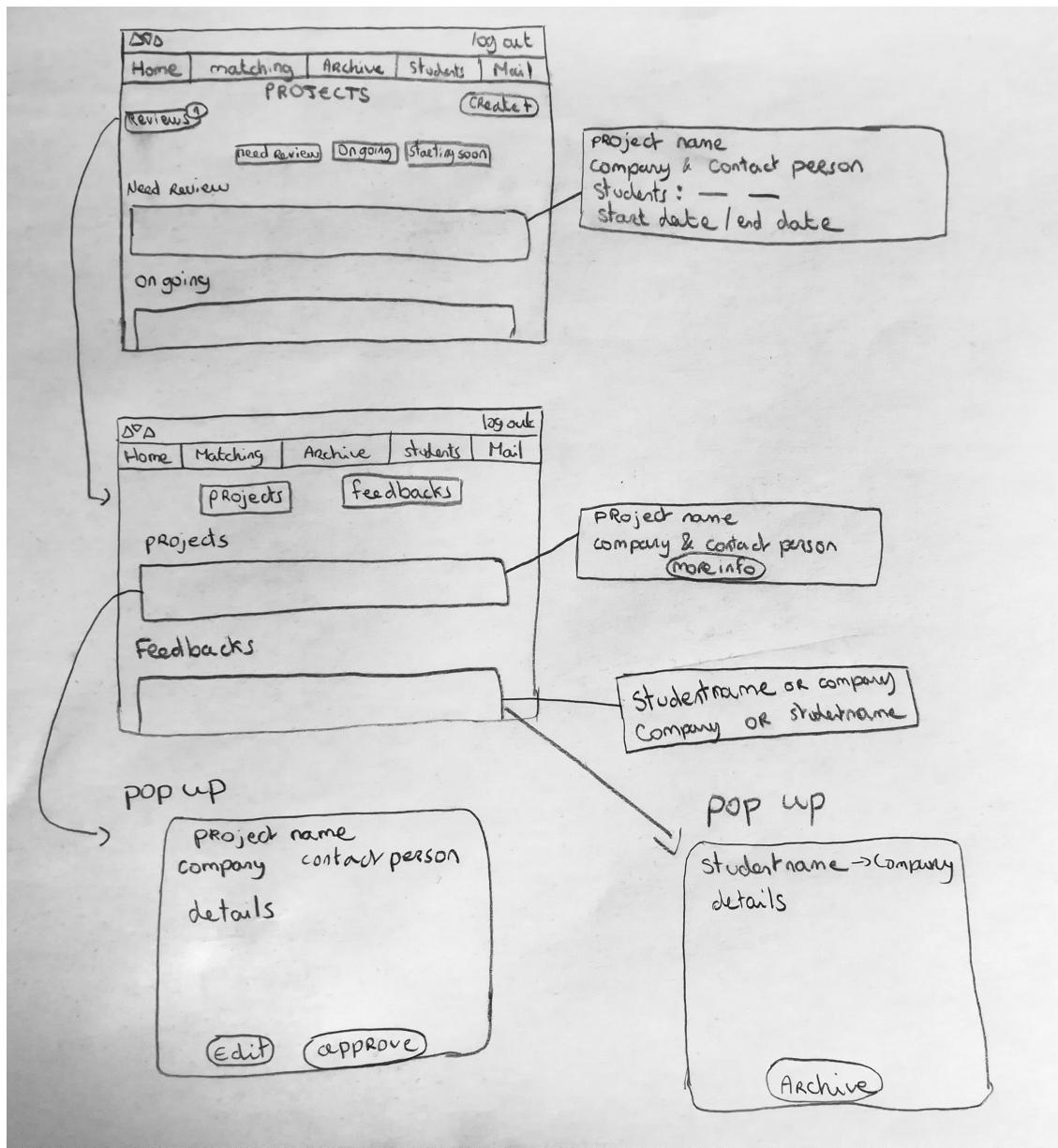


Figure 7: *The lo-fi prototype displaying an idea of what the site will look like when a coordinator has logged in.*

As mentioned earlier, companies will be able to suggest bachelor projects or internships through an application form. A coordinator will be able to see, edit, accept and deny the projects or internships relevant for the course where they are a coordinator as displayed in figure 7. After a coordinator accepts a project, it will be added to a list of internships/ projects where students can view and apply as shown

in figure 1.

As the company representative does not have their authentication process, the coordinator will have the ability to edit the project/internship before and after accepting offers in case something is missing or needs to be reworked.

2.1.9 Pair students and companies

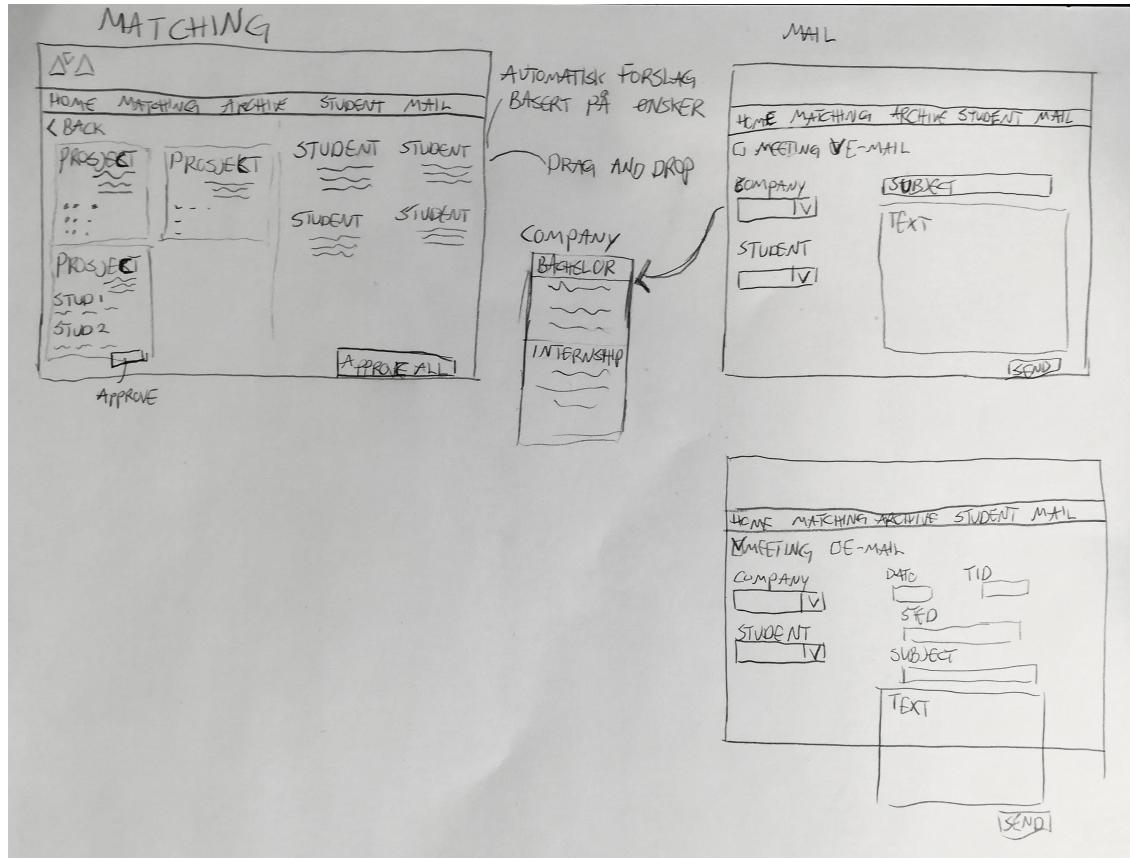


Figure 8: *the lo-fi prototype for pairing up students with projects on the left hand side and for the email function on the right hand side.*

In the lo-fi prototype, a way to pair up students with their selected project/ internship was planned. This would provide a quick overview of each students' and each companies' preferences so coordinators can pair them up accordingly without having to compare several excel sheets. In figure 8 on the left hand side, the lines underneath "student" and "project" symbolises the choices/preferences of the given student and project/company contact.

An algorithm that suggest pair-ups is also kept in mind, as this will reduce the

workload of the coordinators and increase the chances of both students and companies getting their preferred choices. Coordinators can override these suggestions if they choose to do so.

The algorithm, while not developed in the MVP itself, was planned to be fairly simple and focus on having as many "ideal" matches as possible. This means as many students and companies as possible will be paired up with the option they ranked first, and once this is not possible it will aim for second then third.

2.1.10 Email and meeting invitations

Due to something brought up during the meetings with the project owners, an email feature was planned to simplify the process of setting up meetings with students and companies(see figure 8, right hand side). In this feature, it is possible to easily send an email from the Way In application itself. Here the options are to send a regular email with a subject and content, or to send an invitation to a meeting. The meeting feature could possibly use outlook or another external application so it is possible for the recipients to add it to their calendar and for the coordinators to see who has accepted.

In this feature it is also possible to choose between a predetermined group of receivers to reduce the amount of work connected with mass emails. These predetermined groups could, as an example, consist of all students registered to a specific subject or all company contacts with a project registered in a specific subject or of a specific kind.

2.1.11 Student List

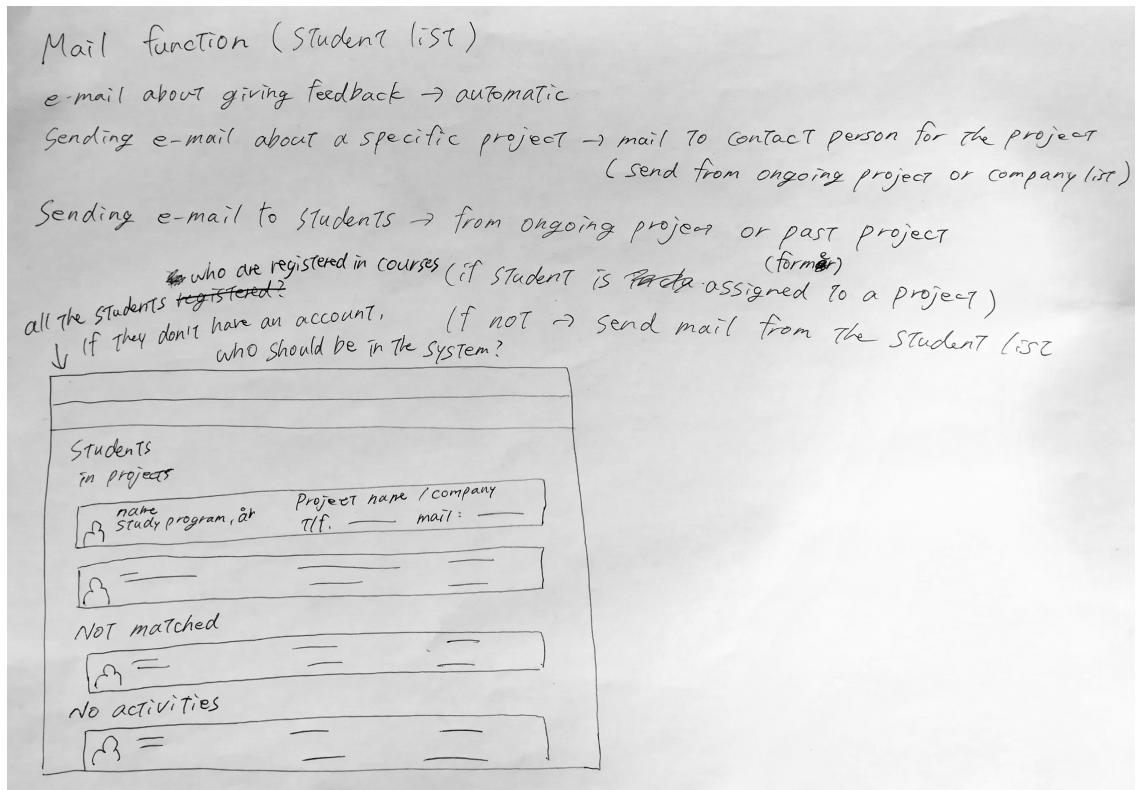


Figure 9: an overview of a list of students in a specific course.

There is also a page displaying a list of all students registered in the course the logged in coordinator is running(see figure 9). This list sort the student into three groups: who who already have an internship/bachelor projects, those who have applied but not yet received one, and those who have not yet applied.

2.2 Universal Design, Accessibility and WCAG 2.0

As the application performs a number of important actions, its design is kept simple to let users concentrate on their tasks. Under the development of Way In application, not only aesthetics, but also universal design was considered.

Web Content Accessibility Guidelines (WCAG) is often used to increase web accessibility of one's website. WCAG 2.0 is based on 4 principals (perceivable, operable, understandable, and robust) which are consisted of 12 guidelines and 61 testable success criteria (W3C, 2008). Norwegian government has imposed laws about uni-

versal design on Norwegian websites in 2013 with an intention of creating a society where everyone can participate regardless one's disability. The laws require websites that target Norwegian citizens to fill at least 35 of 61 success criteria of WCAG 2.0. There are three levels of conformity, A (lowest), AA, AAA (highest), and it requires to fulfill almost all of the criteria marked with level A and AA to meet the minimum requirements (Difi, n.d.).

One of the success criteria of WCAG 2.0 consists of having high enough contrast (contrast ratio 4.5:1 for small texts and 3.0:1 for bigger texts) (W3C, 2008). When testing using Colour Contrast Analyser, the contrast ratio between the background and the text of the application was shown to be 15.37:1 for the body and 12.13:1 for the header which achieve both level AA and level AAA.

The colour palette of the application is mainly consisted of blue grey. The reason for that is the colour blue is considered colourblind-friendly since it appears to be blue even for those with colour vision deficiency.

Colours shall not be the only indication to differentiate important information according to WCAG, therefore there are functionalities such as add/ remove button when users viewing a detailed information of an internship/ bachelor projects. In-

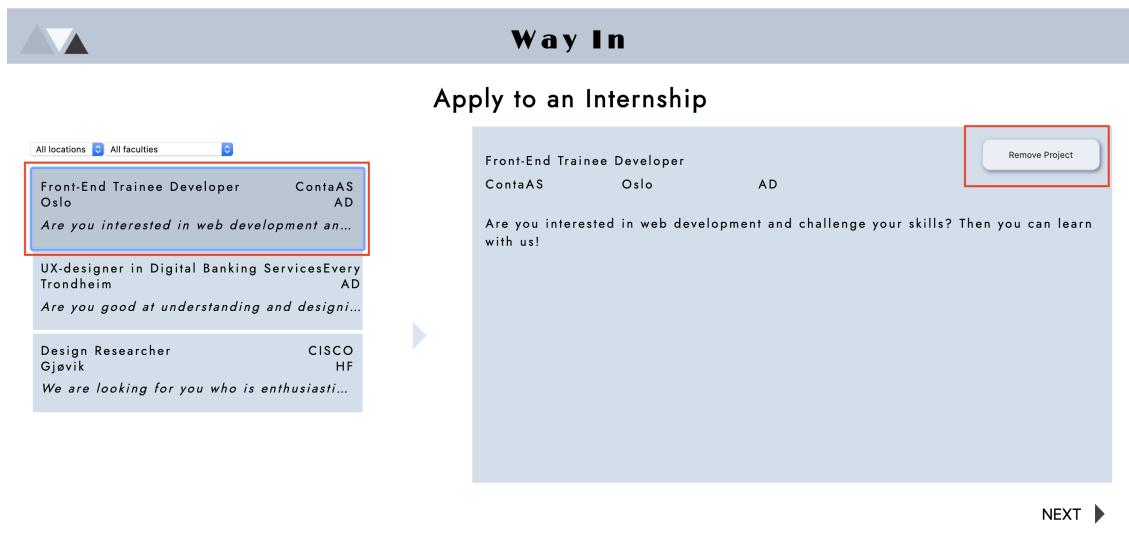


Figure 10: *the text on the button shows "remove project" at the same time as it changes the background colour on the list because the project is added to user's list*

ternships / bachelor projects that are added to user's list will be shown with a darker background colour on a list, in addition to that the button changes its text to "add project" or "remove project" depending on the project is already in the list or not (as seen in figure 10).

Fonts used in the application were both Sans Serif due to the fact that Sans Serif is often preferred using on screens as it keeps legibility even on screens with low-resolution.

While developing the application, HTML5 section elements were used as often as they were relevant to increase the readability particularly for screenreaders. This involved using elements such as <footer>, <header> and <main>. To confirm the usability of the application some practical tests were ran during the end of the development cycle. This involved one round of navigating through the entire page only using keyboard navigation and one round of navigating using the Windows screen reader.

2.3 User Testing

One round of user testing was held early in February as the prototype was being finalized.

Some of the users pointed out that explanations of how the process should be done were missing such as how many projects they were able to choose when students view a list over projects. Some also commented that overview of the whole application process, and showing where they are in the process could have been helpful as well, but otherwise the simplicity of the process got positive responses.

3 Implementation

After building the prototype and doing user testing, a clearer overview of the application was ready.

3.1 Technologies

For the front-end of the solution react.js was used as the main library. To make this work node.js was also installed. Alongside this react-router, a react library, was used to navigate between the different "pages".

In order to set-up the front-end git bash was used to install the different dependencies. The basic folder structure was created automatically through the command 'npm create-react-app way-in'. Similarly 'npm install react-router' was used in the way-in folder to install the react-router library.

As for the back-end solution, Way In application has a mySQL database called "way_in_db". Javascript and php files are used to communicate with the database.

3.2 Front-end

As mentioned react.js was used as the main tool for the front-end of the solution. npm install create-react-app was used to set-up the basic folder structure.

The "public" folder contains the public index.html that every component writes its information to.

The "src" folder is separated into several subfolders. Within the "src" folder itself lies the index.js which is necessary for react router. It's this file that holds all the `|route|` paths and decides which components are shown depending on the url in the browser.

The folders inside the "src" folder and their content is as follows: pictures - holds all graphics and image files used within the solution.

css - holds all the .css files.

fonts - holds all font files.

components - contains all the individual react components as well as another folder: pages.

components/pages - this folder holds all the components that are referred to in the

src/index.js file. Each of these individual .js files corresponds to its own page.

The way_in_db folder contains all .php files necessary to communicate with the database.

3.2.1 components

When deciding what needed to be its own component re-usability was considered. As an example, the header and footer is present on all pages and thus it was logical to have them both be their own component to reduce the amount of code necessary. When working with react.js it is also necessary to consider the hierarchy of components when using 'state' to update the view. It is important to make sure that all the components that needs the same state have the same component as a parent/ancestor so this can be easily shared.



Figure 11: Screenshot of the first page of the internship application page for students. "Front-End Trainee utvikler" has already been added to the list while "UX-designer(...)" is currently being looked at by the user

An example can be seen in figure 11. Here the section on the right containing the full project information and the project list on the left are siblings, children of the same component, as they both need to collect the same information. The component on the right has the "add/remove project" button that adds/removes the project from the project list, which is then saved in the state of its parent. The

component on the left then needs to access this to know if it's in the project list or not as the appearance changes(darker background) to display this change to the user.

Figure 11 is the screenshot of the page when a student is applying to an internship. When a student applies to a bachelor project the design and basic functions of the solution is the same so the same components are re-used for this section. The difference here lies in the components/pages .js files. studInternshipPage.js passes the necessary php file to collect internship information down as a property to its children, while studBachPage passes the necessary php file for the bachelor projects. They both contain and display the same component, but because information is collected from a different section of the database different projects are displayed and applied to.

3.3 back-end

The back-end was created using php, AJAX and mySQL as part of IMT3851 Programming for web II. A database named "way-in" is used for this project, and all of the data displayed on the application is collected from it by using php and javascript.

3.3.1 The database

The database was built using mySQL. Having a database connected to the application is necessary as it will hold project and application information. By utilising a database alongside forms and an application process the need for external emails and excel sheets will be reduced.

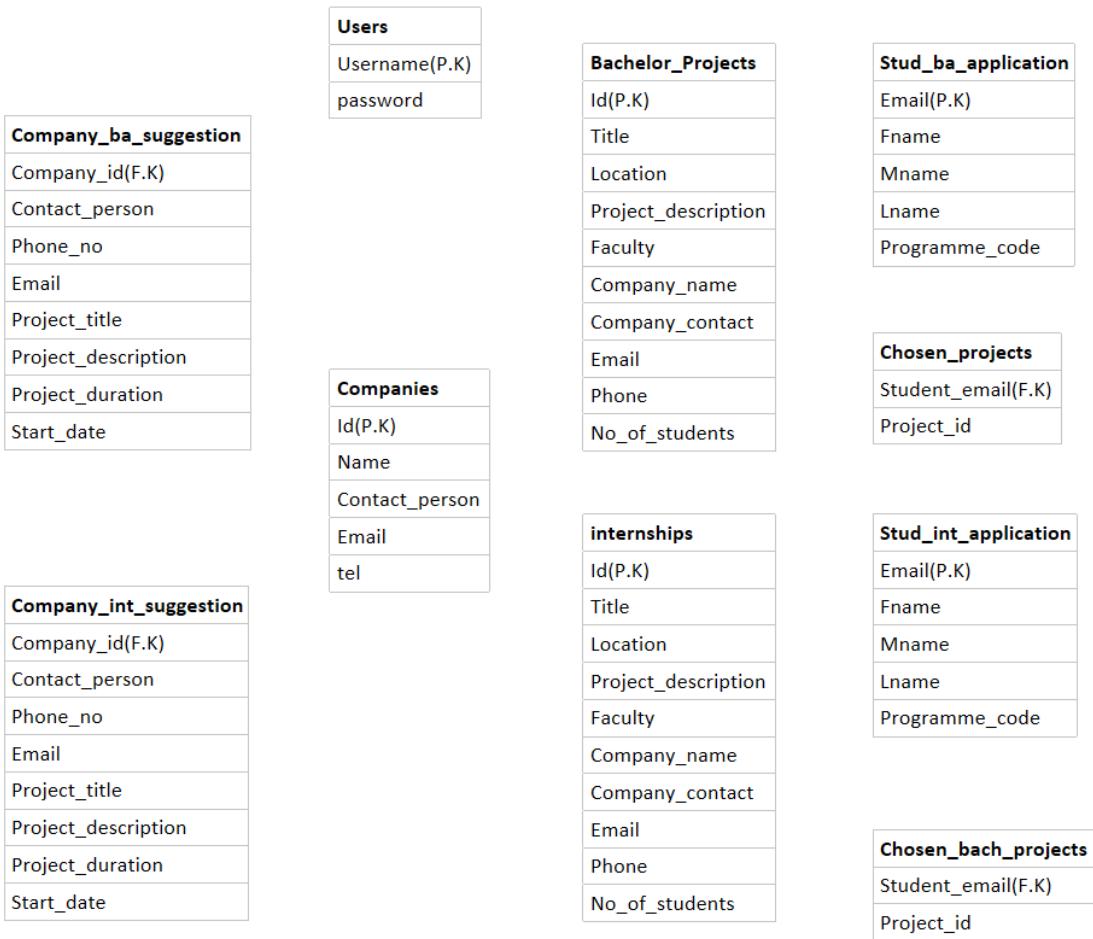


Figure 12: A relational model of the database before it was developed.

A relational model for the database was designed before building it. This model can be seen in figure 12. While working on the forms and building connections to the database it became clear some columns were missing and an entire table was not present. The relational model of the final database can be seen in figure 13. This database contains all tables and columns necessary for the current version of the application.

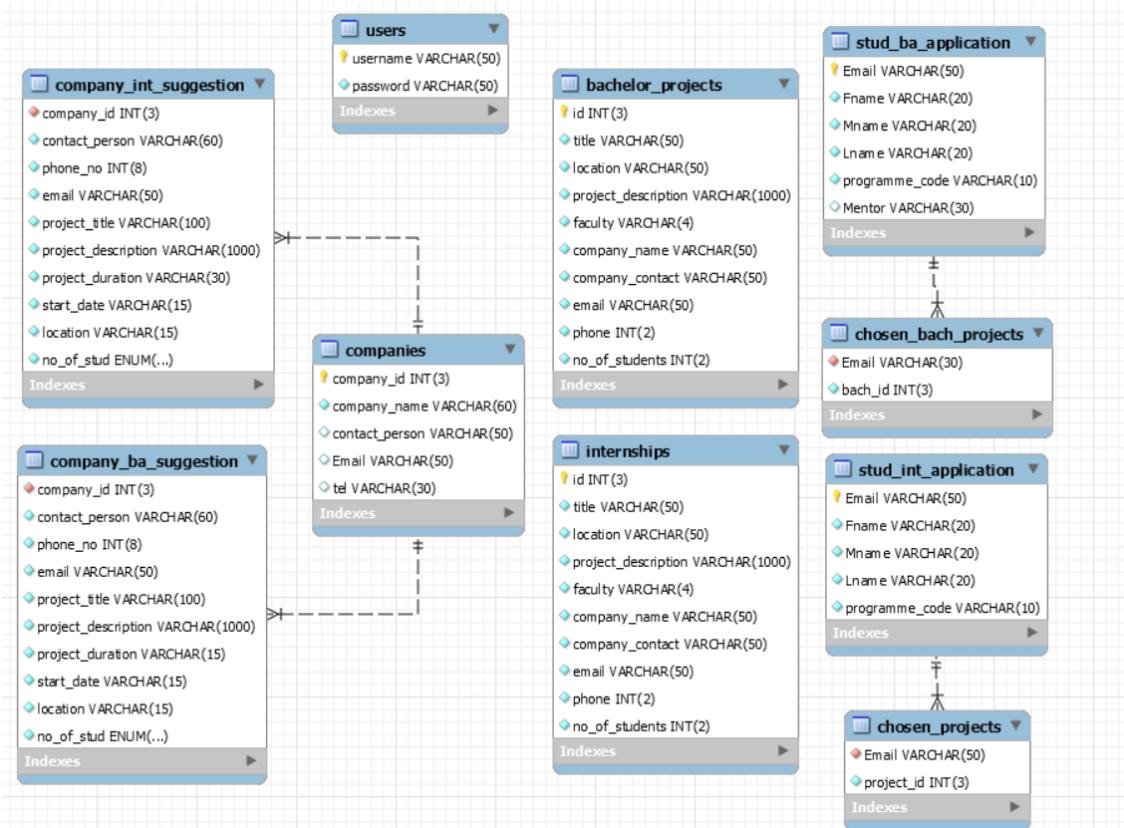


Figure 13: *The relational model for the current database, reverse engineered from the current .sql database file using mySQL workbench.*

4 Future work

There are several features presented in the Prototype chapter that does not yet exist in the application itself.

One of these is the page specifically for students to arrange the chosen projects by preference. As it stands currently the projects are in the order they were added, so the first project added gets to be no. 1, the second is no. 2. So in the current solution the students need to remove and re-add projects to get their preferred order which is not user friendly.

The length of the application process as a whole is also not very visible to the student who is applying, as there is no visual clues to which "step" of the process a user is on. Adding "step 1 out of 3" somewhere on the page will make the process more user friendly.

There are many admin functionalities that are missing in the current solution as well, like the ability to pair up students with projects and companies as explained in 2.1.9 "pair students and companies". This feature would also be quite high on the priority list as it helps reduce the workload for the co-coordinators, particularly with the use of an algorithm to suggest potential pair-ups.

Another functionality in the prototype that is not present in the MVP is the email and meeting invitations feature mentioned in 2.1.10.

For the prototype and the MVP only a coordinator log-in is prototyped and developed. It would be necessary to also develop its own view for one or several admins. This admin, or admins, would have their own unique privileges like the ability to create admins, delete accounts or even projects when necessary.

Aside from this, creating a user registration for new coordinators would be useful to reduce some workload for the admin. Another solution to potentially look into would be to see if there can be some sort of cooperation with one of the log-ins already existing within NTNU, like Feide.

Due to the application forms being open, a stronger form security than what is currently present needs to be developed. While there are some stoppers in the forms to prevent misuse, like only accepting student emails, there is still the risk of misuse.

Even though there has been some user testing, there was only one round of it. More extensive user testing needs to be done before any new features are designed and developed, as well to improve the currently existing features. Getting feedback on the current solution from the product owners would also be ideal before further development. This is to make sure the current solution solves its problem well. It's also necessary before creating a new plan of action regarding the future development.

5 Conclusion

While the application allows the each user type to perform their crucial tasks, a number of features are yet to be implemented. Aside from the existing functionalities, it still requires coordinators to work manually as only the application process has been automated. Moreover, although the application manages to eliminate some of the emails and excel sheets, the absence of admin functionalities prevents the application from being as efficient as it could be. In order for the application to be in use, further development is essential as well as a long-term plan.

References

- Difi (n.d.). *WCAG 2.0-standarden*. Available at: <https://uu.difi.no/krav-og-regelverk/wcag-20-standarden>. (Accessed: 31 May 2020).
- NTNU (n.d.). *NTNU – tall og fakta*. Available at: <https://www.ntnu.no/tall-og-fakta>. (Accessed: 31 May 2020).
- W3C (2008). *Web Content Accessibility Guidelines (WCAG) 2.0*. Available at: <https://www.w3.org/TR/WCAG20/>. (Accessed: 31 May 2020).