

Studie av automatisk justering av  
skjerminnhold avhenging av  
skjermstørrelse og kontekst.

**Njaal Christoffer Gjerde**  
**Mathias Moen**

Master i informatikk

Innlevert: november 2013

Hovedveileder: Terje Rydland, IDI

Norges teknisk-naturvitenskapelige universitet  
Institutt for datateknikk og informasjonsvitenskap



---

*We would like to thank our friends and family for their continued support. A special feeling of gratitude to our loving girlfriends Marte and Emily for their loving support. They have always been there for us for support.*

*We also dedicate this dissertation to our fellow students at the graduate study room Fiol for being there for us throughout our master program. We have shared many rich conversations, and they have given us comprehensive feedback on our work throughout our master thesis.*

*In addition, we want to thank our university: The Norwegian University of Science and Technology for five good years of learning. We would also like to thank Bjørg Danielsen for her help with delivering statistics of the NTNU website.*

*We wish to thank our supervisor Terje Rydland for his rich insight and directions. Without him, this thesis would not exist.*

---

---

---

---

# Abstract

**English** The use of handheld devices has increased greatly over the last years. As an effect of this, Responsive Web Design (RWD) has gained increased popularity and usage as well. Still, responsive design is a quite new concept, so the majority of websites are not responsive.

This thesis will try to explore the concept of Responsive Web Design (RWD) as much as possible. The theory of responsive web design itself will be looked at, mainly based on the book "Responsive Web Design" from Ethan Marcotte, the creator of the term Responsive Web Design. Additionally, some existing websites will be inspected to better understand what is common practice today.

A website which is not responsive will be taken as a starting point, a redesign process will be conducted on this website. An exploration of different frameworks for creating responsive websites will be done. To get an overview of different technology and frameworks, at least one prototype will be made using HTML, CSS and JavaScript.

---

---

**Norwegian** Bruken av håndholdte enheter har økt kraftig de siste årene. Som en effekt av dette, har Responsivt Web Design (RWD) fått økt popularitet og bruk. Likevel er responsivt design et ganske nytt konsept, så de fleste nettsted er ikke responsive.

Denne oppgaven vil forsøke å utforske begrepet Responsivt Web Design (RWD) så mye som mulig. Teorien om responsivt web design i seg selv vil bli sett på, i hovedsak basert på boken "Responsive Web Design" fra Ethon Marcotte, skaperen av begrepet RWD. I tillegg vil noen eksisterende nettsteder inspiseres for å bedre forstå hva som er vanlig praksis i dag.

Et nettsted som ikke er responsivt vil bli tatt som et utgangspunkt, en redesigningsprosess vil bli gjennomført på dette nettstedet. En undersøkelse av ulike rammeverk for å skape responsive nettsteder vil bli gjennomført. For å få en oversikt over teknologi og rammeverk, vil minst en prototype lages ved hjelp av HTML, CSS og Javascript.

---

# Table of Contents

<b>Abstract</b>	<b>3</b>
<b>Table of Contents</b>	<b>7</b>
<b>Abbreviations</b>	<b>10</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Background . . . . .	1
1.2 Motivation . . . . .	2
1.3 Research questions . . . . .	3
1.4 Research method . . . . .	3
1.5 Results . . . . .	4
1.6 Outline of the report . . . . .	4
<b>2 Research method</b>	<b>5</b>
2.1 Design and creation . . . . .	5
2.2 Systems development methodology . . . . .	8
2.3 User-Centered Design . . . . .	9
2.4 Questionnaire . . . . .	9
2.5 Interviews . . . . .	10
2.5.1 Group interview . . . . .	10
<b>3 Literature Review</b>	<b>13</b>
3.1 Responsive Web Design . . . . .	13
3.1.1 Flexible Grid . . . . .	14
3.1.2 Flexible Images . . . . .	16
3.1.3 Media Queries . . . . .	16
3.2 Responsive frameworks . . . . .	17
3.2.1 Bootstrap 3.0 . . . . .	17
3.2.2 Foundation . . . . .	18
3.2.3 Unsemantic . . . . .	18
3.2.4 Skeleton . . . . .	19
3.2.5 Masonry . . . . .	20
3.2.6 Framework summary . . . . .	20

---

3.3	Mobile first . . . . .	21
3.4	Inspiration . . . . .	22
3.4.1	Ideas . . . . .	22
3.4.2	Higher education . . . . .	23
<b>4</b>	<b>Prestudy</b>	<b>31</b>
4.1	Introduction . . . . .	31
4.2	Personas . . . . .	33
4.3	Scenarios . . . . .	35
4.4	Dividing the site . . . . .	37
4.5	Content . . . . .	38
4.6	Resolutions . . . . .	39
4.7	NTNU renewed their website design . . . . .	40
<b>5</b>	<b>Process evaluation</b>	<b>43</b>
5.1	Literature review . . . . .	43
5.2	Prestudy . . . . .	44
5.3	Initial design . . . . .	44
5.4	First interviews . . . . .	45
5.5	Implementing the prototype . . . . .	45
5.6	Feedback on the prototype . . . . .	46
<b>6</b>	<b>Data gathering</b>	<b>49</b>
6.1	Questionnaire results . . . . .	49
6.2	Interview results . . . . .	51
6.2.1	Experienced students . . . . .	51
6.2.2	New students . . . . .	52
<b>7</b>	<b>Design</b>	<b>53</b>
7.1	Initial design . . . . .	54
7.1.1	Internal design . . . . .	55
7.1.2	External design . . . . .	57
<b>8</b>	<b>Implementation</b>	<b>61</b>
8.1	Introduction . . . . .	61
8.2	Prototype design . . . . .	61
8.3	Programming without frameworks . . . . .	62
8.3.1	The layout . . . . .	63
8.3.2	Challenges . . . . .	64
8.3.3	The result . . . . .	69
8.4	Programming With frameworks . . . . .	73
8.4.1	Layout . . . . .	73
8.4.2	Advantages . . . . .	74
8.4.3	Challenges . . . . .	76
8.4.4	The result . . . . .	77

---



---

<b>9 Results</b>	<b>81</b>
9.1 Introduction . . . . .	81
9.2 Structuring the website . . . . .	81
9.2.1 Alternative prioritization . . . . .	82
9.2.2 Alternative class schedule . . . . .	83
9.3 Making a responsive website . . . . .	84
9.3.1 The process of making the site . . . . .	84
9.3.2 Suggested development process . . . . .	87
9.3.3 Should a framework be used? . . . . .	88
9.3.4 Missing functionality . . . . .	88
9.3.5 Which framework should be used? . . . . .	89
<b>Bibliography</b>	<b>89</b>
<b>Appendices</b>	<b>95</b>
<b>A Interview with Jan Erik Kaarø, NTNU info</b>	<b>99</b>
<b>B Questionnaire</b>	<b>101</b>
<b>C Implementation without frameworks</b>	<b>105</b>
C.1 index.html . . . . .	105
C.2 index.css . . . . .	111
<b>D Implementation with framework</b>	<b>123</b>
D.1 index.html . . . . .	123
D.2 index.css . . . . .	127

---

# List of Figures and Tables

2.1	Design and creation steps . . . . .	7
2.2	Our research process . . . . .	7
2.3	Prototyping model . . . . .	8
3.1	Fluid grid . . . . .	14
3.2	Foundation framework, off canvas layouts . . . . .	18
3.3	Unsemantic framework, grid elements . . . . .	19
3.4	Skeleton framework, model . . . . .	19
3.5	Masonry framework, the grid . . . . .	20
3.1	Responsive frameworks summary . . . . .	21
3.6	Smartphone Vs Desktop shipment . . . . .	22
3.7	Hamburger icon . . . . .	24
3.8	UiO's adaptive web site . . . . .	25
3.9	UiB's responsive web site . . . . .	26
3.10	BI's responsive web site . . . . .	29
4.1	Ntnu.no viewed on different devices . . . . .	32
4.2	Two user groups, two url's . . . . .	38
4.3	Most viewed pages on ntnu.no . . . . .	39
4.4	Percentage of ntnu.no accessed by phone, tablet and PC . . . . .	40
4.5	Updated ntnu.no design viewed on different devices . . . . .	41
6.1	Smartphone growth in Norway . . . . .	50
6.1	Top 10, different platforms . . . . .	50
7.1	The screen size problem . . . . .	54
7.2	Sketches: Mobile, internal user group . . . . .	55
7.3	Sketch: Tablet, internal user group . . . . .	56
7.4	Sketch: PC, internal user group . . . . .	57
7.5	Sketch: Mobile, external user group . . . . .	58
7.6	Studies section, old NTNU website . . . . .	59
7.7	Sketch: Tablet, External user group . . . . .	60
7.8	Sketch: PC, External user group . . . . .	60
8.1	Touch areas on a mobile phone . . . . .	63

---

8.2	The first two elements in the design	65
8.3	The #Toprightwrapper at 767px wide resolution	66
8.4	Change priority of div	67
8.5	The prototype displayed in 769px	69
8.6	The prototype displayed in 752px	69
8.7	Non framework 1463px	70
8.8	Non framework 1084px	70
8.9	Non framework 779px	71
8.10	Non framework 660px	72
8.11	The mobile version displayed in two ways	73
8.12	Bootstrap column system	73
8.13	Styled buttons in Twitter Bootstrap 3.0	75
8.1	Resolutions, Twitter Bootstrap	76
8.14	Bootstrap classes when inspecting	77
8.15	Framework at 1527px	78
8.16	Framework at 799px	79
8.17	Framework at 382px	80
9.1	Priority list for an alternative approach to separating user groups	83
9.1	Priority change example	85
9.2	Shuffle feature, Gumby	86
9.3	Blank space	89
B.1	Questionnaire, question 1	101
B.2	Questionnaire, question 2	101
B.3	Questionnaire, question 3	102
B.4	Questionnaire, question 4	102
B.5	Questionnaire, question 5	102
B.6	Questionnaire, question 6	102
B.7	Questionnaire, question 7	103
B.8	Questionnaire, question 8	103
B.9	Questionnaire, question 9	103

---

---

# Abbreviations

RWD	=	Responsive Web Design
CSS	=	Cascading Style Sheet
JS	=	JavaScript
HTML	=	Hypertext Markup Language
CMS	=	Content Management System
jQuery	=	JavaScript library that simplifies JavaScript programming
NTNU	=	Norwegian University of Science and Technology
UiO	=	The University of Oslo
BI	=	BI Norwegian Business School
PC	=	Personal Computer
MAC	=	Macintosh
PhD	=	Doctor of Philosophy
Div	=	Shortening of division. Used as a generic block level division in HTML.
IEEE	=	Institute of Electrical and Electronics Engineers
PX	=	Pixel

# Introduction

This thesis will explore the possibilities of a website utilizing Responsive Web Design (RWD) techniques. We will try to understand why it has become so popular the last couple of years, and try to figure out what possible gains a website will receive from becoming responsive. Going through the complete development process we will create a responsive website starting from the early requirements, through the design phases, and at last to finish a working prototype. This will be done through research, and by creating a prototype to test the technology and its different features. We will look at different best-practices in the Responsive Web Design world, and see how they can be incorporated into our prototype to give the users the best experience possible.

## 1.1 Background

On May 25, 2010 Ethan Marcotte posted an important article[1] that gave us a different view on web design. This article is where the term Responsive Web Design was first introduced. Marcotte writes about limitations on current websites, and compares the web as an opposite to building architecture. A building architect aims for eternity, but with the web it often feels like aiming for the next week.

*"A building's foundation defines its footprint, which defines its frame, which shapes the facade. Each phase of the architectural process is more immutable, more changing than the last(..) Working on the web, however is a wholly different matter. (..) Inconsistent window widths, screen resolutions, user preferences, and our*

*users' installed fonts are but a few of the intangibles we negotiate when we publish our work.(..)"*

Ethon Marcotte

Marcotte continues to talk about how the increase of mobile devices affects what companies orders from the designers and developers. They order specific "iPhone websites". But the range of devices and screen resolutions increases, tablets and phones in many sizes, video game consoles with browsers and so on. Creating separate websites for all devices is simply not an option, and the solution is of course Responsive Design.

## 1.2 Motivation

Quite early in the process of deciding what our master thesis should contain we talked about Responsive Web Design as an interesting topic. It is a technique more and more web developers have been accustomed too, over the last couple of years. It is also becoming a necessity with the rise of smaller screen devices, mainly because of our increased use of the smartphone browser. The focus on web front-end (HTML, CSS and JS) has also been non existent on the university, so we wanted to get more experience in this field. To get some hands on experience with the technology we had to find a suiting application for this "new" design method. We considered many different website areas, mainly sites that we as students visit. Websites for it-consultant firms and online stores also seemed interesting. The main issue we saw, was how to gain access to both statistics and regular users. To solve this problem we thought a school website could be the solution to go for, and the NTNU website came up as an obvious choice.

After realizing that our own university's website was an option, we immediately saw even more advantages. After five years of using this site, we have gained a lot of knowledge on how to use the website. The website is also not using any responsive design, and is definitely not optimized for smaller devices. We have even ourselves been in situations where we have been frustrated over the poor user experience of the website on smaller devices like smartphones and tablets. The full desktop version is also troublesome to navigate. Additionally, it is our university, so any improvement would also be in our benefit. Summing up the NTNU website would give us the advantage of:

- Easy access to statistics
- Easy access to the users
- Five years of experience using ntnu.no

## 1.3 Research questions

Our main focus is to redesign the NTNU website, and to make it responsive. We want to look on the users needs and what functions are important in different contexts. We also want to look at the technology, and try to gain some insight to how we should approach the problem when using different design techniques. We therefore chose to divide the research questions into two separate parts. The first research question takes care of how we could structure the website, and the second research question will answer how a website like this should be created.

1. How should the website be structured?
  - (a) How can we organize the website to accommodate the different user groups?
  - (b) Which functions or information should be prioritized on the different devices?
2. How should the website be created?
  - (a) Which technologies and frameworks could be used to improve the process and the result of a responsive NTNU website?
  - (b) Are there any benefits or disadvantages with the different technologies and frameworks?

## 1.4 Research method

A literature review will be the starting point of our master thesis. The next step will be to analyze the NTNU website and gain as much information possible about what functions the site has and where we can find the different features. Conducting a survey will be the next step. Gaining knowledge on how different users experience the website will be the goal of this survey. To get a complete image of the user mass, we will try to gain information from multiple user groups, to get a good basis for further research. This survey, supplemented with our own knowledge of the domain, will be the basis for the first edition of the design.

Interview sessions with the different user groups will be the next step in the process. Here the interviewees will get the chance to look at, and evaluate the design sketches and give us input on what we can improve.

## **1.5 Results**

The results we want to gain from this study is both theoretical and technical. It will consist of at least one working prototype that displays the NTNU website with responsive features. Since the NTNU website has so much information, we are limiting ourself to the front page in our prototype. We will also explore different types of frameworks and technologies that can ease the process of creating a responsive website.

## **1.6 Outline of the report**

This report starts to explain our research method in Chapter 2. We discuss how we have chosen to work on the thesis, and why. We explain the choices of our data gathering methods. Further, we have the literature review in Chapter 3, which explains responsive web design in depth, as well as responsive frameworks, mobile first and then our ideas for a website to implement.

The planning of the design and implementation is presented in Chapter 4. An evaluation of our own process, including prestudy, design and data gathering is discussed in Chapter 5. The results from our questionnaire and interviews are presented in Chapter 6.

In Chapter 7 we present our initial design for the website, and explain briefly to why we chose this design. At the end of the thesis we have the implementation and result chapters, Chapter 8 and Chapter 9. In the implementation chapter we explain the process of creating the two prototypes. We look at the two implementations and discuss advantages and challenges, as well as explaining the layout and result.

In the results chapter, we try to answer our initial research questions, as well as discussing alternative approaches.



# Research method

## 2.1 Design and creation

Our study will be conducted using the design and creation research strategy. We have found this approach most suitable to us on the basis of Oates'[2] characteristics of the research method.

According to Oates[2], the focus of the "Design and Creation" research strategy is to develop new IT products. Oates calls these IT products artefacts. Further, Oates explains that:

*"For many research projects, especially in computing, the research involves analysing, designing and developing a computer-based product such as a website, group support system or computer animation."*[2]

The instance Oates explains, is the case for us. Oates continues to explain that how projects like this contribute to knowledge depends on the role that the IT system plays. The IT system can have one of three roles: the main focus of the research, a vehicle for something else, or a tangible end-product of a project where the focus is on the development process.

In our research project, we are analyzing, designing and developing a computer-based product. In our case, two prototypes are being developed. Our project explore and exhibit the possibility of digital technology: Responsive Web Design. Our IT system is not the main focus of the research, we are not researching the website of our university, or the website of New York Times. We are neither creating an IT system for the reason that we want to learn about the development process. The project has the role of being a vehicle for something

else: We are creating two implementations of an IT application to learn about the technical implementation. We are creating a responsive website in two different ways, one with the use of a RWD framework, and one without any framework. The evaluation of these two implementations is the contribution to knowledge. The technology and the design are the focus in this project.

A design and creation research is typically a problem solving approach. It uses an iterative process, consisting of five steps. The steps are not to be followed strictly and stepwise. They should instead form a more fluid, iterative cycle. Oates[2] describes the steps as follows:

- **Awareness** is the step where you recognize and define the problem.
- **Suggestion** is a creative step offering a tentative idea of how the problem might be addressed.
- **Development** is where the actual implementation happens.
- **Evaluation** is the step where we examine the developed artefact and looks for an assessment of its worth and deviations from expectations.
- **Conclusion** is where the results from the design process are consolidated, the knowledge gained is identified. We look at loose ends, unexpected results and things that can be subject for further research.

In our research, we have tried to follow these steps. See Figure 2.1 to see how our research process are mapped to the items of design and creation research.

The awareness appears through sections in the introduction chapter by explaining the background, motivation and asking research questions. The literature review and prestudy further recognizes the problem.

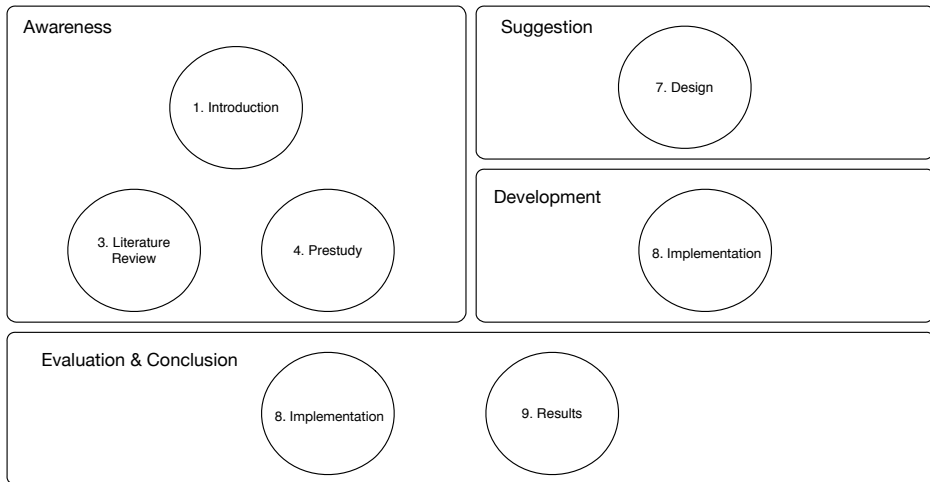
The suggestion appears through the design chapter, where we suggests an idea of how we can implement the prototypes.

The implementation chapter is where we present the implementation of the idea.

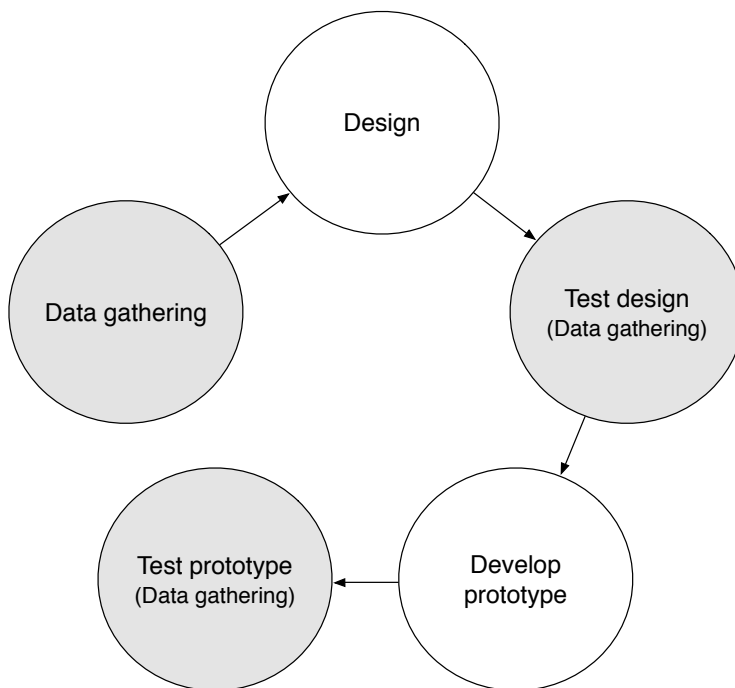
An evaluation of the developed prototypes is done in both the implementation chapter, as well as in the results chapter.

The conclusion where we draw lines from our design phase to our result, is found in the results chapter. Here we also present findings, and suggestions.

If we try to shorten our research process down to as few steps as possible, we can summarize it with two iterative steps: Data gathering and Design/Development. In Figure 2.2 we can see that we first do data gathering, then we design or implement. We have an initial data gathering, then we create our design for the prototypes. Further, we test our design on our target group, before we implement the actual prototypes. Lastly, we try to test our prototypes.



**Figure 2.1:** How the five iterative steps of the design and creation process are related to our chapters in the report

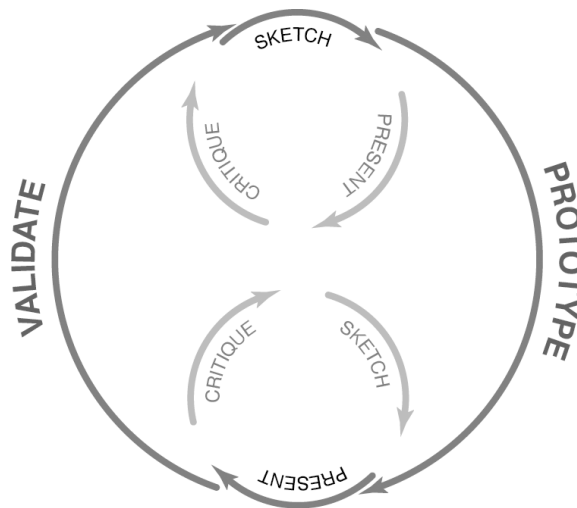


**Figure 2.2:** Our research process, the data gathering phases are highlighted

## 2.2 Systems development methodology

When developing our artefacts we need to work after a systems development methodology, either one that already exists, or we can create our own. Although, a process is merely a means to an end. We think it is a possible pitfall to follow a process and being tied too tightly to it. Therefore, we focus more on the end goal, than on the process itself. Still, we will be using a methodology as a foundation, while being flexible to adapt and change that process as needed.

We have been using the systems development methodology prototyping, which is illustrated in Figure 2.3.



**Figure 2.3:** The prototyping model with its phases[3]

We are firstly doing a prestudy, then creating design sketches. These first sketches are the first prototype, and illustrated in the Figure as "sketch". From this point, we try to get feedback on the already created prototype, by presenting the sketches for our users, and then let them evaluate and critique. We then move on to implement the more complex prototypes, which is illustrated on the right side of the Figure as "Prototype".

Finally, we present our prototypes for our users again to try gaining more valuable feedback, illustrated on the bottom of the figure as "Present" -> "Critique". After this critique, we will try to adapt to the feedback, before we lastly move on to the "Validate" part, illustrated on the left side of the figure.

Even though our development methodology has been prototyping, this does not mean that we will not be using techniques and tools from other methodologies.

For instance, we will be using an online "scrum board" to organize the different tasks. We are already familiar with Symphonical<sup>1</sup>, and will be using this to organize the development process both for the prototypes and the report.

We will also be using daily standup meeting from the scrum methodology <sup>2</sup>. This technique allows us to keep each other updated on what the other person has worked with, and is going to work with the following day. Giving us focus on the start of each day, by helping us gather our thoughts and deciding what to do is also an advantage we gain from using this technique. This way it is easier to stay up to date, and it will hopefully improve the collaboration amongst us. Additionally, we think it can urge us to work harder, since it will trigger us to have done work to show when starting the day using this technique.

## 2.3 User-Centered Design

Approaching the research focusing on the user and what their needs and requirements are, will be the way we want to go in this thesis. User-Centered Design [4] will then be the best method to follow when implementing this idea. The user centered design process is as follows, we first specify the context of use by identifying who will be using the website and under what conditions they will use it. Then we specify the requirements for the product to be successful. The next step is to create the design, and at last evaluate the design by letting the users see and test out the design. There are several analysis tools we can utilize when focusing on a user centered design, we will be constructing personas to encapsulate which user groups ntnu.no has. These personas will be the baseline for making scenarios to see how the different users can interact with the website.

## 2.4 Questionnaire

To find out more about the users habits when browsing ntnu.no, we decided to make a web based questionnaire. A questionnaire is a quantitative data generation method, and the main purpose is to get as many answers as possible. To maximize the result, we will distribute the questionnaire in multiple channels, to get input from multiple user groups. A questionnaire allows us to gain large samples in a short amount of time, it allows anonymity for the users, and gives us the opportunity to gather the results in an easy manner. On the negative side, users are unable to get clarification on the questions during

---

<sup>1</sup>symphonical.com - Project organizer

<sup>2</sup><http://www.scrumalliance.org/why-scrum/core-scrum-artifacts-activities>

the questionnaire, opposed to an interview where we can facilitate and help if needed.

The questionnaire will try to gain knowledge on the usage of the website on PC/Mac, and both tablet and mobile. We will try to find information on the context the users are in when accessing ntnu.no. Still, we also want to know what information they are looking for, depending on their platform of choice. The accessibility of the website is also an important issue we want to address, letting the users comment on how they are able to use the site as it is today. Our hope is to get answers we are able to utilize when making our own version of ntnu.no.

## 2.5 Interviews

To get a more broad knowledge base we will also conduct a few interviews to get some qualitative data we can analyze. There are three types of interview methods to choose from:

- Structured interview where you have a preset list of questions to ask the interviewee, and all interviewees gets the same set of questions.
- Semi-structured interviews gives the interviewer a chance to change the order of the questions as the interview goes on, you might also add additional questions.
- Unstructured interview, here you can start by introducing a main topic for the interview, and let the interviewee talk freely around that topic.

The semi-structured and unstructured interview methods are great for discovering purposes, while a structured interview is mainly for checking information[2]. The *semi structured interview* method gives us the standardized rules from a structured interview, but allows us to adapt depending on the interviewee and situations that comes up during the interviews. Because of this, we have chosen to use the *semi structured interview* method when conducting interviews for our thesis.

### 2.5.1 Group interview

A group interview is often held with a group of around 3-6 people, the advantages of this interview form is according to Oates[2] that they help generating consensus views, can generate more and varied responses, as the participants discuss topics between each other. There are also some disadvantages that might occur, people might be afraid to express their views in front of others. Some of the attendees might be more dominating than others, thus not giving

others the chance to answer. To counter these disadvantages we will try to compose a group of people from the same level, for example from the same class.

Since our team consists of two members, one can consume the role of the interview facilitator and the other can take notes during the interview. Additionally, we will try to record the session if possible.





# Chapter 3

## Literature Review

As a starting point for our master thesis, a literature review has been conducted. We have gathered material from many different sources, including books, but also articles and posts by web designers and developers. One one hand this literature review gave us an opportunity to find reasons of why responsive web design is important in modern websites.

On the other hand it also gave us the possibility to learn about the technical part of RWD. This includes information regarding different frameworks that can help the process of making a responsive web page.

Additionally, the literature review was used to discuss the different web domains that were alternatives for us. When deciding which website to redesign and create a prototype for, we considered several different possibilities. After the decision was made, we also had to gather information regarding the domain we ended up picking.

### 3.1 Responsive Web Design

Because Responsive Web Design is a very important term in this thesis, we will here define exactly what it is, and what a website needs to contain to be called responsive. The definition itself contains terms that needs further explanation. We will in this chapter go through them one by one.

Ethan Marcotte, the creator of RWD, defines it as a design composed of three distinct parts [5]:

1. A flexible grid.

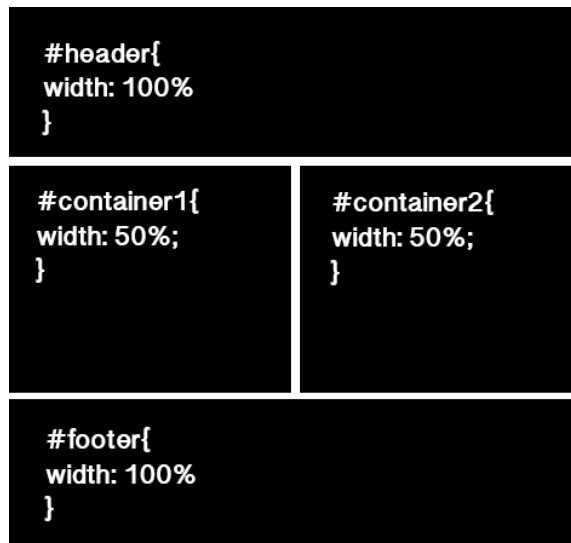
2. Flexible images. Or more specifically, media that work in a flexible context (whether fluid themselves, or perhaps controlled via overflow).
3. Media queries. The final layer of a responsive design, media queries optimize the design for different viewing contexts, and spot-fix bugs that occur at different resolution ranges.

### 3.1.1 Flexible Grid

A flexible grid, or a fluid grid is the main foundation when making a responsive design. When deciding the size of different elements on your web page, instead of using only pixel size, you should adapt a more relative approach: Using percentages of the total browser width. Fluid is defined by the Oxford dictionary as:

"A substance that has no fixed shape and yields easily to external pressure"[6].

This definition is a good fit for fluid grids as well. A very simple example of a design with a fluid grid is given in Figure 3.1. Regardless of the screen size, the header and footer will always fill the width of the browser 100%, the two containers will always take up 50%.



**Figure 3.1:** An example of a fluid grid. The containers will always keep its given width of the browser window given in percentage.

But what exactly is required to create and call your layout a flexible grid layout? Josh Byers has listed five components that together make up a flexible

grid[7]:

1. Flexible Type
2. Flexible Containers
3. Flexible Margins
4. Flexible Padding
5. Flexible Images

We will now explain each of these terms.

### **Flexible Type**

Instead of using pixels as measurement for font size one should use the special value em in a responsive web design. The em size unit is recommended by the W3C and is a relative measure[8]. 1em is equal to the current font size. The default text size in browsers is 16px. So the default size for 1em is 16px. This value is calculated with the formula:

$$Result = \frac{Target}{Context} \quad (3.1)$$

Where Target is the base value you are calculating from, Context is the wanted value of the new element in the design, and Result is the em value of the new element. So if your browsers default text size is 16, and you want your font to have a size of 14px. You can divide the two font sizes 14px/16px that results in a font size of 0.875em. This result makes your design flexible and scalable since you do not have to bother with pixels when scaling either up or down, the content will scale according to the relative size of the element.

### **Flexible Containers**

Flexible containers simply means that you need to specify your containers (div elements) in percentage, and not in pixels. As already illustrated in Figure3.1.

Byers explains that we can use the same procedure on the containers as on the type. In his example, he have a title bar which is 400px wide, and the width of his page is 960px wide. He then explains that you should divide the target (400px) on the context (960px), which will give you .41666667. To get it to percent, you of course multiply it with 100, and you have 41.666667. It is also suggested to keep all the decimals, to keep the design as accurate as possible.

The solution is to specify how wide your container should be relative to the parent container, using a percent value. If you do not have a premade layout and design with specified widths, we believe that you can just go ahead and specify everything in percent right from the start. Without first specifying it

in pixels, and converting it to percent value later. But obviously, you should understand that when setting a width percent on a div, it will fill its parent div with that many percent. We also believe that in the future, people will start to think in percentages instead of pixels, when thinking of widths.

### Flexible Margins

The same principle as illustrated with the containers and type, is also viable for margins. Here, our target will be the margin already set in pixels, and our context will be our container. Byers uses an example where he wants to set 25px margin, his container is 150px wide. This gives us  $25\text{px}/150\text{px}$ , which is 16.666667%.

### Flexible Padding

For the padding, we still follow the same principle. Keep in mind how the box model looks like, the padding is on the inside the border. The context is therefore the width of our container, and our target is the padding we want to set. So say that we have a box which is 300px wide, and wants 10px padding on both sides of this box, our calculation would be:  $10\text{px}/300\text{px}=3.33333\%$ .

## 3.1.2 Flexible Images

The key to get images to be flexible in the same way text is flexible, is to use the max-width property, as shown in Listing 3.1.

**Listing 3.1:** To make an image responsive

```
1 | img
2 | {
3 |     max-width: 100%;
4 | }
```

## 3.1.3 Media Queries

Media queries were first suggested by Håkon Wium Lie's CSS proposal in 1994 [9], the first draft came later in 2001. CSS2 had media style sheets that enabled you to write separate CSS for print and screen, but not media queries that enabled you to write CSS for different screen resolutions[10]. This did not become a W3C recommended standard before as late as June 2012 [11]. Media queries is a module which was introduced in the CSS3 style sheet

language. In responsive web design it is used to target different screen widths and write customized styling. This enables us to handle all the different screen capabilities of different devices. Without media queries, you would be unable to change styles at specific device capabilities[12]. In Listing 3.2, devices with a size of 480 pixels or less is targeted. The CSS rules written in this part of the code will only apply for these devices

**Listing 3.2:** A media query

```
1 @media screen and (max-device-width: 480px)
2 {
3     /* some css here */
4 }
```

## 3.2 Responsive frameworks

Within software development, code reuse is a common strategy to shorten the time to market. The time to market is then reduced by shortening the development process. This strategy also applies when writing responsive websites. Reinventing the wheel every time is not easy, and perhaps not necessary. Because of this, people have created a number of existing frameworks for making responsive web sites. The websites made with frameworks might have features that are fairly similar since every framework is trying to fix the same problem. Below follows a short summary of five of the most popular responsive frameworks today.

### 3.2.1 Bootstrap 3.0

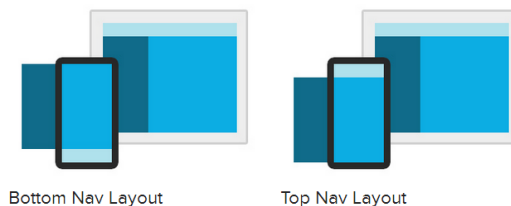
Bootstrap [13] is one of the most popular front end frameworks and open source projects. Actually, it is the most popular project on the source platform GitHub[14]. There it is hard to find unique features that Bootstrap only offer, and the popularity must take some blame. They offer a large set of reusable components, built to offer iconography, dropdowns, navigation, alerts, popovers and more, but all of these are in some form mimicked by other frameworks. On the other hand you can say that Bootstraps most important feature is their popularity, giving developers easy access to support, since the community supporting Bootstrap is so massive. From version 2.0 Bootstrap has been responsive, by offering a grid layout. The newly made version 3.0 is based on the mobile first principle which we explain in Chapter 3.3. The difference between version two being responsive, and three being mobile first, is that version two offered optional mobile friendly styles for key aspects of the framework. Version three

is mobile friendly from the start, mobile first styles can be found throughout the entire library, instead of in separate files.

Bootstrap is a quite extensive framework, it offers numerous features. It is based on HTML, CSS and JavaScript and gives you styling of elements and a responsive grid. Nevertheless, it gives you the opportunity to download only what you need. It lets you set custom colours and styles on the different Bootstrap elements. It is built using CSS LESS variables <sup>1</sup>, so you can actually edit the value of exactly what you want, such as the pixel width of the breakpoints. The custom file is then compiled and ready for download.

### 3.2.2 Foundation

Foundation [15] is also a mobile first framework, it gives you a 12 column flexible grid that can be stretched to any sized screen. Foundation also has a huge set of different predefined styles that can help you make a quick prototype of your site in no time. Foundation has several features that we found was unique to this framework. Interchange is a method using media queries to change content according to the screen size, for instance changing what picture to display. This will make large demanding pictures only to load on devices that can handle it. Another interesting feature is the possibility to use off canvas layouts like in Figure 3.2. Off canvas layouts has the menu positioned outside the screen, when you click the menu element, the menu slides in.



**Figure 3.2:** Example of two off canvas layouts

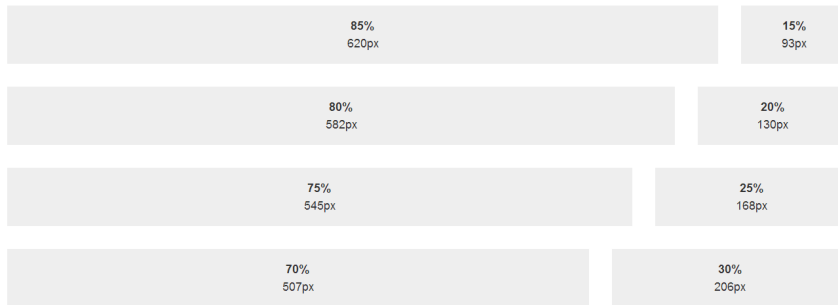
### 3.2.3 Unsemantic

Unsemantic [16] is also a grid based system, but instead of using number of columns like for example Bootstrap, it uses percentages, `<div class="grid-25">` will give you a 25 percent wide column. It uses SASS which stands for Syntactically Awesome StyleSheets which is an CSS extension that has more

---

<sup>1</sup><http://lesscss.org/>

power than normal CSS with features like variables, nesting of selectors, selector inheritance, and mixins(reuse of CSS code). A part of the grid that Unsemantic uses is shown below in Figure 3.3. Unsemantic does not have any pre defined styles on GUI elements, making the user create their own personally styled elements.



**Figure 3.3:** Some of the grid elements and their respective sizes in percent and pixels

### 3.2.4 Skeleton

Skeleton [17] is probably the smallest of these five frameworks. It is a small collection of CSS files that is intended to help you build a site to look good on any sized screen. It gives you the most basic styles and lets you adapt it to your own style. Skeleton is a very good responsive framework if you want to give your website your own style, the framework itself is just supposed to be a basic foundation underneath your own code, Skeleton illustrates this as in Figure 3.4. Skeleton does not have styling for most of the different graphical elements, although there is basic styling for buttons and forms implemented.

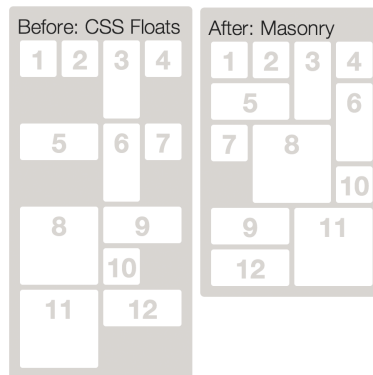


**Figure 3.4:** Your code is the important part

### 3.2.5 Masonry

*"Mason is a person skilled in cutting, dressing, and laying stone in buildings."*[6]

The Masonry framework is all about the same things as a real word masonry: Fitting elements together without making any holes. The framework places the elements for you in a cascading grid. It tries to prevent empty spaces in the grid by placing the elements where they fit best, based on the available space, as illustrated in Figure 3.5. It differs greatly from other responsive frameworks, which in most cases has a grid built up by rows.



**Figure 3.5:** A grid before and after the use of masonry, before using Masonry the elements only uses the standard float property of CSS

Masonry [18] is a dynamic layout plugin for jQuery. In addition to building this grid for you, Masonry animates the movement of the elements when resizing the window. Pinterest<sup>2</sup> is a website that fully shows the possibilities of the Masonry framework.

### 3.2.6 Framework summary

A brief summary of the frameworks can be seen in 3.1. It is easy to notice that Bootstrap is superior in popularity. It does also have the responsive grid, as well as the possibility to customize your stylesheet. It does however not contain any features that are unique in any way. Using a framework, or application that is very popular can have its advantages. When searching for answers on an issue on the Internet, your chances to get results increases with a large user mass.

---

<sup>2</sup><https://www.pinterest.com/>



	Favourites <sup>3</sup>	Responsive grid	Custom styling	Features <sup>4</sup>
Bootstrap	61 483	✓	✓	
Foundation	14 653	✓	✓	Interchange, Off-canvas
Masonry	5 171	✓	✗	Cascading grid layout
Skeleton	4 045	✓	✗	
Unsemantic	612	✓	✗	

**Table 3.1:** A feature summary of the responsive frameworks discussed in this chapter.

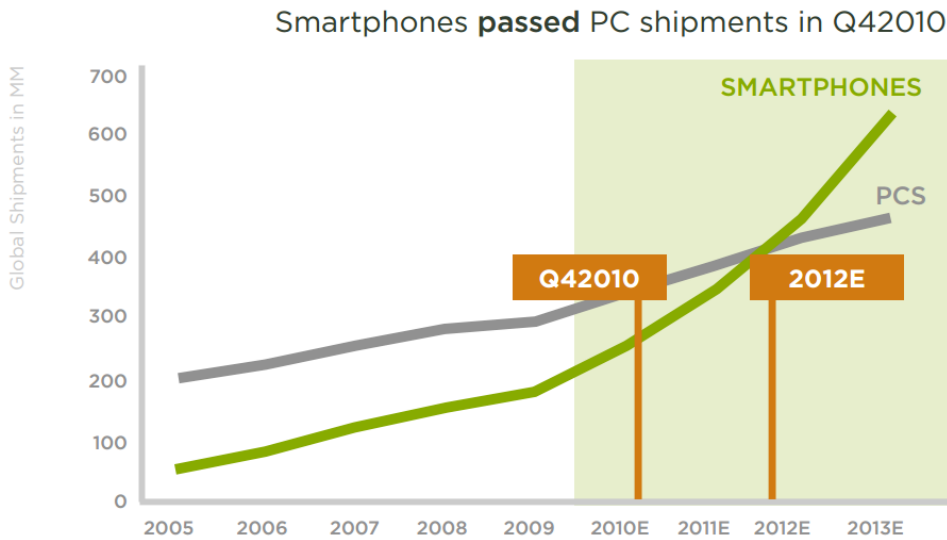
<sup>3</sup> Number of stars on <http://github.com>, accessed 28. Nov 2013 <sup>4</sup> Features that are considered unique to this framework. The features are explained under the respective frameworks' subsection.

### 3.3 Mobile first

Mobile first is a term constructed by the author Luke Wroblewski [19]. The reason to design for mobile first is to get ready for the explosive growth in mobile Internet usage, resulting in more people using a mobile phone to access the Internet than PC users. The author uses three key elements to explain the idea behind mobile first.

- Growth
- Constraints
- Capabilities

Growth is the first element, smartphones outgrew the PC and laptop market already in 2010 and is continuously rising, as illustrated in Figure 3.6. On one hand, massive growth is demanding better and more efficient ways to present content on smartphones, which leads to the idea about mobile first. On the other hand, smartphones also have some constraints, which is the second key element. Smartphones does not have the same screen capabilities as desktop or laptop computers. They can also be limited by network speeds, and it depends which context the smartphone is used in(since its often used on the move). The contextual use can also be a positive capability, and not only a limitation, which is the third element the author suggests. The smartphone open up new innovative ways in how this media can be used. By using smartphone features as an accelerometer, location detection and device orientation you get a big specter of possibilities when designing your web page.



**Figure 3.6:** Smartphone Vs Desktop shipment [20]

## 3.4 Inspiration

We looked at Norway’s top 100 visited websites[21], to see if any of them could be ideal for us to evaluate and test responsive web design techniques on. While browsing, we noticed that a minority of the websites were responsive.

Additionally, we have been looking at numerous different websites that already are responsive, to get inspiration from them, and to fully understand the concept.

Here we will present some of the ideas we have considered conducting our research on. Further, we will present a few of RWD websites we have been looking at for inspiration.

### 3.4.1 Ideas

One of the first ideas we had was to implement and test out responsive design on some sort of web store. These often lack support for smaller screens, and ordering can be troublesome on small screens. Newspapers is also populating the top 100 visited list, and most of them is not responsive. They also use customized versions for both mobile and tablet devices, which we think is unnecessary work, since a responsive page will do the same job. On the

other hand, newspapers does not have their content changing much between the screen sizes. Thus not being very attractive to our thesis.

As students, we know many with different it-consultant firms from job interviews, and also when the firms visit NTNU to present themselves to the students. When browsing their sites, for example when delivering a job application, we saw that many of the firms did not have a responsive site. A collaboration with one of those firms, testing responsive web design on their website was one of our top ideas. Here we would have access to users, but the issue was getting access to statistics from the website. We did not see a possibility that the firms would let outsiders gain access to their systems, and thus would inhibit us in the development process. We decided to go with an idea that we felt more familiar with, the idea is to implement responsive web design on a website for school that offers higher education. We therefore want to inspect and evaluate different websites from this domain.

### 3.4.2 Higher education

Eric Runyon[22] splits responsive sites into three categories.

- Fully responsive sites
- Hybrid sites with responsive behaviour at some sizes
- Adaptive sites which is not quite responsive, usually lacking some features like a flexible grid

Runyon has made a list of current HigherEd homepages which falls within either of these three categories, they are labeled accordingly[22]. These pages has been inspirational for us. Although, since NTNU is based in Norway, it is highly relevant for us to be looking at the other Norwegian higher education websites. We will be inspecting three websites: The University of Bergen, The University of Oslo, and Bedriftsøkonomiske Institutt (BI). These are the three largest universities in Norway.

#### **The University of Oslo**

The website for the university of Oslo (uio.no) is illustrated in Figure 3.8. The site is responsive, although not by definition: It does not have a fluid grid, but it does have responsive images and media queries. When the browser width is over 585px wide, we can see that we get a horizontal scrollbar until we reach 1000px. Below 585px, the site turns into a mobile/tablet version. By definition, UiO's website is therefore adaptive as stated by Runyon[22].

By inspecting the UiO website, we can see at the top of their HTML file that they include an extra CSS file which is being used at certain screen sizes. This

stylesheet inclusion is shown in Listing 3.3. If we convert 15.5cm to pixels, we end up with 585.826771654px, which corresponds to the pixel width we noticed a break point for the tablet/mobile version.

**Listing 3.3:** UiO's "adaptive" feature

```
1 <link type="text/css" rel="stylesheet" media="only screen and (
    max-width: 15.5cm) and (orientation : portrait), only screen and
    (max-width: 17.5cm) and (orientation : landscape)" href="/vrtx/
    decorating/resources/dist/style/responsive.css">
```

The most visible element on UiO's site is the find study program, and find courses box. Before NTNU updated their website during spring 2013, this was functionality that was very hard to find. UiO's way of displaying this functionality therefore gave us ideas on how we could implement a similar solution in our design.

### The University of Bergen

The university of Bergen's website (uib.no) is illustrated in Figure 3.9. Their website is considered fully responsive, even though we noticed one bug: Their "carousel" with pictures does not seem to be fully fluid when resizing the browser window. The picture overflow the browser window, thus adding a horizontal scrollbar, however it resizes the way it should when refreshing the page. So if it works with a refresh, it will look good on handheld units. Their responsive tablet and mobile version include the "hamburger icon", seen in Figure 3.7, which has seemed to become a standard within RWD. The hamburger icon is used as a menu button, hiding the full menu, and instead giving the user multiple new options if clicked.



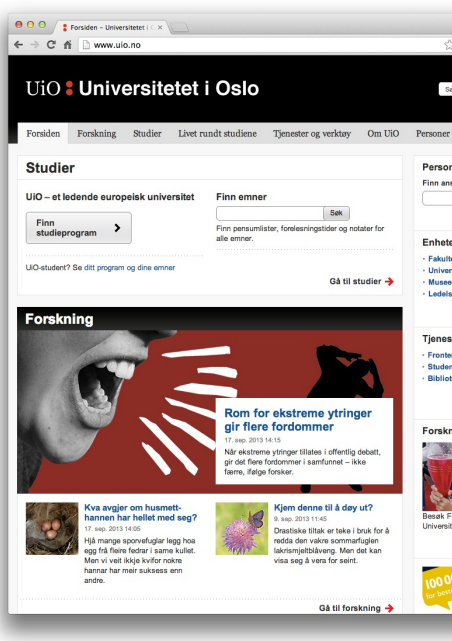
**Figure 3.7:** Hamburger icon

By inspecting the site, we can see that UiB is using Drupal as their content management system (CMS). They have a lot of JS files for their carousel plugin, which is a jQuery cycle plugin<sup>5</sup>.

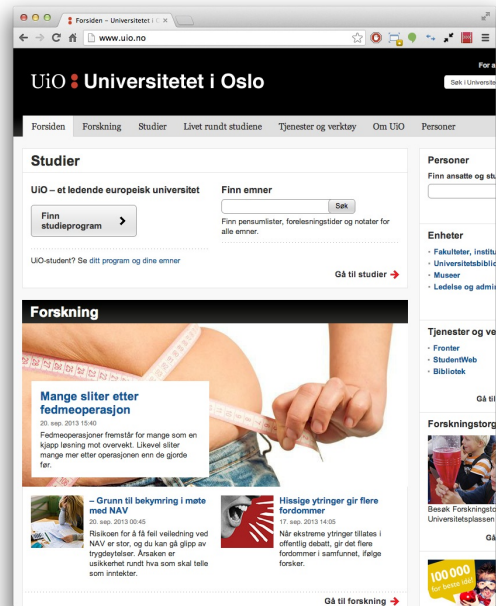
All their CSS files were minified, hence rendering them very hard to read. Finding information regarding their responsive features was not an easy task, the only clues we found was that they used four CSS files that was included for media="all" which is all browser widths.

---

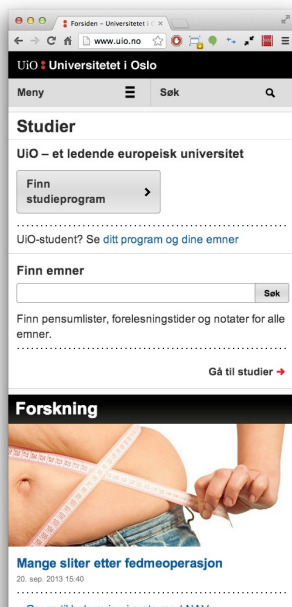
<sup>5</sup><http://jquery.malsup.com/cycle/>



(a) 992px wide



(b) 825px wide

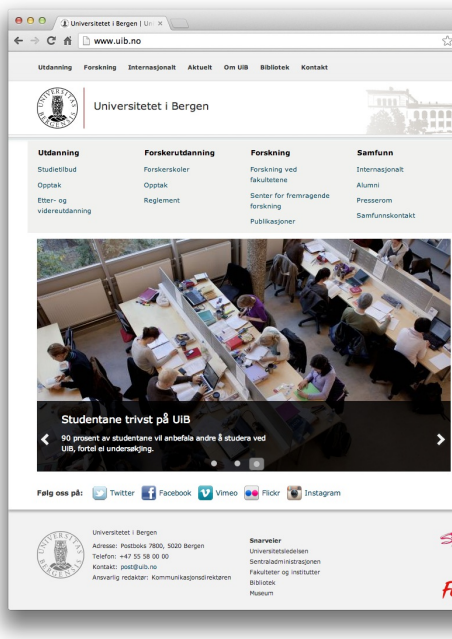


(c) 491px wide

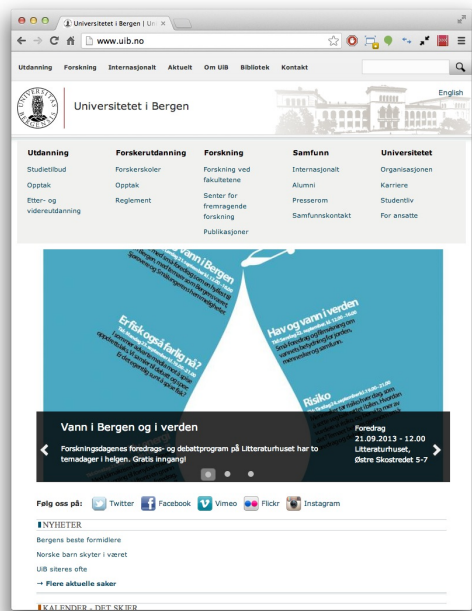


(d) 400px wide

Figure 3.8: UiO's adaptive website (uio.no accessed 22.09.2013)



(a) 988px wide



(b) 788px wide



(c) 547px wide



(d) 400px wide

Figure 3.9: UiB's responsive web site (uib.no accessed 22.09.2013)

## Bedriftsøkonomiske Institutt (BI)

Bedriftsøkonomiske Institutt' (BI) website is a fully responsive site, consider Figure 3.10. They even exclude content as the screen gets smaller. As you see from the 1758px version to the 1127px version they exclude content that is not important for the user; the decorative illustrations on both sides of the main content. These illustrations actually gets reduced in size gradually, starting with four persons on each side of the browser window. When the screen size is too small for them to fit, they are removed at 1127px wide resolution. At 931px, they remove the padding on the side of the box showing their studies.

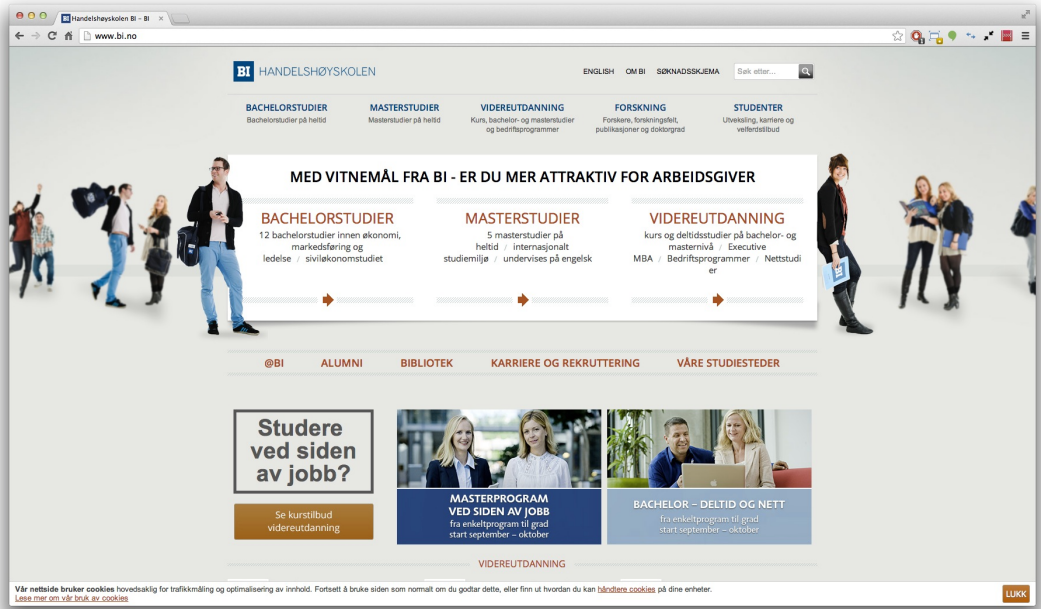
From 1127 to 931 we can see one image disappearing. The image does not just do a line shift like in some web sites, it simply removes it. This is probably done by having a box with `overflow: hidden`. Finally, the website turns mobile and we get the menu listed vertically instead of horizontally. A notable difference between BI's mobile version and the two Norwegian university mobile versions is that BI have chosen not to use the hamburger icon. This gives them less space to display the rest of the content, but gives more focus and visibility for the elements in the menu.

By inspecting the sites CSS file, we can see that it has over 4 000 lines of code. We can also see that it is developed by Epinova, we can see that they seem to be focused on CMS solutions after a quick Google search. Furthermore, it seems that Epinova has developed a custom grid system for the BI site. Of course, they might use this on other sites they produce as well. This grid style is shown in Listing 3.4. Additionally, we can find media queries at the bottom of their style sheet. Our guess is that they have not used a framework to achieve responsive features on their site.

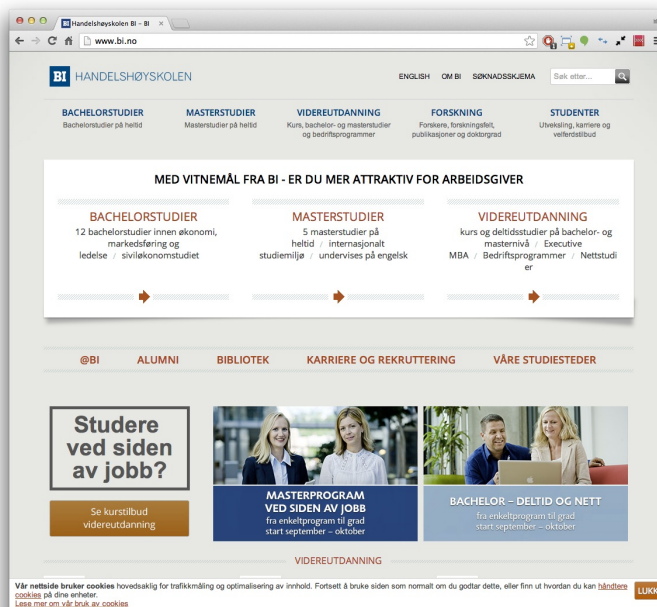
**Listing 3.4:** BI's grid styles

```
1 | .width1 {width: 8.33%;}
2 | .width2 {width: 16.66%;}
3 | .width3 {width: 24.99%;}
4 | .width4 {width: 33.32%;}
5 | .width5 {width: 41.65%;}
6 | .width6 {width: 49.98%;}
7 | .width7 {width: 58.31%;}
8 | .width8 {width: 66.64%;}
9 | .width9 {width: 74.97%;}
10 | .width10 {width: 83.3%;}
11 | .width11 {width: 91.63%;}
12 | .width12 {width: 99.96%;}
```

## Chapter 3. Literature Review

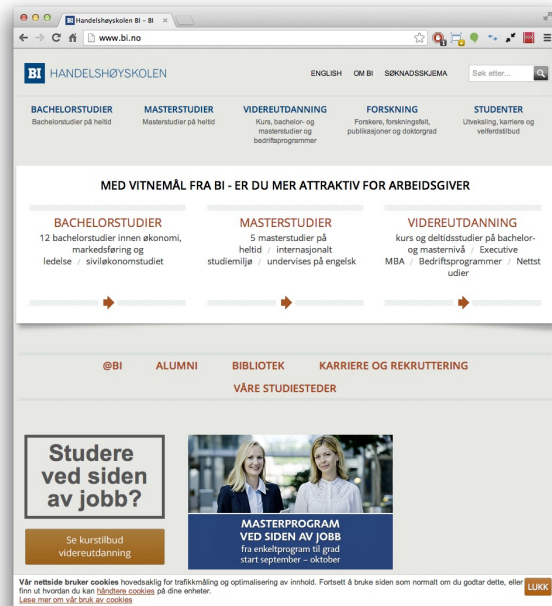


(a) 1758px

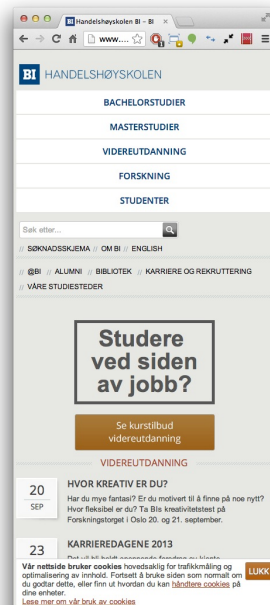


(b) 1127px





(c) 931px



(d) 448px

Figure 3.10: BI's responsive web site accessed 22.09.2013



# Prestudy

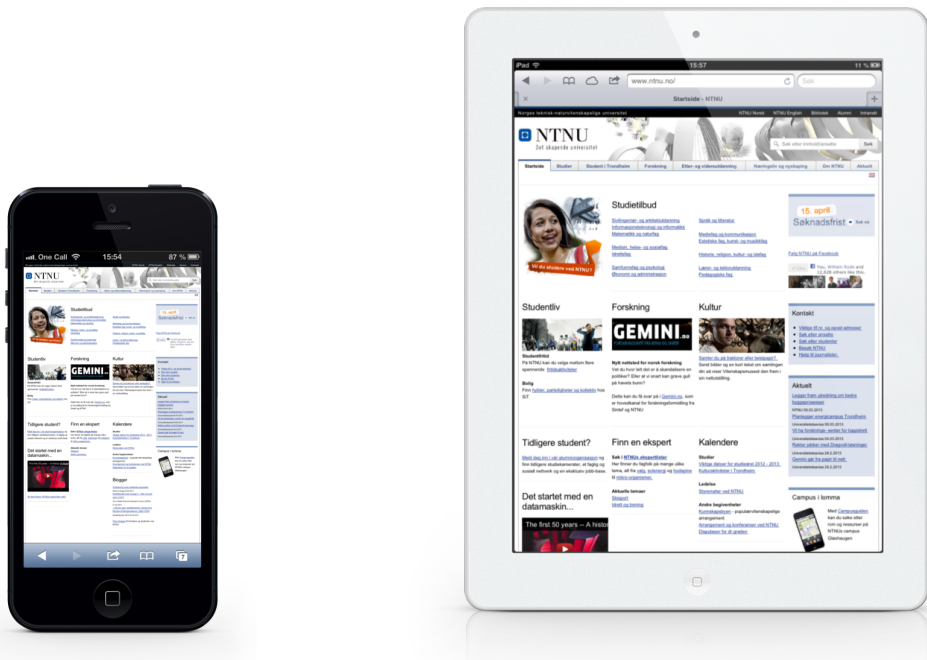
## 4.1 Introduction

To achieve results for our research questions, we need to work with a concrete webpage. Giving general guidelines for prioritising content would be quite troublesome, and perhaps unrewarding since every type of website has own preferences on which content is important or not. Since we are students at the Norwegian University of Science and Technology (NTNU), the university's webpage is quite accessible in terms of statistics and quantitative data. Additionally, this is a webpage we have been using frequently ourselves, and we have easy access to other students and professors who use the webpage. Because of this, we believe we have a good chance to gather a decent amount of qualitative data as well. The NTNU website is neither responsive nor does it have a mobile or tablet version, which are important factors for us to consider when using this website as the test subject in our thesis. This is shown in Figure 4.1.

The NTNU webpage is a quite large webpage, with several different user groups that each have different sections specified for them. It is important that we manage to identify these user groups, and identify what content these user groups use the most. Of the 80-100 tasks that someone might be able to do on a website or intranet, four to six tasks get between 20 and 35 percent of the votes [23]. We need to identify these top task identifications, to be able to structure and prioritise the content of the webpage. Additionally, this needs to be done for the different devices, since the top tasks will change accordingly to the context transmutation.

In this chapter we will explore the problem domain. We will set up a list of

assumed user groups, and create personas to get an overview of the NTNU website users and their specific needs.



(a) Viewed on an iPhone

(b) Viewed on an iPad



(c) Viewed on a computer

Figure 4.1: Ntnu.no viewed on different devices

## 4.2 Personas

Before we can start analyzing the NTNU website, we have to get a view on which people are using the site. By looking at the different elements of [ntnu.no](http://ntnu.no), we envisage these six user groups:

- Current students
- Future students
- Former students
- Parents of future students
- Professors and other employees of NTNU
- Media (Journalists, and others seeking contact with specialists of a specific field)

To better understand the users goals and limitations, limitations meaning what types of equipment they use, and in what context they are when accessing the website. We have decided to make up fictional characters to represent the different user types within the website. We believe that this helps us to create a more user-centered design.

### Pål

- **Professor, SVT, NTNU**
- Age: 47 years
- Male, married
- Residence: Heimdal
- Likes to listen to classic music and to read books
- Likes to be active and is a yearly contesteer in "fjellseterløpet".
- Rarely uses his PC at home, but uses it daily at work.
- Has a simple and cheap Android phone with a small screen. Uses the web browser on the phone exceptionally rarely.



Image ID: 1363173 sxc.hu

### Silje

- **Student, Computer Science, NTNU**
- Age: 22 years
- Female, single
- Residence: Rosenborg, Trondheim
- Hobbies include ballet and orienteering
- Enjoys reading blogs about fashion and celebrities
- Have a windows based laptop which is frequently used both in school and at home
- Have a Galaxy SIII which is being used a lot when on the move, and even in bed before going to sleep. She also has an iPad 2.



Image ID: 727713 sxc.hu

### Jenny

- **Possible future student of NTNU**
- Age: 18 years
- Female, single
- Residence: Fyllingsdalen, Bergen
- Hobbies include playing handball and going to the mall with her friends
- Enjoys watching her favourite TV series; Gossip girl and True Blood
- Is considering applying for Energy and Environment or the chemistry study on NTNU
- Have a macbook air which is used both at home and on school
- Have a Iphone 4S which is being used a lot, and also a Ipad 2 which is used mostly at home



Image ID: 917483 sxc.hu

**Ørjan**

- **Journalist**
- Age: 32 years
- Residence: Grünerløkka, Oslo
- Hobbies include traveling to urban locations, taking photographs and playing squash
- Focuses on environmental journalism
- Have a 15" Macbook retina which is frequently used on work, while traveling and at home
- He has a Iphone 5, and also a Ipad 4 which is used especially a lot while traveling



Image ID: 1287009 sxc.hu

## 4.3 Scenarios

The scenarios will try to encapsulate different needs that users of ntnu.no has. The answer to each scenario reflect a perfect solution on how the website should perform to best accommodate the users needs, and will be a good guideline on how we should develop the prototype.

### Scenario 1 - The student

Silje the student has a class in 15 minutes, but she is fairly new at school and have no idea where the H1 auditorium is. She decides to take up her smartphone and visit ntnu.no. On the front page there is a search field with the name "finn rom" where she enters "H1", the site gives Silje suggestions while she type, and she clicks the H1 auditorium suggestion, which then directs her to the map with the correct building and floor.

### Scenario 2 - The professor

Pål the professor is supervising two students writing their master thesis, the students are wondering if they can fit in certain courses in their degree. Pål then uses his macbook to visit ntnu.no, he immediately sees the find course search field on the top of the page and starts typing in the course name. The website immediately gives back suggestions while he is typing, and he does

not have to finish writing the full name, since he can just click the correctly suggested course name which sends him directly to the course page.

### **Scenario 3 - The student**

Silje is relaxing on the couch a Sunday afternoon while wondering when she has to be up the next morning. She logs in to ntnu.no, quickly discovers her lecture plan on the top of the page, sees the start time of her first lecture and logs out again.

### **Scenario 4 - The student**

Silje is studying on school, but have difficulties logging in to It's Learning, the student portal which gives her information about her courses. She logs in to ntnu.no to see if there is any news regarding this. She quickly sees under the RSS feed that It's Learning are having server issues, and that it will be up and running within the next hour. She then logs out again.

### **Scenario 5 - Possible future student**

Jenny is wondering where she wants to study. She is browsing through the homepage of different universities on her macbook air. She enters ntnu.no and quickly sees nice pictures of the main building on NTNU Gløshaugen. She gets a good first impression, but needs to check if there are any studies that she is interested in there. She quickly discovers a large section of studies, and sees two of her main interests there: "Natural Sciences" and "Information technology".

### **Scenario 6 - Possible future student**

Jenny has been to NTNU's website earlier, and wants to revisit it for more information. She quickly localize the links for the studies she is interested in, since she easily remembers the icons they are associated with. She visit both the "Natural Sciences" and "Information technology" links, to read more about the studies. She then goes back to the front page to see if there is anything more interesting, she discovers a section about the campus life, cycles through the links there and gets up to speed with how the daily life of a NTNU student is. She then logs out.



### **Scenario 7 - The journalist**

Ørjan is working on an article about global warming and needs to find someone who is an expert on the ozone layer. Since he currently is on the airport express train, he pulls his iPhone 5 up from his pocket and heads to ntnu.no. He finds a link marked "help for journalists" on the front page which he quickly clicks on. On the next page, he can easily browse through different fields of experts or search. Ørjan searches for ozone layer and finds a PhD candidate that currently is writing his thesis about a hole in the antarctic ozone layer. Ørjan clicks his phone number and his iPhone instantly phones up the PhD candidate.

## **4.4 Dividing the site**

From our analyze of content and users on the website, we have discovered a separation in wanted content. The users can be divided in two user groups where each group shares similar content. This eases the possibility to serve the users the content they want, and not flood them with excessive information. These user groups are as follows:

### **User group 1**

- Future students
- Former students
- Parents of future students
- Media (Journalists, and others seeking contact with specialists of a specific field)

### **User group 2**

- Current students
- Professors and other employees of NTNU

It seems that adding all this content on one site creates a unnecessary cluttered website. In this case, it seem to go on the compromise of user group 2. Of course, the front page of the university needs to focus on the external user groups, which does not have a need for using the site as the internal user groups do. The university needs to sell itself to the future students.

On the basis of this, we have decided to create two versions of the ntnu.no site, where user group 1 will have the same url which is being used today; ntnu.no, and user group 2 will need to have a new url. The new url for the second user group will need to be short and easy to remember, but still linked to the NTNU domain. Examples of this new url would be s.ntnu.no, ntnus.no, ntnu.no/stud, stud.ntnu.no etc. This concept is presented in Figure 4.2.

By this division, one would fulfill the needs of both user groups to a better extent.



**Figure 4.2:** The two user groups will have their own website with two different addresses

## 4.5 Content

The NTNU website is a frequently visited site, in 2012 it had 5 839 674 visits<sup>1</sup>. When redesigning the website, it is important to determine which parts of the website that is mostly used. In Figure 4.3, we can see the most visited pages on ntnu.no in 2012.

As we can see, the most used function on the NTNU website is the search function. 30.5% of all clicks on the NTNU website in 2012 was on the search button. What do we get from this? Many people search for a lot of things, this could be either documents, articles, courses, or anything really. This lead us to believe that people don not easily find what they want from the front page, so they search instead. It can also be an indication that the users prefer to search over browsing. This again can be a sign that many of the users that visits ntnu.no knows exactly what they want to find. When searching, you need to type in what you want to find, as opposed to when browsing.

Secondly, we have the studies with 13.6%. Under this page there is a Listing of all the different areas of studies, which you also can find on the front page.

---

<sup>1</sup>A visit is a series of events which starts when a visitor reads the first page and ends when the visitor becomes idle for more than 30 minutes(industry standard)

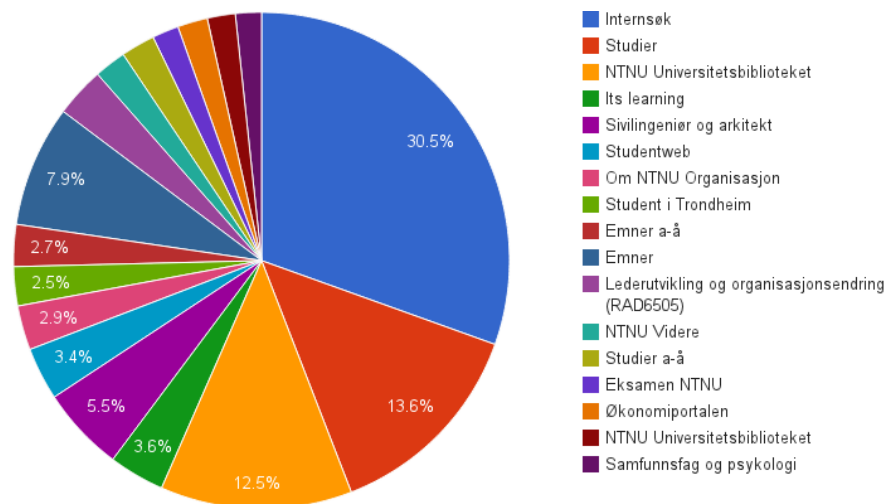
Additionally there is a left hand menu containing links, one of them being to courses. When going further down on our list of most clicked pages we can see that "Sivilingeniør og arkitekt" is quite popular, which is an area of study. We also find courses ("Emner a-å"). Both of these being subpages of the studies page.

We can also see that there are quite a few links to external sites and tools. The NTNU university library link is the third most used, It's Learning the fourth, and "studentweb" as the sixth most popular link.

But this in itself is not enough in our case, we need to try to figure out which use cases are the most used in each of our scenarios:

1. On a computer
2. On a tablet
3. On a smartphone

Then we can try to customize the content on the different screen sizes more accurately according to the users needs. Although, there might not be big or any differences for this website.

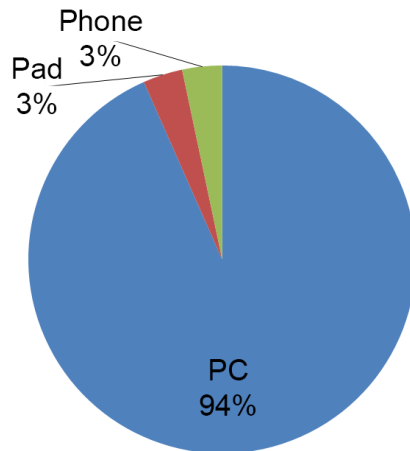


**Figure 4.3:** The most viewed pages on ntnu.no in 2012 [24]

## 4.6 Resolutions

Which devices is accessing the website today? This is an essential question and is partly answered in Figure 4.4. Although the percentage of mobile and

tablet users are relatively small, you need to take into account how the website is designed today. Many users might be avoiding the website on purpose when on their smartphones and tablets.



**Figure 4.4:** Percentage of ntnu.no accessed by phone, tablet and PC in 2012. This data is based on the breadth of screen resolutions. Assuming 0-480px is phone, 480-768px is tablet and all above 768px is a PC.

## 4.7 NTNU renewed their website design

On the 6th of May 2013 the technical group released a new version of ntnu.no with some new design elements. The webpage improved, and we can see some similarities to our redesign. Although, the webpage is still neither responsive or mobile friendly. The new design is illustrated in Figure 4.5.

NTNU has been wanting to go through the design and the structure of NTNU's external website, but they haven't had the capacity to do something about it until now. NTNU explains that the main problem has been that the website was overcrowded and too little focused. As it has seemed to us, NTNU admits that the main target group is potential applicants for NTNU. The redesign does also try to point this out better than before. Which for us students does not exactly improve the website.

This small design update of the front page of NTNU's website is a project that they have been working on for three months. The structure of the whole NTNU website was decided back in 2004, and since then the design and content of central parts have been changed now and then. To this date (19.09.13) the design update does only include the front page and one sub page (the studies

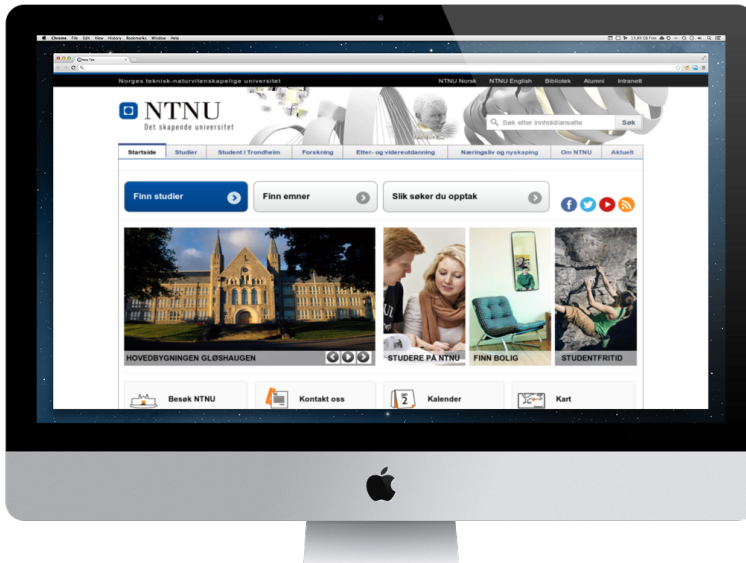
## 4.7 NTNU renewed their website design



(a) Viewed on an iPhone



(b) Viewed on an iPad



(c) Viewed on a computer

**Figure 4.5:** Updated ntnu.no design viewed on different devices

page). They do also have plans on redesigning the front page of the English version of the site (ntnu.edu) within the near future. Their next big planned change on NTNU's external web is a full review of the research pages under ntnu.no/forskning, which is the research pages of NTNU. This was said to begin at the earliest, the fall of 2013. What have been taking up NTNU's time? NTNU have been working on a new publishing system (Liferay) and a new intranet (Innsida 2.0).

RWD? NTNU does absolutely have RWD on their wish list for further development of both the external web and the intranet (Innsida 2.0). So whats holding them back? They are using a CMS solution called Liferay, and have to prioritise very hard what they use their time and money on. This of course means that they are bound to the possibilities of Liferay, but a quick Google search for "liferay responsive" reveals that there are many options available, such as premade responsive themes and even responsive theme builders<sup>2</sup>.

It have been a discussion within NTNU that they want to focus more on the "mobile first" principle for information regarding the students. This was raised as a question from the student representative Michael Johansen in a board meeting on the 24th of April 2013. His answer from the top administrative leader Ida Munkeby was that an ongoing strategy for mobile services at NTNU is currently being worked on.

NTNU says that they are fully aware of the potential for improvement in many areas. In the future they plan that the intranet (Innsida) will be the students portal, and will cover everything they would need. However, NTNU claims that their site is not that bad. They actually got 5 out of 6 stars in DIFI's (Agency for Public Management and eGovernment) quality test [25]. The full version of the interview with NTNU-info can be found in appendix A.

---

<sup>2</sup><http://www.liferay.com/web/nathan.cavanaugh/blog/-/blogs/liferay-com-mobile-sites-and-responsive-layouts>

# Chapter 5

## Process evaluation

This chapter will give an overview on how this research was conducted, starting from the literature review, all the way to a finished and tested prototype. We will try to evaluate the different steps to look at what worked, and what did not.

### 5.1 Literature review

After we agreed on basing our master thesis on the topic responsive web design, we started looking for relevant literature. There were several end goals of this literature review:

- Learn about RWD
- See if there were any research of doing things with RWD beyond the ordinary
- Find a suiting website to do RWD on

We wanted learn both the theory and technical points around RWD. For this, we took a starting point in Ethan Marcotte's book, and gathered deeper and more information from elsewhere, when there was a need. We were also curious to if there was done something extraordinary regarding RWD. Especially if there had been done any research on doing a content prioritization which was different than just scaling down the elements when decreasing screen width. We tried to look for published and well known articles in research databases like IEEE[26] and Springer Link[27], but quickly found out that it was not much to gain from these. Since RWD is a fairly new subject, there are not a lot of published articles on the it.

Lastly, it was important to find interesting areas we could connect and implement a responsive web design on. We looked in different types of domains to see how the websites were created, and gathered some thoughts on how RWD could be used to improve the user satisfaction and usability. After we had decided for a website, we looked at other websites of the same type gain inspiration.

The literature we gathered was mainly from online articles made by web designers and professors in the design field. But, also from books about RWD and front end programming.

## 5.2 Prestudy

After choosing to implement RWD on the NTNU website, we conducted a prestudy to get an overview of the website. We got a large amount of data material from the NTNU IT staff which was analyzed in this phase. We managed to get valuable information from this analysis. We could see which screen resolutions and Operating Systems that had accessed the site, which was very relevant when creating a RWD website. Furthermore we could see which parts of the site that had the most traffic, this proved to be very relevant when redesigning the website.

Information on the usage of ntnu.no on different device types, and Figures on what parts of the site which was most used was valuable information we gathered. The prestudy also consisted of finding the user groups and their different needs.

An important part of this study was to gather information from the users. Since we wanted to accumulate as much data as possible in a short time frame, we decided creating a questionnaire would be a viable solution. We tried using all possible channels to distribute the questionnaire. We shared it among acquaintances via instant messaging and different social media platforms. Also, we managed to share a link to the questionnaire on Innsida, NTNU's student portal. We received roughly two hundred answers that gave us a decent amount of data to continue working with.

## 5.3 Initial design

The initial design was constructed using mainly results from the questionnaire. We also use our own experience as students, considering we have used this page extensively during our five years at NTNU. The initial design phase will be



giving content to discuss in the interviews, by letting the interviewees compare the current design versus the new design.

## 5.4 First interviews

The first interviews was conducted with two different types of students, the first group consisted of experienced students whom we interviewed one by one. These students were familiar with several of the issues on the current website and had quite a few ideas to how it could be improved. The second group contained only first year students who were taking the course "TDT4110 - IT Grunnkurs", a course our supervisor lectures. The first year students were interviewed in a group rather than one by one. The reason for this, was simply convenience. We could not take much time from the students, and we wanted to get as much feedback on as short time as possible. We offered them free coffee as a thank you for using their time to talk with us. This group interview was a success as the students was very interested in new technology and was eager to answer and contribute to the discussion. The only downside we could see, was that the group consisted of only boys which might have caused the phenomenon group thinking[28] and might have led to a less creative session. The students were also very new at NTNU since we interviewed them only three weeks after they started their first year. This could have both positive and negative effects on the results. The positive is that we get a fresh view on how someone that is not used to ntnu.no sees the website, the negative part is that they might be afraid to speak up in a group. The group was very mixed on that part, maybe half of them participated lively in the discussion, while others just listened in on the discussion.

## 5.5 Implementing the prototype

We decided to split the website in two parts. One for current students and professors, and one for external people as media, potential students and previous students. After this decision we were suddenly left with two websites to implement. After some discussing, involving our supervisor, we decided that we had to pick one version to implement. We wanted to go deeper into only one version, to be able to fully focus on the important part which is the responsive features. Implementing both pages would have affected the quality. This would later give us the opportunity to create one version in multiple ways, to explore the different technologies in a better way. The student version felt like an natural choice since it contained the user groups that did not get as much

attention on the website at the time. Of course, we as students are part of the user group of this version as well, so that may have affected our choice.

We decided to implement two versions of the prototype, one with the use of a responsive framework, and one without any framework. This was made possible since we now only had the internal design to implement, and gave us the possibility to explore different frameworks, and coding techniques when writing a RWD website. It also gave us the possibility to explore differences, and look at advantages and disadvantages for the two solutions. The prototype without any framework was an important part, since it gave us a much better understanding on how a RWD website is built, from the ground and up. As we see it, there were only a few disadvantages. Making a prototype without a framework is more time consuming, and it takes a great deal of effort to make the prototype look pretty.

## 5.6 Feedback on the prototype

When the development of the prototypes was done, we wanted to get some feedback on them. As we started to plan the test, we realized that this would be very hard. Our implementation contains no functionality, only design and "responsiveness". We were unsure to which questions we could ask our test subjects. We could load the website, and ask them to look at it, and then ask for comments. But when not asking for specific things, it will be very hard for the test subject to criticize or complement something. Often in prototyping you would set up scenarios and ask the tester to go through these. You can then see if the application is intuitive if the user manages to complete the scenarios without any problems. But in our case, we do not have any functionality, so we can not ask them to use any functions, or perform any tasks. We can not ask the tester "Try to find room R1" for instance. Because in our prototype the find room function contains only an input field which is not mapped to anything, as well as a static image.

We asked one of our fellow students to test the prototypes. We started with the prototype made without any frameworks, opened it on a computer with a big screen, an iPhone and an iPad. He browsed and studied the site for some time, then we did the same with the version made with Bootstrap. After this, we asked him for any comments.

The student was a master degree student. He said that, for him, the "find room" function was not that important, since he has been at this school for five years, he knows where most of the rooms are. At least the ones he are using. He also said that the contact section at the bottom was good. The schedule section could have been scaled down differently when the screen size decreased. The

schedule could have scaled down to only show the schedule for the current day. This was actually a really smart idea which we had not thought of ourself, and could be something to keep in mind if we were to actually implement the site. He also mentioned that he would like a link to the dinner menu at the canteen, and said we had a nice size of our buttons.

Basically, there was not very much feedback. And as we understood that this testing and feedback led to so few comments, we decided to not continue with more test subjects. Our supervisor agreed to this and said it was probably more interesting to write about the technical features of the prototype.



# Chapter 6

## Data gathering

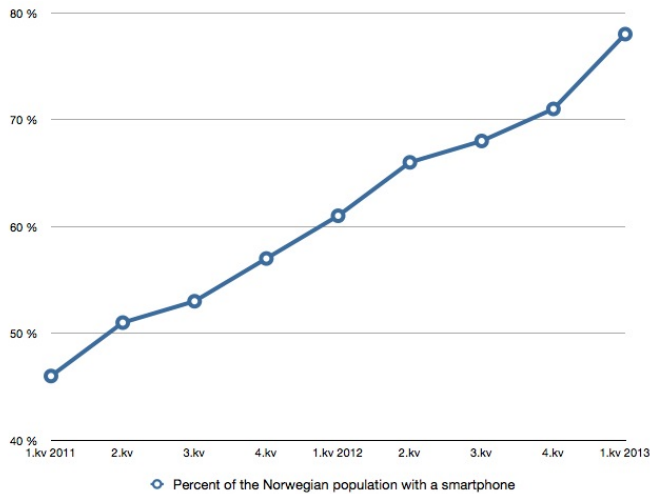
### 6.1 Questionnaire results

The questionnaire was sent out to both students and professors at NTNU and was answered by 197 people. The full results are available in Appendix B.

The results shows that quite a large amount of people (51%) rather uses Google instead, or in addition to going directly to ntnu.no looking for the information there. This might confirm that the NTNU website does not have a very high user friendliness. Still, 76% visits ntnu.no when looking for information related to NTNU.

When asked which type of unit the users most often use to visit ntnu.no, 96% answers that they most often use a PC or Mac. This is not a surprise since we already have seen the website statistics, in addition to see how unusable ntnu.no is on smaller screen devices. However, it does not mean that users do not want to seek information on their smartphones. As illustrated in Figure 6.1, the number of people with smart phones in Norway has increased dramatically the last years. Medienorge shows a survey TNS Gallup conducted that states 78% of the Norwegian population had a smartphone in the first quarter of 2013. If we assume that the younger generation uses technology more than the older, we can assume that the percent might be even higher for the users of ntnu.no. The point being, almost every user of ntnu.no have a smartphone in their pockets at all times.

When analyzing which pages on the NTNU websites that are most visited on different devices, we got some confirmations on our previous thoughts. We said that we believed the "find room" function were more important on smarth-



**Figure 6.1:** The number of people with smartphones in Norway has grown dramatically over the last years [29]

phones than on a computers or tablets. As we see in Table 6.1, the "find room" function is the most important function on the smartphone. It is the fourth most popular function on tablets, but its not even on the top 10 for computers.

Furthermore we can see that courses is popular on the computer and tablet, but not that important on a smartphone. "Its learning" is more popular on the smartphone and tablet than on the computer.

PC/Mac	Tablet	Smartphone
1. Emner	1. Emner	1. Finn rom
2. Studier	2. Its learning	2. Its learning
3. Its learning	3. Studier	3. Emner
4. Innsida	4. Finn rom	4. Studier
5. Universitetsbiblioteket	5. Innsida	5. Innsida
6. Studweb	6. Universitetsbiblioteket	6. Studweb
7. Sivilingeniør og arkitekt	7. Om NTNU	7. Studier
8. Om NTNU	8. Sivilingeniør og arkitekt	8. Om NTNU
9. Finn en ekspert	9. Student i Trondheim	9. Universitetsbiblioteket
10. Student i Trondheim	10. Finn en ekspert	10. Sivilingeniør og arkitekt

**Table 6.1:** The top 10 pages people are visiting on the different platforms. The highlighted cells is considered to be content for usergroup 2, which is the usergroup we have made a prototype for.

## 6.2 Interview results

### 6.2.1 Experienced students

We asked a group of students of their opinion on the ntnu.no website as it is today. The students were mainly from the 4th and 5th year.

#### **NTNU website**

One of the most annoying issues with the NTNU website as it is today is that it is often hard to find what you are looking for, the information you are looking for might not be placed on the most intuitive path. To illustrate this we can look at how hard the map of NTNU was to find, you had to find the tab "om ntnu" and then click "kart" in a cluttered sublist with 15 different links. This is just one example of bad usability. Furthermore another issue, and probably the issue most of the interviewees took notice of, was that ntnu.no is not optimized for the mobile platform. One student said "When i tried using ntnu.no on my mobile phone i couldnt see or click on any information without zooming in on that specific button or text, so i gave up and used my laptop instead". Many of the students also mentioned that the most important feature when using ntnu.no on the mobile phone would be to find a room at NTNU. This is not very easy to find as stated first in this section, and should instead be a function on the front page for best accessibility. Finally many of the students mentioned the bad search function on the webpage, Google is often preferred as a better option for finding information on the NTNU website.

#### **Initial design**

After finishing the sketches for our initial design we did some interviews to get the users to comment on the first sketches. The feedback was mainly positive compared to what they felt was offered by the current ntnu.no website.

There were a few issues that came up during these interviews. The news part of the design was a component we took from the old NTNU website, but many of the interviewees said that this was something they did not see as prioritised content, and if it was to be an important part of the design it should contain more interesting news, maybe in the form of personal customized news for each student, so that you could have the possibility to set up what you want to receive news from.

On the desktop sketch in Figure 7.4 we choose to prioritise the search for courses part lower than on tablet and mobile, but even if it still was visible

without having to scroll or clicking a link, some students felt it was not easy enough to spot. They felt it should be more prioritised and put higher up on the screen. This is also backed up by the questionnaire results seen in Figure B.6 where 76% said they were looking for information about courses then browsing ntnu.no on the desktop platform.

Some other interesting ideas were also mentioned during the interviews. A calendar with important dates, either as a full calendar or a notification widget on the front page. NTNU already has a function like this called academic calendar [30], but its not very well implemented. Another very interesting feature that the students came up with was a map with all the different student activities and leisures. Where to find the different cafeterias and cantinas, where to find the NTNUI facilities, and where to buy books was some of the examples.

### 6.2.2 New students

We organized a group interview session with a group of new students to discuss the NTNU website compared to our initial design. This was a very productive session and got us some good feedback from a quite different perspective(since these students has almost none experience with ntnu.no) than we had earlier results from. We will now list the results from this interview session.

The NTNU website is not very easy to traverse, things can be hard to find and is not always on the place you expect them to be. On a mobile phone the website is to small, and is not really usable at all. One student said "I should not have to use the zoom function as much as i did when browsing the current site with my mobile phone". To click on content links you have to zoom, or else its almost impossible to hit the correct button. The new students usually use the NTNU website when they know exactly what they are looking for, information about a course, rooms or class schedules is the most used features.

The feedback we got on our initial design was mostly positive, they really liked the map where you should be able to find out where the different rooms and buildings are. A student commented "I couldnt even find the map on the current website when i looked for it the first time. With a map easily vislible at the front page like you have designed it, I can easily find it when i need it. Then i would not have to search Google when trying to find out where my next class is." A good idea was also to be able to extend the class schedule to improve visibility. The students said the schedule was easy to read on a large screen, but in our design the text got a bit tiny on the smaller device sizes. By being able to enlarge the view on smaller devices, with for example a "click to zoom" function. Which will temporarily overlay the other content, and retract when you click it again.



# Chapter 7

## Design

This chapter describes the design of the two different prototype ideas. One for the external website and one for the internal website. We designed using Axure, an interactive wireframe software and mockup tool <sup>1</sup>. We have images of the design on the three platforms, mobile, tablet and desktop. The initial design phase contains sketches for both the external and internal version, even if we only are going to make a full prototype for the internal version.

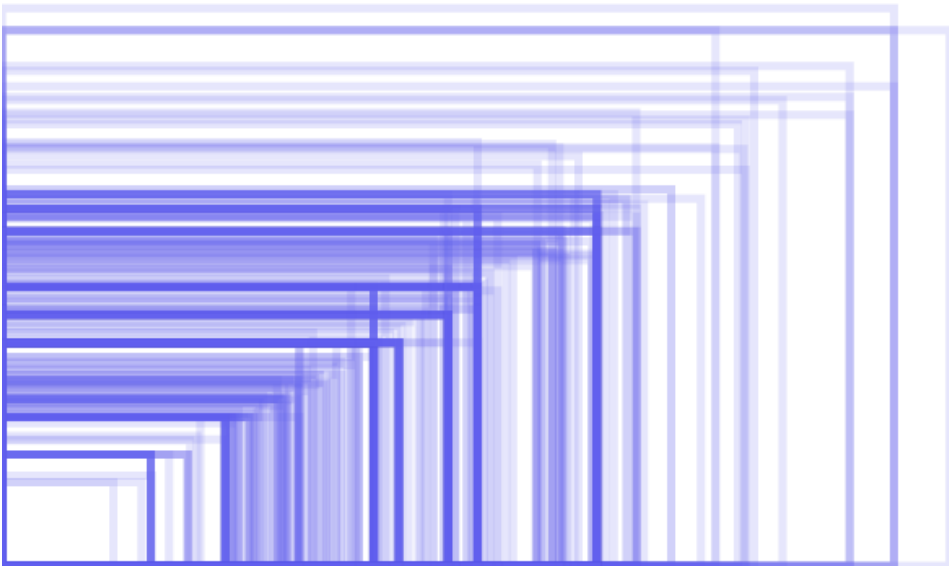
When designing for the various screen sizes, there were many interesting factors we had to consider. First of all there is not easy to design for specific screen sizes because of the massive amount of different sized devices. In Figure [31] we can see all the different screen sizes on the android platform. As you can see its not viable to make specific rules, and try to design for each screen size. So another approach was needed, and the use of responsive web design was an obvious choice.

According to Jacob Nielsen [32] its often a correlation between how much information a webpage have, versus how much the user actually reads. This tells us that we have to limit what content is visual to the user, and this is where our collected data helps us. The users have pointed out the most important features of the site, and we will try to implement them in a way that hopefully will be a lot less cluttered than todays design.

The users are also according to Nielsen[33] selective in what information they actually look at on the screen, therefore we have to carefully plan where to place the different elements in the prototype. We plan to use gestalt theory when building up the page. Gestalt theory[34] consists of several laws on

---

<sup>1</sup><http://www.axure.com/>



**Figure 7.1:** The screen size problem

how elements are perceived by humans. One of the laws states how similar elements that are placed next to each other, can be perceived as a unity. By giving our elements same styling and design it gives our design a more uniform look. The elements look like they are fitted together even when they still are different shapes.

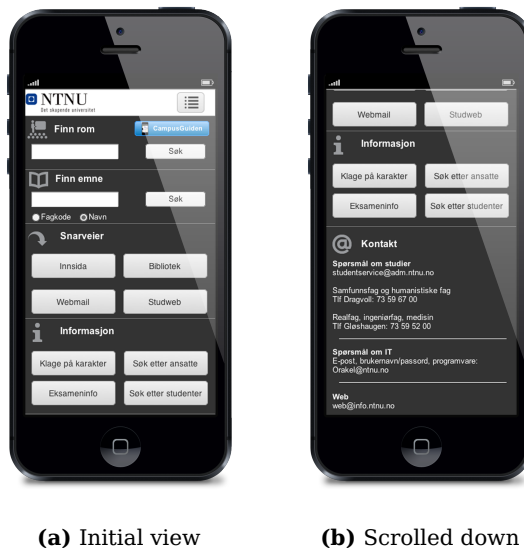
## 7.1 Initial design

After gathering data from the questionnaire and interviews, we got some interesting pointers to base our work on. Many of the answers gathered tells us that we should focus more on mobile design. On one hand, almost no one uses ntnu.no on the mobile platform, and on the other hand many of the answers we got from the open ended questions in the questionnaire says that ntnu.no is not very usable on the mobile platform. Our approach to these answers is to try to implement the webpage by using the mobile first[19] method.

On the background of our own use of ntnu.no, a questionnaire and interviews with user groups we created some initial mock-ups of how we thought the the website should look like. Since we are working with mobile first, the mobile versions where created first.

### 7.1.1 Internal design

This is the website the internal user group should access. The internal design consists of all the elements that the students and professors need to assist them in their everyday work. We know that these people are quite familiar with the university in many cases, but they could also be first year students. All the material that displays nice pictures and information of the school are excluded in this version. Likewise for the material for journalists and companies. This of course gives us more space for the functions that are actually important for this user group.

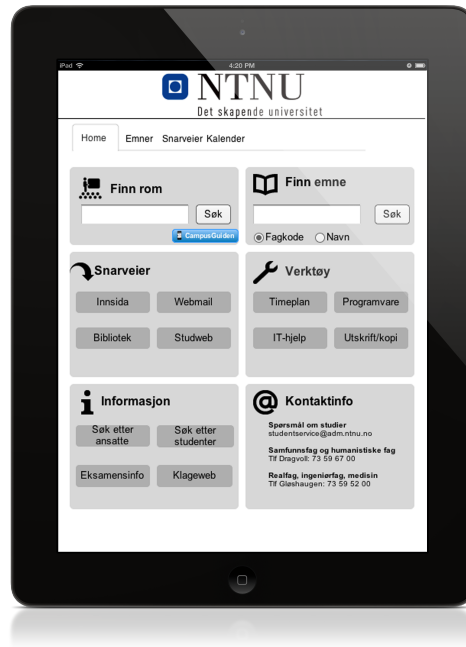


(a) Initial view

(b) Scrolled down

**Figure 7.2:** Initial sketches of the mobile version for students and NTNU employees

The internal design for mobile devices is presented in Figure 7.2. The find room function has proven to be the most important function on smartphones through the questionnaire. This is therefore the first element on the mobile version. Furthermore, the courses has been proved to be very popular as well, so this will be the second element. The rest is basically links to other functions that has proven to be important and useful for the students and employees at NTNU. At the bottom we have some contact information in case someone needs help quickly with something.



**Figure 7.3:** Initial sketch of the tablet version for students and NTNU employees

When we scale up to the pad version which is illustrated in Figure 7.3, there is not big differences from the mobile version. It is basically the same content, except for an added section: "verktøy". The sections of course take up more space on the screen, since have more space to take from. The content is therefore wrapped in boxes and placed around on the screen. The prioritizing is the exact same as on the mobile version.

The menu icon, or the "hamburger icon" is also removed on the pad version and replaced with a regular menu. At this point, we have not thought through exactly which items should be in the menu. If there had been too many menu items to fit in on the pad version, you would want a menu icon here as well as on the phone version.

You can notice some color differences on the background and buttons. This is not intentional, the theme will be consistent across all screen widths in the final implementation.



**Figure 7.4:** Initial sketch of the PC version for current students, and professors

When the site is scaled all the way up to the desktop version we can clearly see many changes. Firstly, the find room function has an imbedded map which will instantly show the location of a room as you type it in and select the right room from the suggested list. The "timeplan", which on the two earlier versions only was a link is now also embedded right into the front page, so the student can view its schedule quickly. We have a couple of new sections: news and a rss feed. The news are supposed to be news regarding things happening on campus or other things regarding the students and professors. The RSS feed is supposed to contain more "quick messages", for instances "The it systems are experiencing some technical problems, and will be up within one hour", or "There will be an open course for study techniques on Monday at R4".

The shortcuts and find course are poorly placed on this PC version, but we will try to find a better placement for them in the real prototype implementation. Since they are important, they should have better visibility.

### 7.1.2 External design

The External design tries to give future students, and parents of future students a possibility to look through the different studies and their respective

courses.

Even though this page focuses mostly on future students, we have not forgot the other user groups. People that needs to acquire experts (journalists), and alumni students(former students), has their own parts of the page.



(a) Initial view

(b) "Studietilbud"

**Figure 7.5:** Initial sketches of the mobile version for future students, former students and media

The mobile version for external users are presented in Figure 7.5. Displaying the studies was the intention here, but if we listed all the studies directly on the mobile webpage, the users would not see more than just the studies. It was troublesome to come up with a solution that enabled us to show all the studies, and at the same time let the users see the other elements without having to scroll far down on the page. The solution we came up with was to have a dropdown view. As you can see in Figure 7.5a we have designed this as a dropdown button. We gave each study area a personalized icon for easier recognizability and better looks. When looking at the list ntnu.no has been using to list up studies, the usability does not seem to be a factor that was considered very much. This list is shown in Figure 7.6. The list is hard to read, because some of the items are placed in groups, and the names are long. The old list does not contain icons either. We understand that the reason for grouping some studies together, is because each group is connected to a certain faculty, but for a future student we believe it only will be confusing.

Additionally we have placed a carousel at the top, an element which cycles

through pictures of the school. These pictures will help build NTNU's image, and make it appealing to future students. The school wants to sell itself as an interesting "product" to the future students, to gain more applicants. The "carousel" will help make the future students think "This looks like a good university, I want to study here".

### Studietilbud

<a href="#">Sivilingenier- og arkitektutdanning</a>	<a href="#">Språk og litteratur</a>
<a href="#">Informasjonsteknologi og informatikk</a>	<a href="#">Mediefag og kommunikasjon</a>
<a href="#">Matematikk og naturfag</a>	<a href="#">Estetiske fag, kunst- og musikkfag</a>
<a href="#">Medisin, helse- og sosialfag</a>	<a href="#">Historie, religion, kultur- og idrefag</a>
<a href="#">Idrettsfag</a>	
<a href="#">Samfunnsfag og psykologi</a>	<a href="#">Lærer- og lektorutdanning</a>
<a href="#">Økonomi og administrasjon</a>	<a href="#">Pedagogiske fag</a>

**Figure 7.6:** NTNU's listing of studies on their frontpage 23. March 2013

When we scale the browser up to the tablet size, we get enough space to expand the studies list. Although, if we have it as the top level element, the other buttons will be hidden. This is not ideal, since the other buttons are really important as well. The buttons has information the two other user groups, alumni students and journalists need. This is also the only features that are relevant for these two groups and should have a higher priority. The study section is mainly targeted at the future students, and this element is still visible if we implement it below the other four buttons. If the users needs to use the scrollbar to see these buttons, the site could possibly loose visits from the user groups that does not have an interest viewing the studies section.

When we get to the computer version of the webpage, we gain a lot of space. This gives us the freedom to add additional elements. We still have the carousel and the studies, as they were on the tablet version. But we have expanded the buttons to boxes, containing more specific links to the content they contain. Instead of having a button with a link to a new page with even more links that have content journalists want, we have a box with the relevant links implemented on the front page. This includes links to an expert list, research central and Gemini (research magazine). The same applies for the three other buttons we implemented on the tablet version.

Additionally, we have added a news section, appealing to all the user groups. This section can hold interesting news stories regarding NTNU.

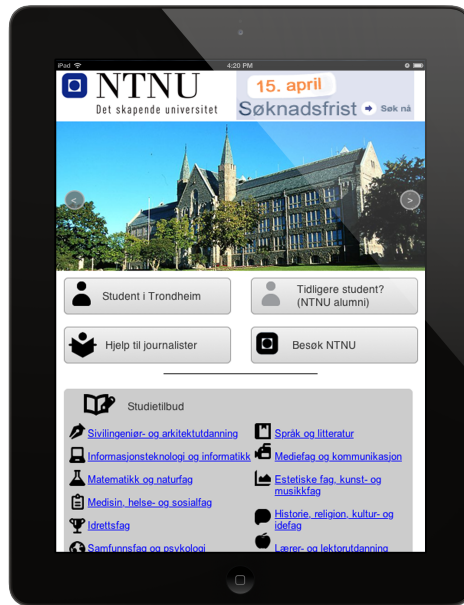


Figure 7.7: Initial sketch of the tablet version for future students, former students and media



Figure 7.8: Initial sketch of the PC version for future students, former students and media



# Implementation

## 8.1 Introduction

This chapter will give an overview on how the programming of the prototype was conducted, it will contain two separate parts. One for the prototype where no frameworks were used, and one for the prototype coded with a framework. We will try to find some advantages and disadvantages with the two, and try to summarize the coding techniques.

## 8.2 Prototype design

After reviewing the initial design mock-ups, there were several issues that arose. A few of them was simple design flaws that we knew was there, and fixed in the new prototype design. An example of this is from the desktop version of the site. The shortcuts and find room functions were placed at the bottom right of the site. The reason at the time was that this location was the most suitable, considering the size of the element and the space left in the design. Then we remembered that these functions were actually highly important according to the statistics, and was indeed proven right in the questionnaire and interviews.

There were also some inconsistencies between the different device versions, but this was mostly minor differences with low importance. There were difference in the grey box colour and button colours.

There were also features that were missing from one screen size to another, this was intentional. Although, as we continued with our research, we noticed that some of the features that we would have removed on some resolutions actually were needed on all device types. To make content completely unavailable on a device type is quite drastic, and we discovered that we could not do this on the website.

When we presented the design sketches for our interviewees, several of them expressed that the changes we had done to the design improved the website. Still, some of them could point out changes they would have made to the design. The menu button in the top bar on the mobile sized design, should be on the left instead of the right side. This is what most users are accustomed to since many of the large Internet websites like Facebook and Gmail use this in their mobile versions, according to the students. Although, what they are referring to is a swipe out menu that are available in some mobile applications as Facebook, Gmail and YouTube. In these swipe out menus, it is not needed to hit the button, you can also swipe from the far left of the screen towards the right. Using gestures like this is not common on websites, but we often find it in mobile applications which are downloaded to the phone. On our website this icon does not open a swipe out menu, but a menu list that drops downwards. When the users implicated that they wanted the button on the left side, we listened and chose to implement it this way. But on afterthought we believe our choice: the menu button on the right side, is a preferable choice. Quite often, users use their smartphone with only the right hand when browsing. As Wroblewski points out in his book; *Mobile First*[19], some areas of the screen is more reachable when using the phone this way. See Figure 8.1. Placing the menu button on the top right side holds the button within "ok" reach length. Although, placing the button on the top left side makes it hard to reach.

### 8.3 Programming without frameworks

One of the problems with code reuse can be that the developers does not gain knowledge about the elements within the library, framework or technology they are reusing. In our implementation we wanted to gain as much knowledge about RWD as possible, including possible issues and possibilities. In light of this, we have chosen to make two implementations of the same website: One without any frameworks to help us learn more about a responsive implementation, and one with the use of a framework to ease the responsive part of the implementation.

As we see it, the advantages of writing all the front end code from scratch is that you have more control of your code. You can customize everything to the smallest detail.



**Figure 8.1:** When holding a mobile phone with only the right hand, the different areas varies in how easy they are to reach with the thumb[19].

Programming without frameworks means no frameworks that helps the creation of the responsive part of the website. Standard libraries such as jQuery and Font Awesome does not ease the responsive creation, so we will still be using these frameworks.

### 8.3.1 The layout

Our layout is based on containers. Each function is embedded within a box. We want the boxes to appear in a specific order. We will use two containers per row in the largest desktop version, and as the resolution decreases some of the boxes will fill a whole row. On the lowest resolution, the boxes will all fill an entire row.

The strategy will be to follow Ethan Marcotte's guidelines[35], which were presented in Section 3.1. We will firstly try to create a responsive grid, and then fix the rest using customized media queries. When creating our grid, we will never set a static width on any element (px), but instead use percent.

As we know, in CSS id's are individual, meaning they can only be applied to one element, meanwhile classes can be applied to multiple elements. We use a class to specify the basic styling of our boxes, shown in Listing 8.1.

**Listing 8.1:** Box styling

```
1 | .box-container{
2 |   border-radius: 5px;
3 |   margin: 5px;
4 |   background-color: #F0F0F5;
5 |   overflow: hidden;
6 | }
```

For each individual element, we also have an id to specify more styling, for example the see the styling for "find room" which are presented in Listing 8.2.

**Listing 8.2:** Box styling for an individual element: Find room

```
1 | #findRoom{
2 |   float: left;
3 |   width: 48%;
4 |   margin-bottom: 5px;
5 | }
```

This means that the "find room" function will take up about 50% of the browser width at all times. The reason that we have not written `width: 50%` is because the margin is using up a few percent of the total width. With 50 % width for each box-container the total will exceed 100% and only one of the two elements will fit on each row.

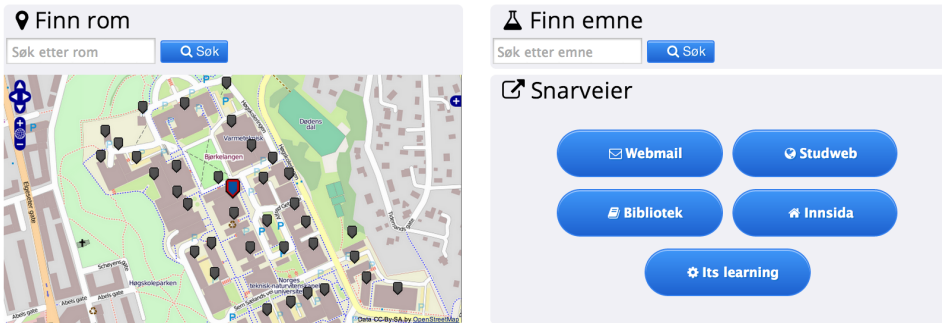
### 8.3.2 Challenges

#### Div heights

For our layout, we want the boxes on the same line to have the same height, so that we do not get any blank spaces, as the pre masonry design in Figure 3.5. To achieve this, we Figured that we could set the same static height on both elements. However, if the box contains images, which of course needs to be flexible, then the box height needs to be set to `auto`, to adjust to the image size.

Consider Figure 8.2, here the "find room" function contains an image in our case, but this would be a real clickable map in the real website. Still, we think the object would be inserted in the same way, and that it would be ideal to let the object fill the entire box by width. As we want to maintain the aspect ratio, the object will adjust in height, meaning the box container also needs to do this.

So, what happens to the two other boxes on the right side of "find room"? We Figured that we need to set those box' height to the same height as the first box. There is no way of doing this with plain CSS. We can not say `#div1 {height:auto;} #div2 {height:#div1.height;}`, or something similar that makes



**Figure 8.2:** The first two elements in the design

the height of `div2` to adjust according to `div1`'s height. Because of this we are forced to use JavaScript to achieve the wanted result.

Consider Listing 8.3. The resizing needs to be done every time someone enters the website, but also every time someone readjust the browser width. We call a function on page load by including the function name in the `onload` argument of `body`: `<body onload="bodyLoad()">`, and use jQuery to call a function each time the browser gets resized: `\$(window).resize(function())`<sup>1</sup>. Within the function, we use jQuery again to set the height of the `#topRightWrapper` to the height of `#findRoom`<sup>2</sup>. The `#topRightWrapper` is a wrapper that includes the find course and the shortcuts elements, this should now scale according to the find room element. As you can see, this is only done if the browser width is more than 767px wide, because if it is smaller, the layout is different. At the smaller resolution, the "find room" function uses the whole width of the browser, which then leads to the `#topRightWrapper` to share the browser width with `#news` and `#information` instead, as illustrated in Figure 8.3.

**Listing 8.3:** `resizeDiv()`

```

1 | <body onload="bodyLoad()">
2 |
3 | <script>
4 | $(window).resize(function()
5 | {
6 |     resizeDiv();
7 | });
8 |
9 | function bodyLoad()
10 | {
11 |     resizeDiv();
12 | }
```

<sup>1</sup><http://api.jquery.com/resize/>

<sup>2</sup><http://api.jquery.com/height/>

```

13
14 function resizeDiv(){
15
16   if($(window).width()>767)
17   {
18     $('#topRightWrapper').height($('#findRoom').height());
19
20   }else if($(window).width()>480){
21     $('#topRightWrapper').height("auto");
22     var height = $('#findCourse').height() + $('#shortcuts').height()
23     ;
24     $('#news').height(height/2);
25     $('#information').height(height/2);
26   }
27 }
28 </script>
</body>

```

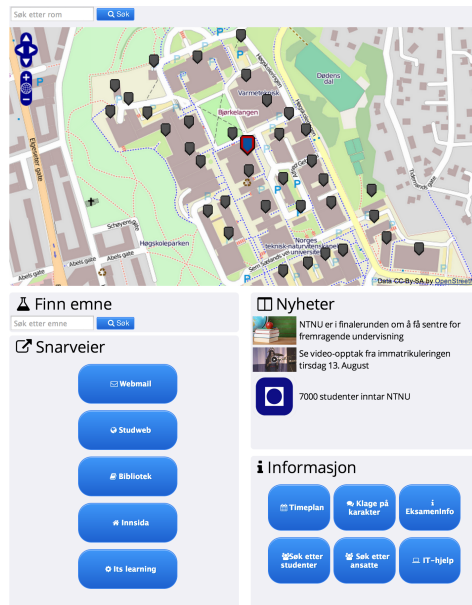
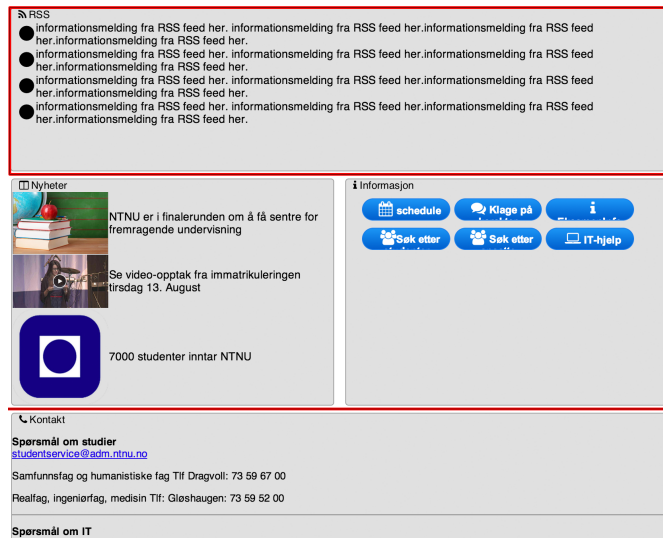


Figure 8.3: The #Toprightwrapper at 767px wide resolution

### Changing the priority of content

From the results of our design phase, we discovered that some content might be less important in some screen sizes than in others. One example of this is

that the RSS feed gets less important on small tablets and mobile phones than on a computer screen. We therefore want the RSS feed to change location by dropping down beneath the news and information boxes, as shown in Figure 8.4.



**Figure 8.4:** As the screen gets smaller we want to give less priority to the RSS feed by moving it beneath news and information

All our content are div's that are placed with the css float property. It is not possible to change the order of these div's by only by using CSS. We concluded that having two div's at the two different locations was the only way to go.

Consider Listing 8.4 for the HTML code. We already have a `bodyLoad()` method which are being called on pageload. Here we can take the content from `#rss-large-screen` and copy it to `#rss-small-screen`. If we inspect the CSS in Listing 8.5, we see that the initial display of `rss-small-screen` is none. If the screen is narrow, that is, bellow 767px, the media query will trigger. It will disable the `#rss-large-screen` by setting its display to none, and give the `#rss-small-screen` an inline visibility.

**Listing 8.4:** Priority change HTML

```

1 <div id="rss-large-screen" class="box-container rss">
2 <div class="sectionHeader">
3 <i class="icon-rss section-title"></i> RSS
4 </div>
5 <table>
6 <tr>
7 <td><i class="icon-circle"></i>

```

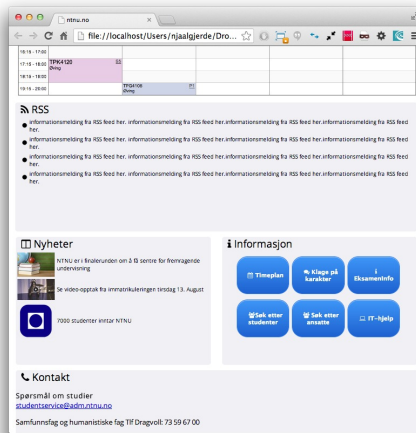
```
8 </td>
9 <td>informationsmelding fra RSS feed her. informationsmelding fra RSS
   feed her.informationsmelding fra RSS feed
   her.informationsmelding fra RSS feed her. </td>
10 </tr>
11 </table>
12
13 </div>
14
15 <div id="rss-small-screen" class="box-container rss">
16 </div>
17
18 function bodyLoad(){
19   $('#rss-small-screen').html($('#rss-large-screen').html());
20   resizeDiv();
21 }
```

**Listing 8.5:** Priority change CSS

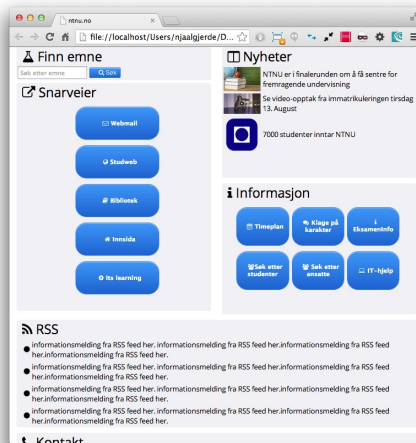
```
1 #rss-large-screen{
2   float: right;
3   width: 48%;
4   overflow: scroll;
5 }
6
7 #rss-small-screen{
8   float: left;
9   width: 100%;
10  margin-right: -1px;
11  display: none;
12 }
13
14 @media screen and (max-width: 767px) {
15   #rss-large-screen{
16     display: none;
17   }
18   #rss-small-screen{
19     display: inline;
20   }
21 }
```

The difference from going from 769px screen width to 752 screen width is illustrated in Figure 8.5 and 8.6, thus illustrating the repositioning of the rss div.





**Figure 8.5:** The prototype displayed in 769px, before the media query at 767px

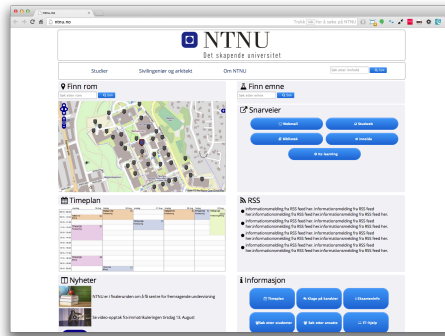


**Figure 8.6:** The prototype displayed in 752px after the media query has been triggered

### 8.3.3 The result

The HTML and CSS code for the result of this prototype is available in Appendix C, and also as an attachment. In this section we will shortly go through the result of the website prototype created without any responsive frameworks. To see how a responsive website will look on a device with a smaller screen width, we can adjust the width of our browser window on a computer. We will

start with a quite wide screen width, and then decrease it step by step to see its changes.



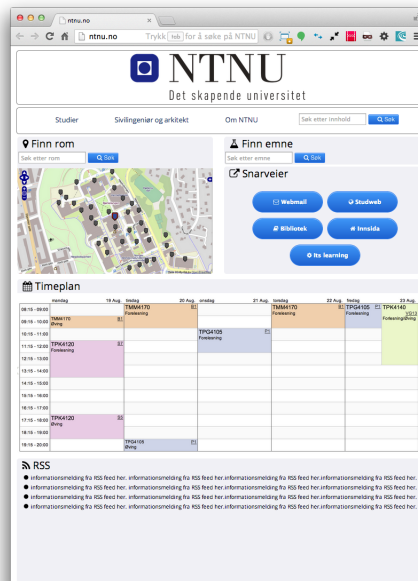
**Figure 8.7:** Non framework 1463px



**Figure 8.8:** Non framework 1084px

In Figure 8.7 we can see the page at its widest. It is added padding on both sides of the content, to avoid filling the entire browser width with content. As we see in Figure 8.8 this padding is removed at 1463px and the content now fills the width 100%.

Consider Figure 8.9. When decreasing the screen width down to 1084px, we can observe that the schedule now fills the width 100%. The reason for this change is the decreased visibility on a small screen. On this screen width, when filling only 50% of the screen, it becomes difficult to read the text.

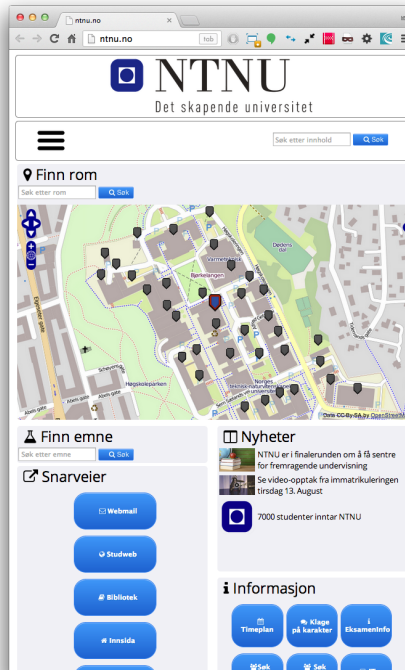


**Figure 8.9:** Non framework 779px

We continue to decrease the screen size down to 660px, shown in Figure 8.10. Note that the iPad is running at 768px width, so this is the version we would get on an iPad. We see that the find room element now fills the screen width 100%. This, simply because it gets too small when displayed at 50% of this screen resolution. As the find room fills the whole width, both the find course and shortcuts falls down on the page. We can also notice that the schedule is now removed. Although, it is not completely removed, the link for schedule is still available as a button under the information section. The content is still available, but less visible. Having both the find room and schedule filling the screen 100% would not give any room for the rest without scrolling down on the page. The shortcuts and information should be visible to the user since these are highly used functions.

We can also note us that the top menu has now turned from a regular menu into a menu icon, the "hamburger icon".

Reducing the screen width down to 424px wide, as illustrated in 8.11b, we can see that the webpage now resembles a mobile site. This is the version that the most mobile phones will get, including the iPhone. At this size, we do not have space to display the map for find room any more. All the boxes here needs to fill the width 100% so everything gets as readable as possible. The most used and important functions are prioritized and placed at the top. This includes



**Figure 8.10:** Non framework 660px

Find room, find course, shortcuts and information. To see this version running on a real iPhone, consider Figure 8.11a which is a screenshot of the website running on the iPhone simulator running iOS7. This environment is exactly as a real device, although we have of course tested the website on the real physical device as well.



(a) Non framework displayed on iPhone

(b) Non framework 424px

**Figure 8.11:** The mobile version displayed in two ways

## 8.4 Programming With frameworks

### 8.4.1 Layout

The Bootstrap prototype is built up using a standard Bootstrap grid system which is illustrated in Figure 8.12.

.col-xs-12 .col-sm-6 .col-md-8	.col-xs-6 .col-md-4
.col-xs-6 .col-sm-4	.col-xs-6 .col-sm-4 .col-xs-6 .col-sm-4

**Figure 8.12:** Bootstrap column system

The grid system is set up with a maximum of 12 columns, rows is used to create horizontal groups of columns. To make the rows align properly we have to put them inside a container class element, which is illustrated in Listing 8.6. The content for each section is residing inside a Bootstrap element called a "well", giving styling that sets a border and background color to the section. The four different breakpoints built in within the framework is what we have to work

with. You can of course create your own media queries, but we wanted to test the framework, so we have only used the features of Twitter Bootstrap in this version. The "Finn rom" element is set up to use a column size of 6 on large screens, 7 on medium and small screens, and the maximum of 12 columns when the screen is extra small. This is simply done by applying several classes to the div, and our own id for convenience: `<div class="col-lg-6 col-md-7 col-sm-7 col-xs-12 well" id="rom">`. We use rows as placeholders and place the elements we want in each row, the first row consist of "Finn rom", "Finn emne" and "Snarveier". The second row consist of "Timeplan" and "Nyheter". The row feature make sure that these two elements always stay below the elements in the first row. When you resize the browser window the elements still maintain the row feature even if each element are now using the full browser width, thus keeping each row separated with a fixed padding set by Bootstrap.

**Listing 8.6:** All of the content needs to be wrapped in a .container class

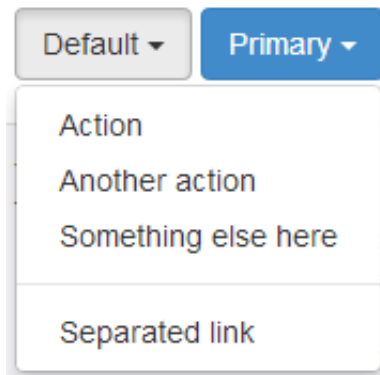
```
1 <body>
2 <div class="container">
3 <div class="row">
4 <div class="well"
5 <!--Content for a box here-->
6 </div>
7 </div>
8 </div>
9 </body>
```

## 8.4.2 Advantages

We see several advantages using a responsive framework like Twitter Bootstrap. You get a preconfigured flexible grid system that helps you organize the content of your page. Twitter Bootstrap also assist you with the design. Elements like buttons, navigation menus, and form fields are already designed for you, so you do not have to concern yourself with writing bulks of CSS to make the site look good. In Figure 8.13 we see an example of the styling set by Twitter Bootstrap on two drop down buttons. Not needing to manually style elements gives us the opportunity to quickly add elements to our page. This enables us to get results at an early stage of the implementation.

The grid system already incorporated in the Bootstrap framework helps us immensely. We can easily specify the width of each element, for each of the four screen sizes.

When using a responsive framework like twitter Bootstrap we do not have to write as much CSS, since a lot of it is incorporated in the framework. While a solution not using a responsive framework strongly relies on writing own media



**Figure 8.13:** Styled buttons in Twitter Bootstrap 3.0

queries to get the full responsive features, most of the responsive features is written directly in the HTML code when using Bootstrap 3.

Consider Listing 8.7. On the first line, we can see that the div has four classes. These classes specifies how many columns the div should span across, on the different screen widths. Twelve is the maximum for each row. You set one width for each of the predefined screen widths the Bootstrap framework have set up for us. One for large screens with 1200 pixels or more, Medium for screens larger than 992 pixels, small for screens larger than 768 pixels, and extra small for screens smaller than 768 pixels. These are shown in Table 8.1.

In some instances, the content is not optimal to be displayed on a small screen. Therefore, a valuable feature in Bootstrap, is to be able to hide content based on the screen size. An example of this is shown in Listing 8.7. Here, we hide the map in the find room function for screen sizes below 768 pixels, because it at this point will take up too much space and become hard to read. We can add the class `class="hidden-xs"` which hides the picture when the screen width is below 768 pixels.

The two images we use for the schedule and find room elements is made responsive by using a built in Bootstrap feature. We simply call this function by by using `class="img-responsive"`. These are some of many minor features that makes programming much quicker when using a responsive framework like Twitter Bootstrap.

Device size	Screen width	HTML class	Device type
Extra small devices	<768px	.col-xs-	Phones
Small devices	>=768px	.col-sm-	Tablets
Medium devices	>=992px	.col-md-	Desktops
Large devices	>=1200px	.col-lg-	Desktops

**Table 8.1:** The Twitter Bootstrap device types and their according screen widths. These can easily be targeted to select how many columns an element should span across. They can also be used to hide elements.

**Listing 8.7:** responsive features

```

1 <div class="col-lg-6 col-md-8 col-sm-6 col-xs-12">
2 <form>
3 <input class="inputfelt" type="text" placeholder="Søk på rom">
4 <button type="submit" class="btn btn-info">Søk</button>
5 </form>
6 
7 </div>

```

### 8.4.3 Challenges

When using a framework you are forced to use a specific syntax, you do not have as much control of your code as some programmers want when designing a website. When trying to fix specific design issues, an effective and common approach is to inspect the website in the browser window. Using the browsers developer tools you can effectively change the CSS in realtime, and see the changes instantly. However, when using a large front end framework as Twitter Bootstrap, containing an immensely large amount of CSS, this becomes more complicated. It can be hard to spot and override Bootstrap functionality, because of all the different classes and styles. Figure 8.14 illustrates an instance when inspecting the find room function. Because Bootstrap adds styling to the different elements, this might contradict with the styling you add yourself. Because of this, it is important to review the Bootstrap styling, before you add your own. One example is the button styles, Bootstrap sets a certain padding and margin to the button element you have to be aware of. This is important if you want to customize the placement or look of your buttons.

When using Bootstrap, the design of your site will perhaps be pleasing to the eye. Still, there might be a downside with the premade styling. There is probably numerous other sites using the exact same styling as your site. As the framework becomes more popular, the more pages with the same look can emerge. This can make your site feel less special. If it is important for your site to have a unique design, the Twitter Bootstrap framework might not be



```

}
-----
.col-md-1, .col-md-2, .col-md-3, .col-md-4, .col-md-5, .col-md-6, .col-md-7, bootstrap.css:1074 @media (min-width: 992px)
.col-md-8, .col-md-9, .col-md-10, .col-md-11 {
  float: left;
}
.col-sm-2 {
  width: 16.6667%;
}
bootstrap.css:936 @media (min-width: 768px)
.col-sm-1, .col-sm-2, .col-sm-3, .col-sm-4, .col-sm-5, .col-sm-6, .col-sm-7, bootstrap.css:920 @media (min-width: 768px)
.col-sm-8, .col-sm-9, .col-sm-10, .col-sm-11 {
  float: left;
}
.col-xs-2 {
  width: 16.6667%;
}
bootstrap.css:872
.col-xs-1, .col-xs-2, .col-xs-3, .col-xs-4, .col-xs-5, .col-xs-6, .col-xs-7, .col-xs-8, .col-xs-9, bootstrap.css:854
.col-xs-10, .col-xs-11 {
  float: left;
}
.col-xs-1, .col-xs-2, .col-xs-3, .col-xs-4, .col-xs-5, .col-xs-6, .col-xs-7, .col-xs-8, .col-xs-9, bootstrap.css:800
.col-xs-10, .col-xs-11, .col-sm-1, .col-sm-2, .col-sm-3, .col-sm-4, .col-sm-5, .col-sm-6,
.col-sm-7, .col-sm-8, .col-sm-9, .col-sm-10, .col-sm-11, .col-sm-12, .col-md-1, .col-md-2, .col-md-3, .col-md-4, .col-md-5,
.col-md-6, .col-md-7, .col-md-8, .col-md-9, .col-md-10, .col-md-11, .col-md-12, .col-lg-1, .col-lg-2, .col-lg-3, .col-lg-4,
.col-lg-5, .col-lg-6, .col-lg-7, .col-lg-8, .col-lg-9, .col-lg-10, .col-lg-11, .col-lg-12 {
  position: relative;
  min-height: 1px;
  > padding-right: 15px;
  > padding-left: 15px;
}
bootstrap.css:289
*, *:before, *:after {
  -moz-box-sizing: border-box;
}

```

**Figure 8.14:** Bootstrap classes when inspecting

a good solution for you. Although, as Bootstrap has gained an incredible big popularity, themes to the framework has also emerged. Own sites with both free and paid themes exists for Twitter Bootstrap.

Bootswatch<sup>3</sup> has several free to use templates available for download . Wrap-Bootstrap<sup>4</sup> has numerous of templates as well, these cost money, but you can find themes for a very reasonable price. Still, if you really want your site to be unique, you would need to design your website from scratch. In most cases, it will exist other sites using the same theme as you.

From a technical view, we discovered that most of the issues that occurred without using a responsive framework, were still persistent when using Bootstrap. There was no solution for setting the same height on multiple div's, as described in Sub Section 8.3.2.

#### 8.4.4 The result

We quickly discovered that creating the prototype using a framework would save us time. Twitter Bootstrap comes bundled with most of the CSS styling

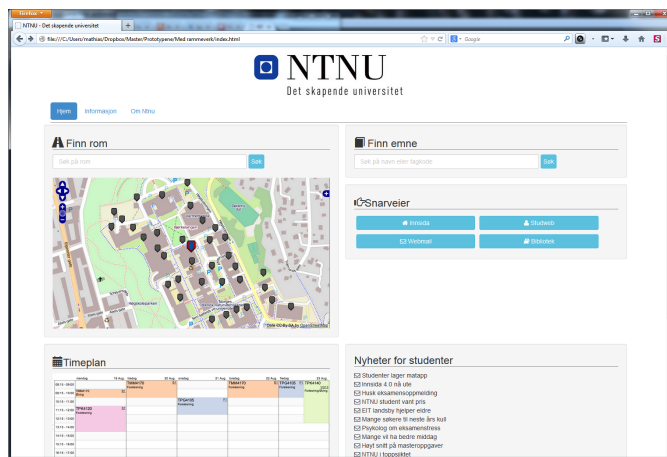
<sup>3</sup><http://bootswatch.com/>

<sup>4</sup><http://wrapBootstrap.com/>

needed to create a good looking responsive website. This gives the opportunity to focus more on the content to implement on the website.

In this section, we will inspect the prototype made with Twitter Bootstrap. We will explain differences that is worth making note of, as the screen width reduces.

The largest media query, set by Bootstrap at 1200 pixels and up is illustrated in Figure 8.15. We have two sections both taking up 50 % of the page, the media query taken from the Bootstrap CSS file is shown in Listing 8.8. The media query sets a fixed width in pixels on the `.container`, which is wrapped around all of the content. `.col-lg-6 {width:50\%;}` sets the width of the sections to 50%.



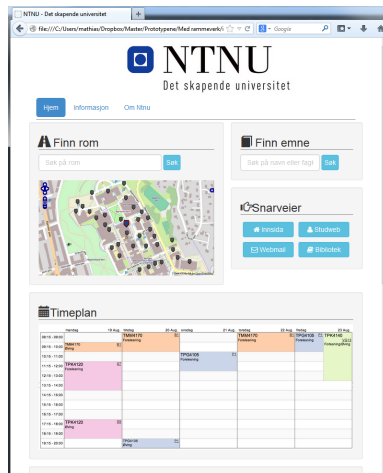
**Figure 8.15:** Framework at 1527px

**Listing 8.8:** Bootstrap media query

```
1 @media (min-width: 1200px) {  
2   .container {  
3     max-width: 1170px;  
4   }  
5   .col-lg-6 {  
6     width: 50%;  
7   }
```

When we resize the window below 1200 PX, we can see some changes, these are illustrated in Figure 8.17. The two image elements; "Timeplan" and "Finn rom", were starting to get too small. "Finn rom", "Finn emne" and "Snarveier" all had an initial value of 6 in column width. We decided to increase the "finn rom" element by setting the column width to 7, the column size for the "finn emne" and "snarveier" then had to be decreased to 5. We could have used the

same idea for the "Nyheter" and "Timeplan" elements but saw that a column size of 5 would be too small for the "Nyheter" element. The solution we decided on was to give both a column size of 12, thus splitting them into two different rows.



**Figure 8.16:** Framework at 799px

The last breakpoint we can notice changes at, is at 768 pixels. We can see the changes illustrated in Figure 8.17. Here we really had to take the screen size limitations into consideration. For the "Find room" function, we thought that filling only about 20% of the screen height would not be sufficient. For the map to be useful, it needs to have a certain size, so that the user can fit their fingers inside the box to zoom and navigate. Filling the whole screen height is of course not an option, since this would hide all the other items before scrolling. Because of this, we decided to remove the map, and only keep the search field on this screen width.

The class schedule is also removed and can now only be accessed by clicking a link in the information box. As to the top menu, there is no longer room for all the items to be placed directly on the site. In our example we only have three links, but in a real implementation we know that it would be more. Because of this, the menu is swapped out with a hamburger icon. This icon can be clicked to get a full view of the navigation menu, which displays all the elements. Further we set all elements to use the maximum of 12 columns at this screen size. The idea is to be able to see as much content as possible on the mobile platform. Additionally one should be able to easily click the elements without manipulating the website by zooming in on the content.



**Figure 8.17:** Framework at 382px

# Results

## 9.1 Introduction

Initially we asked the following research questions:

1. How should the website be structured?
  - (a) How can we organize the website to accommodate the different user groups?
  - (b) Which functions or information should be prioritized on the different devices?
2. How should the website be created?
  - (a) Which technologies and frameworks could be used to improve the process and the result of a responsive NTNU website?
  - (b) Are there any benefits or disadvantages with the different technologies and frameworks?

In this chapter we will elaborate how we found the answers to these research questions.

## 9.2 Structuring the website

As we have presented earlier in this thesis, the NTNU website today contains a lot of content for many user groups. The website is not optimized for all the user groups. The user groups can be analyzed to be either internal or external to the university. As clearly stated in Section 4.4 and illustrated in Figure 4.2 we have suggested to divide the NTNU website into two different parts, with

their own separate web address. The external website should keep its original address `ntnu.no`. This is because it is the easiest url address, and it will be a complicated process to send a message out to all external parts, telling them that they should use a different address. The internal user group can easily get familiar with using a new address like `s.ntnu.no` or something similar.

Since we have divided the site into these two versions, it should be easier to give the users the right content. The front page should now also be less cluttered with content that some user groups find irrelevant. This solution also has negative effects. One of them being that we need two different url addresses.

When we looked at the second sub question and tried to find data on which functions or pieces of information should be prioritized on the different devices. The answer did not seem clear, and we did not manage to find a good conclusion. Based on the questionnaire and the interviews we concluded that the most important difference in content prioritization, is that the find room function is rated more popular on mobile devices than on the other platforms. At the tablet size its rated fourth place as we already stated in Table 6.1. The problem we had after evaluating the results from the questionnaire, was that there were only a small amount of people who answered which functions they were using, or wanted to use on smartphones and tablets.

To make the website have a good fit in every screen size we had to do some adjustments to the content, by removing large pictures on smaller screen sizes we can more easily fit content on the smaller devices.

### 9.2.1 Alternative prioritization

Regarding content prioritization we have been thinking about other alternatives. One of them is to let the users identify themselves when they first enter the website, this could be done by using a modal pop up<sup>1</sup> which requires the user to interact and select what user group he or she belongs to, before continuing to browse through the site. We envision that each element on the website will have a variable giving it a priority number. For each user group, it will have a different number depending how important it is for that group.

For instance, lets say the priority scale goes from 1-100, 100 being the top priority. Take the find room function from our prototype. This element could have a priority list suggested in Table 9.1.

This gives us the possibility to focus more on the user content customization. All the user groups will have their own customized version, the journalists do

---

<sup>1</sup>**Modal window:** A child window that requires the users to interact with it before they can return to operating the parent application

<b>User group</b>	<b>Priority</b>
Students	95
Professors	80
Companies	75
Default user	50
Future-students	5
Journalists	5

**Table 9.1:** Priority list for an alternative approach to separating user groups

not need to get their version of the site ruined by elements that is important to future students, and so on. This way we hope to give the users the best experience possible by giving them a feeling that the site is specially tailored for their needs.

With this solution, we will not have the issue with having two separate addresses, but removing one problem creates other problems in this solution. The first problem is that the users needs to make a selection when entering the site. We have the option to save their choice with a cookie, but it will still be an annoyance for first time visitors. Also, users may have cookies disabled or move between devices, and suddenly they are browsing on their mobile phone, a popup window here would be even more of a nuisance.

The priority tagging must also be maintained. It can be troublesome even from the start to decide what the users needs in an exact way. The more you add and customize to your website, the harder it will be to choose the right content priority for each user group. If we do mistakes when deciding what is important or not for the different user groups. E.g a journalists is interested in content we have set a low priority number on, they may feel that its hard to find the content they need on the website. Still, all the content should always be accessible from menu items for all user groups. It might just be less accessible.

### **9.2.2 Alternative class schedule**

In our design we remove the class schedule when the size comes down to the mobile version. This is because a full scale class schedule will be unreadable at such a small screen size. A valid possibility is to display the schedule, only viewing a single day. This is a good trade-off since we still will be able to see the class schedule. To be able to have consistency on what information we display we want to still be able to show the schedule and not only a link sending the user to another page. This variant might sort the problem, and give the users

an optimal solution.

## 9.3 Making a responsive website

### 9.3.1 The process of making the site

When creating a responsive website, it is important to keep in mind that the users is accessing the site from different devices and browsers. This might be true for all websites, but is especially important for a responsive website. It is important to test your website in as many types as possible. There exist css files that are created to eliminate inconsistencies, and render elements according to new standards, an example of this is `normalize.css`<sup>2</sup> that takes care of many known bugs. The issues that are still present, need to be found by conducting tests on the different screen sizes. Additionally, the website might look different on an actual device than on a computer browser with the same width as the device. Therefore, testing it on iOS and Android, on both the phone and tablet size is really useful.

There is also some tools available to make the responsive website creation easier. While testing your site in the browser there is a need to always know the screen size width your browser is set to. You can get this functionality from the developer tools available in any browser. Still, there are extensions or add-ons in most modern browsers that gives you this functionality. Also including additional functions, such as typing in a screen width to set your browser to this width. This is useful for testing exact screen widths that match the width of a certain device. At the chrome extension store<sup>3</sup> you can search for "responsive" or "screen width", and you will get several matches for extensions that delivers the same functionality. There is also websites that offers to display your website in different resolutions side by side. One of these can be found at [36].

Considering the content of the website, we think it is important to try to figure out which functions that are most important on the different devices. This can be very different from website to website, and in some cases the content could have the same prioritisation on all devices. But in most cases, some content might be of a higher priority on one platform than the other. Consider a Chinese restaurant's website, an example taken from two web designers; Gavin Hoffman and Robert Bavington[37]. Say the website contain a menu, some reviews and a location and contact section. When you are on your home computer, the chances are that you have some more time on your hands than

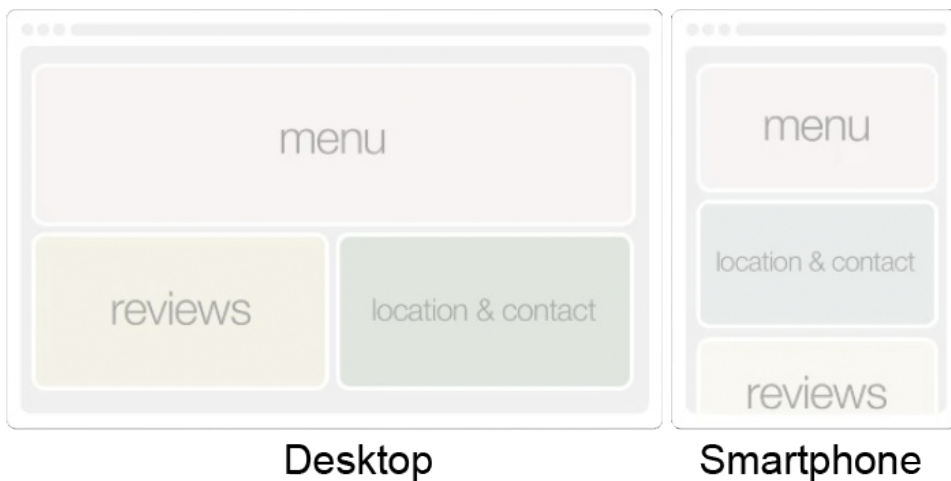
---

<sup>2</sup><http://necolas.github.io/normalize.css/>

<sup>3</sup><https://chrome.google.com/webstore/category/extensions>



when one your smartphone. You probably want to look at the menu, read a couple of reviews, and then use the contact information to get in touch with the restaurant and order a meal. When you are using your smartphone, you might have less time on your hands. The reviews section might not be as important. The priority of the content could then be that you first have the menu, then the contact information, and lastly the review section, as illustrated in Figure 9.1.

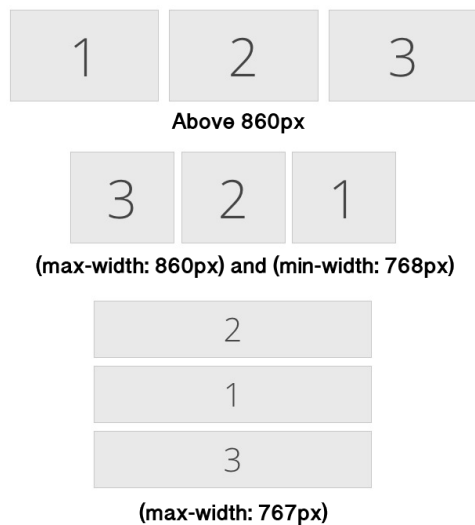


**Figure 9.1:** An example of how the priority can change from desktop to smartphone in a responsive website

With the NTNU website, we did not manage to find any very suitable cases where we could do this priority change. Because of this, the technical part, changing prioritisation on content lost some focus. Although, on the implementation without any frameworks, we did switch the RSS feed, placing it beneath the news and information sections. This is described in Section 8.3.2. The content switch was implemented because we wanted to check out how this would be done technically. On the version where we used a responsive framework, we did not do any priority change. The reason is that we could not find any function in the framework to directly do implement this feature on. The method would be similar to the method used in the version without a framework. We would need to have two elements with the same content, at two different locations, and hide and show them at the wanted breakpoints. You can argue that this would be done easier with Bootstrap, because we have the possibility to hide elements at certain breakpoints directly from the HTML. Although, when using a framework, you want the functionality to be better customized for your needs than this.

Although, after the prototypes were implemented, we made a discovery re-

garding this feature. A framework called Gumby[38], which was not one of the frameworks we evaluated, has an extension which gives us the prioritisation feature. This extension gives us the possibility to set different prioritisation on the elements, based on different screen sizes. This extension is not a part of the core of the framework, but is a feature you manually need to install. It is installed via a package manager named Bower[39] and built with Gumbys own helper scripts called Claymate[40]. The feature is called Shuffle, and from the documentation we consider this to be exactly the feature we need, to do the prioritisation change. Consider Listing 9.1, this is an example of the use of the Shuffle feature. Here we see a row with an added parameter "gumby-shuffle". They specified that within the screen width 768px-860px the columns should have the prioritisation 2-1-0. When the screen is under 767px wide, the prioritization should be 1-0-2. This is 0-indexed. When the screen is not within these ranges, the standard prioritisation will be the applicable, which of course is 0-1-2. We can see the graphical representation of the code in Figure 9.2.



**Figure 9.2:** The Gumby shuffle feature graphically illustrated. The three sections will arrange differently according to different screen sizes.

**Listing 9.1:** An example of the Shuffle feature available as an extension to Gumby

```
1 <div class="row" gumby-shuffle="only screen and (max-width: 860px)
    and (min-width: 768px)|2-1-0,only screen and (max-width: 767px)|1
    -0-2">
2   <div class="four columns">
3     <h1>1</h1>
4   </div>
5   <div class="four columns">
6     <h1>2</h1>
7   </div>
8   <div class="four columns">
9     <h1>3</h1>
10  </div>
11 </div>
```

As we can see, this prioritisation change can only be used within one "row". So, it can not be used in all circumstances. Although, in the Chinese restaurant example illustrated in Figure 9.1, this function could have been used. The menu item resides in one row, and will stay here on all devices. The "Reviews" and "Location and Contact" sections is on the same row. Here we can set a different prioritisation on the elements that changes when we resizes from desktop to smartphone resolution.

### 9.3.2 Suggested development process

From our experience through this thesis, we have managed to come up with some key points to how one should approach the process of developing a responsive website.

This process is general, and not specific for any domain. When creating a responsive website, it is important that the users needs is addressed. Additionally, a key aspect is that the most important features, specific for each device, is discussed and prioritized.

1. Identify your users
2. Identify use cases for each device type
3. Design and sketch the site, focusing on the user needs
4. Develop a prototype
  - (a) Look for a framework that best suits your website
  - (b) Code the website responsively
  - (c) Add media queries to fix glitches or add customization

### 9.3.3 Should a framework be used?

There are many considerations to be made when making a responsive website, first and foremost you have to for look what you need, and the functionality you want to fit in the website. This might lead you to argue whether to make your site with or without the use of a responsive framework. There are several factors that might point you to one direction or the other.

How much time you have to develop is probably one of the most important factors when deciding whether to utilize a framework or not, in our experience, the use of a framework reduced the development time significantly. Coding the HTML took approximately the same time, but when it came to the CSS we had to add all the media queries ourself when coding without a framework. Bootstrap has almost all the CSS you need incorporated in its core, so the only CSS you will code yourself when using Bootstrap is for small tweaks to the design. Setting up the grid system was the most troublesome in the non-framework solution, and it takes a long time to tweak a self made grid to perfection.

You know your code better. If the framework you are using does something that does not fit with your design, it could be troublesome to tweak it to fit your needs. The framework might also provide you with a great deal of things which you do not really need, so you will be including libraries that are unnecessary, which can exacerbate the performance. Still, some of the frameworks is starting to give the possibility to customize your download, which means you can add only the features you need, and customize the look of the different elements.

### 9.3.4 Missing functionality

#### Prioritisation

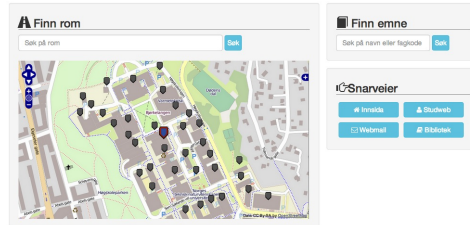
Something we wanted to see in a responsive framework was the possibility to set different priority on content on different resolutions. In the implementation using Bootstrap, this functionality was not available. Although, we later found this functionality in Gumby[38], the only minus here was that it was only for within one row. We could not find this feature available in any other responsive frameworks.

#### Blank space

In Bootstrap, the framework did not take use of the screen space as much as it could, leaving out some blank space. As illustrated in Figure 9.3, we see a lot of empty space under the two boxes on the right side of the find room function.

---

This is actually something the Masonry framework (mentioned in Section 3.2) solves, since it is focused on using up all the space. But even Bootstrap could have done some kind of measures here to fill out the space, like creating the two boxes on the right higher and filling out all the space.



**Figure 9.3:** Blank space in the layout, while using Twitter bootstrap 3.0

### 9.3.5 Which framework should be used?

So if you were to use a responsive framework, which one would you choose? You have to carefully consider whether to use a feature heavy responsive framework like Bootstrap, a stripped down responsive framework like Skeleton, or simply no responsive framework at all. This of course depends on your implementation and your specific needs, but we will try to sum up what we saw as the best solutions.

If you want to customize and have as much power as possible though, we would recommend using a minimal framework which only gives you a fluid grid. A good example of this type is Skeleton. From our experience of creating a website with a premade grid, and one without any framework, we would recommend using a framework. You will still have the power to change the things you want, the grid just makes your job a lot easier by handling all the responsive features.

The choice is very dependent on your needs, and your website. If you are looking to use as little time as possible on designing the website, maybe a Twitter Bootstrap[13] website with a custom theme would be the best solution. If you are creating a website for a restaurant and wish to be able to change content prioritisation, then Gumby[38] is the optimal solution. Similar, if you are creating a site where you are presenting many elements and wish to leave as less blank space as possible, perhaps Masonry[18] is the responsive framework that best suits you.

---

---

# Bibliography

- [1] Ethan Marcotte. Responsive web design. <http://alistapart.com/article/responsive-web-design>, 2010. [Online; accessed 24.09.2013].
- [2] Briony J Oates. *Researching information Systems and Computing*. Sage Publications Ltd, 2006.
- [3] Todd Zaki Warfel. *Prototyping*. Rosenfeld Media, 2009.
- [4] User-Centered Design. [http://www.usabilityprofessionals.org/usability\\_resources/about\\_usability/what\\_is\\_ucd.html](http://www.usabilityprofessionals.org/usability_resources/about_usability/what_is_ucd.html)). [Online; accessed 15-april-2013].
- [5] Ethan Marcotte. On being "responsive". <http://unstoppablerobotninja.com/entry/on-being-responsive/>, 2013. [Online; accessed 13-February-2013].
- [6] *Oxford Dictionary of English*. Oxford University Press, 2010.
- [7] Josh Byers. Mobile responsive design: The flexible grid. <http://www.studiopress.com/design/flexible-grid.htm/>, 2011. [Online; accessed 23.09.2013].
- [8] W3Schools. Css. [http://www.w3schools.com/css/css\\_font.asp](http://www.w3schools.com/css/css_font.asp), 2013. [Online; accessed 16-September-2013].
- [9] Håkon W Lie. Cascading html style sheets – a proposal. <http://www.w3.org/People/howcome/p/cascade.html/>, 1994. [Online; accessed 06.03.2013].
- [10] Håkon W Lie. and Tantek Çelik. Media queries, W3C Working Draft. <http://www.w3.org/TR/2001/WD-css3-mediaqueries-20010404/>, 2001. [Online; accessed 06-Mars-2013].

- 
- [11] Håkon W Lie., Tantek Çelik., Daniel Glazman., and Anne van Kesteren. Media Queries W3C Recommendation. <http://www.w3.org/TR/css3-mediaqueries/>, 2012. [Online; accessed 06-Mars-2013].
- [12] Ben Frain. *Responsive Web Design with HTML5 and CSS3*. Packt Publishing Ltd., 35 Livery Street Birmingham B3 2PB, UK, 2012.
- [13] Twitter Bootstrap Framework. <http://getbootstrap.com/>, 2013. [Online; accessed 12-March-2013].
- [14] GitHub. Repository toplist. <https://github.com/search?q=stars%3A%3E1&s=stars&type=Repositories>, 2013. [Online; accessed 27-November-2013. twbs/bootstrap as the most starred repository].
- [15] Foundation Framework. <http://foundation.zurb.com/>, 2013. [Online; accessed 12-March-2013].
- [16] unsemantic Framework. <http://unsemantic.com/>, 2013. [Online; accessed 12-March-2013].
- [17] Skeleton Framework. <http://www.getskeleton.com/>, 2013. [Online; accessed 12-March-2013].
- [18] Masonry Framework. <http://masonry.desandro.com/>, 2013. [Online; accessed 12-March-2013].
- [19] Luke Wroblewski. *Mobile First*. A Book Apart, 2011.
- [20] Luke Wroblewski. Mobile First Presentation. <http://www.lukew.com/presos/preso.asp?26>, 2012. [Online; accessed 07-Mars-2013].
- [21] TNS gallup. TNS listene. [http://www.tnslistene.no/?list\\_id=1&list\\_type=1&week=39&year=2013&report=day&metric=uv](http://www.tnslistene.no/?list_id=1&list_type=1&week=39&year=2013&report=day&metric=uv), 2013. [Online; accessed 19.09.2013].
- [22] Erik Runyon. Highered rwd directory. <http://weedygarden.net/highered-rwd-directory/>, 2013. Online; accessed 23-Sept-2013.
- [23] G. McGovern. *The Stranger's Long Neck: How to Deliver What Your Customers Really Want Online*. A&C Black, 2010.
- [24] Bjørg Danielsen. Given data statistics from ntnu-info, 2013.
- [25] Direktoratet for forvaltning og IKT. Kvalitet på nett - norges teknisk-naturvitenskapelige universitet. <http://kvalitet.difi.no/resultat/resultat-detalj/?tailid=26759&aar=2011/>, 2011. Online; accessed 19-Sept-2013.
- [26] IEEE Xplore Digital Library. <http://ieeexplore.ieee.org/>). [Online; accessed 05-March-2013].



- 
- [27] Springer Link. <http://link.springer.com/>). [Online; accessed 05-March-2013].
- [28] Irving L Janis. *Groupthink : psychological studies of policy decisions and fiascoes*. Boston : Houghton Mifflin., 1982.
- [29] Medienorge. Andel som har smarttelefon i norge. <http://www.medienorge.uib.no/?cat=statistikk&medium=it&queryID=379&aspekt=oppdatering>, 2013. Online; accessed 20-Sept-2013.
- [30] NTNU. Academic Calendar. <http://www.ntnu.no/studier/akalender/>, 2013. [Online; accessed 19.04.2013].
- [31] Screen size problem. <http://opensignal.com/reports/fragmentation.php>. [Online; accessed 05-May-2013].
- [32] Jacob Nielsen. How Little Do Users Read? <http://www.nngroup.com/articles/how-little-do-users-read/>, 2008. [Online; accessed 07-April-2013].
- [33] Jacob Nielsen. Tunnel Vision and Selective Attention. <http://www.nngroup.com/articles/tunnel-vision-and-selective-attention/>, 2012. [Online; accessed 07-April-2013].
- [34] Gestalt Theory. <http://graphicdesign.spokanefalls.edu/tutorials/process/gestaltprinciples/gestaltprinc.htm>. [Online; accessed 07-April-2013].
- [35] Ethan Marcotte. *Rwsponsive Web Design*. Jeffrey Zeldman, 2011.
- [36] Matt Kersley. Responsive web design testing tool. <http://mattkersley.com/responsive/>. [Online; accessed 05.11.2013].
- [37] Gavin Hoffman and Robert Bavington. Prioritising content with a responsive website. <http://www.create-design.co.uk/blog/videos/prioritising-content-with-a-responsive-website.html>, 2013. Online; accessed 13-March-2013.
- [38] Gumby Framework. <http://gumbyframework.com/>, 2013. [Online; accessed 28-November-2013].
- [39] Bower A package manager for the web. <http://bower.io/>, 2013. [Online; accessed 28-November-2013].
- [40] Gumby. Claymate. <https://github.com/GumbyFramework/Claymate>, 2013. [Online; accessed 28-November-2013].

---

---

# Appendices



---

---

---

## Interview with Jan Erik Kaarø, NTNU info

1. What was the background for doing the redesign?

*For a long time we wanted to go through both the design and stucture of NTNUs external website, but we havent had the capacity to do something about it until now. The main problem with ntnu.no was that it was overcrowded and too little focused. The main target group is potential applicants for ntnu, and the redesign tries to point this out better than before.*

2. How long has the redesign process been going on?

*Ca three months*

3. How long ago was the decision to redesign decided?

*The sctucture on NTNUs external web was desiced in a project that went on in 2004/2005, but the design and content of the central parts has been changed many times since.I reccomend you to take a look at [http://web.archive.org/web/\\*/http://www.ntnu.no](http://web.archive.org/web/*/http://www.ntnu.no) to follow the development of the main site.*

4. We noticed that it was only the front page that got a redesign in this update. Are there any plans of redoing the other parts of ntnu.no as well?

*We are relatively soon going to change <https://www.ntnu.no/studier> (student portal) og [www.ntnu.edu](http://www.ntnu.edu) (english main site) and some more sub-pages, but from there we need to take a timeout due to capacity issues. The next big planned change on the external web is a full review of the research pages under [www.ntnu.no/forskning](http://www.ntnu.no/forskning), but this can happen earliest in the fall off 2013. We have in the last three years focused on launching*

---

a ny publishing system (Liferay) and a new intranet (Innsida 2.0). This is the reason that we had to wait on further developement of the external web.

5. Are you considering making the website responsiv, or maybe make a mobile and/or tablet version of the website?

*Resonsiv design is absolutely on our wishlist for further developement of both the external web and intranet (Innsida 2.0). WE are now using Liferay as a publishing system on both sites. But im afraid there are many other parts that are in front of responsive in the queue. Again we have to prioritize very hard on what we use time and money on.*

6. Have you considered developing a mobile application to offer the students the content they need?

*This was also raised as a question from student representative Michael Johansen in NTNUs last board meeting 24.april 2013. He got an answer from organizational director Ida Munkeby (Who also is the top administrative leader for the IT-area on NTNU) that an ongoing strategy for mobile services at NTNU is currently being worked on. (vedlegg fra mailen her)*

7. NTNUs IT-systems have had a reputation of being outdated ( ) How do you interpret the feedback, and what are the further plans to give the students the content they need?

*We are fully aware that we have potential for improvement on many areas. The biggest problem is probably that our systems is pourly connected and ( ) are not very user friendly, but we still hope that the students see some good parts as well. The plan is that the new Innsida in the future is going to be the main entrance to the most important systems that students have to deal with. The project "Gruppevare for studenter" that is going to improve important shared services as email and file storage. Contact Arne Fjerdrumsmoen in the IT-department if you want more info about this project. At the same time external evaluations as DIFIs review of norwegian, public websites concludes that ntnu.no isnt as bad as many people think. In the last evaluation ntnu.no got 5 out of 6 stars[25].*



# Appendix **B**

## Questionnaire

1. Hvilken brukergruppe tilhører du?

	Response Total	Response Percent	Points	Avg
Nåværende NTNU student	81	79%	n/a	n/a
Mulig fremtidig NTNU student	1	1%	n/a	n/a
NTNU Ansatt / Professor	17	17%	n/a	n/a
Tidligere NTNU student	4	4%	n/a	n/a
Presse/Journalist	0	0%	n/a	n/a
Annet, vennligst spesifiser	0	0%	n/a	n/a
<b>Total Respondents (For this Question)</b>	<b>103</b>	<b>100%</b>		

**Figure B.1:** Questionnaire, question 1

2. Hvis du nå er NTNU student, hvilket klassetrinn er du på?

	Response Total	Response Percent	Points	Avg
1. klasse	17	21%	n/a	n/a
2. klasse	9	11%	n/a	n/a
3. klasse	12	15%	n/a	n/a
4. klasse	17	21%	n/a	n/a
5. klasse (og oppover)	27	33%	n/a	n/a
<b>Total Respondents (For this Question)</b>	<b>82</b>	<b>100%</b>		

**Figure B.2:** Questionnaire, question 2

3. Hvor går du som regel for å hente informasjon relatert til NTNU?

	Response Total	Response Percent	Points	Avg
Går inn på NTNU.NO	78	76%	n/a	n/a
Googler	53	51%	n/a	n/a
Går til studentservice og spør	6	6%	n/a	n/a
Sender epost til studieadministrasjon	10	10%	n/a	n/a
Spør en venn	24	23%	n/a	n/a
Annet, vennligst spesifiser	10	10%	n/a	n/a
<a href="#">view</a>				
<b>Total Respondents (For this Question)</b>				<b>103</b>

**Figure B.3:** Questionnaire, question 3

4. Tilfredsstiller NTNU.NO dine behov for informasjon relatert til NTNU?

	Response Total	Response Percent	Points	Avg
Ja	69	68%	n/a	n/a
Nei	33	32%	n/a	n/a
<b>Total Respondents (For this Question)</b>				<b>102</b>

**Figure B.4:** Questionnaire, question 4

5. Hvilke undersider/informasjon er du ute etter når du besøker NTNU.NO?

	Response Total	Response Percent	Points	Avg
Studier	73	71%	n/a	n/a
NTNU universitetsbiblioteket	42	41%	n/a	n/a
Its Learning	55	53%	n/a	n/a
Sivilingeniør og arkitekt	30	29%	n/a	n/a
Studweb	39	38%	n/a	n/a
Om NTNU	20	19%	n/a	n/a
Student i Trondheim	9	9%	n/a	n/a
Emner	73	71%	n/a	n/a
Studier	43	42%	n/a	n/a
Finn rom	56	54%	n/a	n/a
Finn en ekspert	6	6%	n/a	n/a
Innsida	51	50%	n/a	n/a
Annet, vennligst spesifiser	15	15%	n/a	n/a
<a href="#">view</a>				
<b>Total Respondents (For this Question)</b>				<b>103</b>

**Figure B.5:** Questionnaire, question 5

6. Hvilke undersider/informasjon er du ute etter når du besøker NTNU.NO via PC/MAC?

	Response Total	Response Percent	Points	Avg
Studier	68	66%	n/a	n/a
NTNU universitetsbiblioteket	43	42%	n/a	n/a
Its Learning	60	58%	n/a	n/a
Sivilingeniør og arkitekt	26	25%	n/a	n/a
Studweb	41	40%	n/a	n/a
Om NTNU	24	23%	n/a	n/a
Student i Trondheim	6	6%	n/a	n/a
Emner	76	74%	n/a	n/a
Studier	48	47%	n/a	n/a
Finn rom	53	51%	n/a	n/a
Finn en ekspert	7	7%	n/a	n/a
Innsida	46	45%	n/a	n/a
Annet, vennligst spesifiser	15	15%	n/a	n/a
<a href="#">view</a>				
<b>Total Respondents (For this Question)</b>				<b>103</b>

**Figure B.6:** Questionnaire, question 6

7. Hvilke undersider/informasjon er du ute etter når du besøker NTNU.NO via TABLET?

	Response Total	Response Percent	Points	Avg
Studier	25	24%	n/a	n/a
NTNU universitetsbiblioteket	12	12%	n/a	n/a
Its Learning	26	25%	n/a	n/a
Sivilingeniør og arkitekt	8	8%	n/a	n/a
Studweb	13	13%	n/a	n/a
Om NTNU	9	9%	n/a	n/a
Student i Trondheim	2	2%	n/a	n/a
Emner	27	26%	n/a	n/a
Studier	14	14%	n/a	n/a
Finn rom	21	20%	n/a	n/a
Finn en ekspert	2	2%	n/a	n/a
Innsida	21	20%	n/a	n/a
Annet, vennligst spesifiser	46	45%	n/a	n/a
<a href="#">view</a>				
<b>Total Respondents (For this Question)</b>				<b>103</b>

**Figure B.7:** Questionnaire, question 7

8. Hvilke undersider/informasjon er du ute etter når du besøker NTNU.NO via SMARTPHONE?

	Response Total	Response Percent	Points	Avg
Studier	24	23%	n/a	n/a
NTNU universitetsbiblioteket	9	9%	n/a	n/a
Its Learning	28	27%	n/a	n/a
Sivilingeniør og arkitekt	6	6%	n/a	n/a
Studweb	17	17%	n/a	n/a
Om NTNU	10	10%	n/a	n/a
Student i Trondheim	4	4%	n/a	n/a
Emner	26	25%	n/a	n/a
Studier	16	16%	n/a	n/a
Finn rom	29	28%	n/a	n/a
Finn en ekspert	3	3%	n/a	n/a
Innsida	17	17%	n/a	n/a
Annet, vennligst spesifiser	36	35%	n/a	n/a
<a href="#">view</a>				
<b>Total Respondents (For this Question)</b>				<b>103</b>

**Figure B.8:** Questionnaire, question 8

9. Hvilken enhet er du som oftest innom NTNU.NO via?

	Response Total	Response Percent	Points	Avg
PC/MAC	99	96%	n/a	n/a
Smartphone	3	3%	n/a	n/a
Tablet	1	1%	n/a	n/a
<b>Total Respondents (For this Question)</b>				<b>103</b>
		<b>100%</b>		

**Figure B.9:** Questionnaire, question 9



# Implementation without frameworks

## C.1 index.html

```
1 <html>
2   <head>
3
4     <link href="css/index.css" rel="stylesheet">
5     <link rel="stylesheet" href="ikoner/font-awesome/css/
      font-awesome.min.css">
6
7     <meta charset="utf-8">
8     <meta name="viewport" content="width=device-width" />
9     <script src="js/jquery-1.10.2.min.js"></script>
10    <link href='http://fonts.googleapis.com/css?family=Open+Sans:400
      ,300' rel='stylesheet' type='text/css'>
11
12    <title> ntnu.no </title>
13  </head>
14
15  <body onload="bodyLoad()">
16    <div id="mainWrapper">
17      <div id="header">
18        <center></center>
19      </div>
20
21      <div id="menu">
```

```

22     <div id="menu-links">
23         <a href="#">Studier</a>
24         <a href="#">Sivilingeniør og arkitekt</a>
25         <a href="#">Om NTNU</a>
26     </div>
27     <div id="menu-icon"></div>
29
30     <div id="search-field"> <input type="text" placeholder="Søk
31         etter innhold"></input><button class="search-button"><i
32             class="icon-search search-button"> Søk</i></button></div>
33     <div style="clear: both;"></div>
34 </div>
35
36 <div id="findRoom" class="box-container">
37     <div class="sectionHeader">
38         <i class="icon-map-marker section-title"></i> Finn rom
39     </div>
40     <span><input placeholder="Søk etter rom"></input><button class=
41         "search-button"><i class="icon-search search-button"> Søk</
42         i></button></span>
43 </br>
44 
45 </div>
46
47 <div id="topRightWrapper">
48
49     <div id="findCourse" class="toprightBoxes">
50         <div class="sectionHeader">
51             <i class="icon-beaker section-title"></i> Finn emne
52         </div>
53         <span><input placeholder="Søk etter emne"></input><button
54             class="search-button"><i class="icon-search search-button">
55             Søk</i></button></span>
56     </div>
57
58     <div id="shortcuts" class="toprightBoxes">
59         <div class="sectionHeader">
60             <i class="icon-external-link section-title"></i> Snarveier
61         </div>
62         <div class="shortcutsButtons">
63             <button> <i class="icon-envelope"></i> Webmail </button>
64             <button> <i class="icon-globe"></i> Studweb </button>

```

```

62     <button> <i class="icon-book"></i> Bibliotek </button>
63     <button><i class="icon-home"></i> Innsida </button>
64     <button><i class="icon-cog"></i> Its learning </button>
65
66     </div>
67 </div>
68
69
70 </div>
71
72 <div id="schedule" class="box-container">
73     <div class="sectionHeader">
74         <i class="icon-calendar section-title"></i> Timeplan
75     </div>
76     </img>
77 </div>
78
79 <div id="rss-large-screen" class="box-container rss">
80     <div class="sectionHeader">
81         <i class="icon-rss section-title"></i> RSS
82     </div>
83     <table>
84         <tr>
85             <td><i class="icon-circle"></i>
86             </td>
87             <td>informationsmelding fra RSS feed her. informationsmelding
88                 fra RSS feed her.informationsmelding fra RSS feed
89                 her.informationsmelding fra RSS feed her. </td>
90         </tr>
91         <tr>
92             <td><i class="icon-circle"></i></td>
93             <td>informationsmelding fra RSS feed her. informationsmelding
94                 fra RSS feed her.informationsmelding fra RSS feed
95                 her.informationsmelding fra RSS feed her. </td>
96         </tr>
97         <tr>
98             <td><i class="icon-circle"></i></td>
99             <td>informationsmelding fra RSS feed her. informationsmelding
100                fra RSS feed her.informationsmelding fra RSS feed
                her.informationsmelding fra RSS feed her. </td>

```

```

101     </table>
102
103 </div>
104
105 <div id="news" class="box-container">
106     <div class="sectionHeader">
107         <i class="icon-columns section-title"></i> Nyheter
108     </div>
109
110     <table>
111         <tr>
112             <td></i>
113             </td>
114             <td>NTNU er i finalerunden om å få sentre for fremragende
115                 undervisning</td>
116         </tr>
117         <tr>
118             <td>
119                 
120             </td>
121             <td> Se video-opptak fra immatrikuleringen tirsdag 13.
122                 August</td>
123         </tr>
124         <tr>
125             <td>
126                 
127             </td>
128             <td>
129                 7000 studenter inntar NTNU
130             </td>
131         </tr>
132     </table>
133 </div>
134
135 <div id="information" class="box-container">
136     <div class="sectionHeader">
137         <i class="icon-info section-title"></i> Informasjon
138     </div>
139
140     <div class="informationButtons">
141         <button> <i class="icon-calendar"></i>
142             Timeplan </button>
143         <button> <i class="icon-comments"></i> Klage på karakter </
144             button>
145         <button> <i class="icon-info"></i> EksamenInfo </button>
146         <button> <i class="icon-group"></i>Søk etter studenter </
147             button>

```



```
144     <button> <i class="icon-group"></i> Søk etter ansatte </  
145         button>  
146     <button> <i class="icon-laptop"></i> IT-hjelp </button>  
147  
148     </div>  
149 </div>  
150  
151 <div id="rss-small-screen" class="box-container rss">  
152  
153 </div>  
154  
155  
156 <div id="kontakt" class="box-container">  
157     <div class="sectionHeader">  
158         <i class="icon-phone section-title"></i> Kontakt  
159     </div>  
160  
161     <div class="section-seperator">  
162         <p>  
163             <b> Spørsmål om studier </b></br>  
164             <a href="mailto:studentservice@adm.ntnu.no">  
165                 studentservice@adm.ntnu.no  
166             </a>  
167         </p>  
168  
169         <p>  
170             Samfunnsfag og humanistiske fag  
171             Tlf Dragvoll: <span>73 59 67 00</span>  
172         </p>  
173  
174         <p>  
175             Realfag, ingeniørfag, medisin  
176             Tlf: Gløshaugen: <span>73 59 52 00</span>  
177         </p>  
178     </div>  
179     <div class="section-seperator">  
180         <p>  
181             <b> Spørsmål om IT</b></br>  
182             Epost, brukernavn/passord, programvare:</br>  
183             <a href="mailto:orakel@ntnu.no">  
184                 orakel@ntnu.no  
185             </a>  
186         </p>  
187     </div>  
188     <p>  
189         <b> Web </b>
```

```

190     <a href="mailto:web@info.ntnu.no">
191         web@info.ntnu.no
192     </a>
193 </p>
194 </div>
195 <script>
196
197 $(window).resize(function(){
198     resizeDiv();
199 });
200
201 function bodyLoad(){
202     $('#rss-small-screen').html($('#rss-large-screen').html());
203     resizeDiv();
204 }
205
206 function resizeDiv(){
207     if($(window).width()>1050)
208     {
209         $('#rss-large-screen').height($('#schedule').height());
210         $('#information').height($('#news').height());
211     }
212     if($(window).width()>767)
213     {
214         $('#topRightWrapper').height($('#findRoom').height());
215     }
216     }else if($(window).width()>480){
217         $('#topRightWrapper').height("auto");
218         var height = $('#findCourse').height() + $('#shortcuts').
219             height();
220         $('#news').height(height/2);
221         $('#information').height(height/2);
222     }
223
224     if($(window).width()<1050 && $(window).width()>691){
225         $('#information').height($('#news').height());
226     }
227 }
228 </script>
229 </div>
230 </body>
231 </html>

```

---

## C.2 index.css

```
1 body{
2   font-family: 'Open Sans', sans-serif;
3   font-size: 100%;
4   height: 100%;
5   background-color: #FAFDFF
6   font-size: small;
7 }
8 table{
9   font-family: 'Open Sans', sans-serif;
10  font-size: 100%;
11  font-size: small;
12 }
13
14 button {
15   -moz-box-shadow:inset 0px 1px 0px 0px #97c4fe;
16   -webkit-box-shadow:inset 0px 1px 0px 0px #97c4fe;
17   box-shadow:inset 0px 1px 0px 0px #97c4fe;
18   background:-webkit-gradient( linear, left top, left bottom,
19     color-stop(0.05, #3d94f6), color-stop(1, #1e62d0) );
20   background:-moz-linear-gradient( center top, #3d94f6 5%, #1e62d0
21     100% );
22   filter:progid:DXImageTransform.Microsoft.gradient(startColorstr='#3
23     d94f6', endColorstr='#1e62d0');
24   background-color:#3d94f6;
25   -webkit-border-top-left-radius:20px;
26   -moz-border-radius-topleft:20px;
27   border-top-left-radius:20px;
28   -webkit-border-top-right-radius:20px;
29   -moz-border-radius-topright:20px;
30   border-top-right-radius:20px;
31   -webkit-border-bottom-right-radius:20px;
32   -moz-border-radius-bottomright:20px;
33   border-bottom-right-radius:20px;
34   -webkit-border-bottom-left-radius:20px;
35   -moz-border-radius-bottomleft:20px;
36   border-bottom-left-radius:20px;
37   text-indent:0;
38   border:1px solid #337fed;
39   display:inline-block;
40   color:#ffffff;
41   font-weight:bold;
42   font-style:normal;
43   text-decoration:none;
44   text-align:center;
45   text-shadow:1px 1px 0px #1570cd;
```

---

```
43 }
44
45 center img{
46     max-height: 100px;
47 }
48
49 .sectionHeader{
50     text: nowrap;
51     font-family: 'Open Sans', sans-serif;
52     font-size: 150%;
53 }
54
55 #header{
56     width: 99%;
57     background-color: white;
58     border-radius: 5px;
59     border: 1px gray solid;
60     margin: 0 auto;
61 }
62 #menu{
63     width: 89%;
64     border-radius: 5px;
65     border: 1px gray solid;
66     margin: 0 auto;
67     margin-top: 5px;
68     padding: 5px;
69     padding-left: 5%;
70     padding-right: 5%;
71     padding-top: 1%;
72     padding-bottom: 1%;
73     text-align: center;
74     white-space: nowrap;
75
76 }
77 #menu-links > *{
78     margin-left: 5%;
79     margin-right: 5%;
80     padding-top: 5px;
81     float: left;
82 }
83 .search-button{
84     border-radius: 2px;
85     height: 19px;
86     margin-bottom: 10px;
87 }
88 .search-button i{
89     font-size: 1em;
```

---

```
90 }
91 input{
92     padding: 2.5px;
93 }
94 #menu a:link {text-decoration:none; color:#1A4C80;}
95 #menu a:hover {text-decoration:none; border-top: 2px solid;}
96 #menu a:visited {text-decoration:none; color: #1A4C80;}
97
98 #menu-links{
99     text-align: left;
100 }
101
102 #search-field{
103     display: inline-block;
104     text-align: right;
105     float: right;
106 }
107
108 #menu-icon{
109     display: none;
110 }
111
112 .toprightBoxes{
113     border-radius: 5px;
114     background-color: #F0F0F5;
115 }
116
117 #findRoom{
118     float: left;
119     width: 48%;
120     margin-bottom: 5px;
121 }
122
123 #topRightWrapper{
124     float: right;
125     width: 48%;
126     overflow: hidden;
127     margin: 5px;
128     border-radius: 5px;
129     margin-bottom: 5px;
130 }
131
132 #shortcuts{
133     height: auto;
134     float: right;
135     width: 100%;
136     margin-bottom: 5px;
```

---

```
137     height: 80%;
138 }
139
140 #findCourse{
141     width: 100%;
142     float: right;
143     margin-bottom: 5px;
144     height: 20%;
145 }
146
147 #schedule{
148     width: 48%;
149     float: left;
150     clear: both;
151 }
152
153 #rss-large-screen{
154     float: right;
155     width: 48%;
156     overflow: scroll;
157 }
158
159 table{
160     font-size: small;
161 }
162 #rss-small-screen{
163     float: left;
164     width: 100%;
165     margin-right: -1px;
166     display: none;
167 }
168 #news{
169     width: 48%;
170     float: left;
171     overflow: scroll;
172 }
173
174 #information{
175     float: right;
176     width: 48%;
177 }
178 #kontakt{
179     float: left;
180     width: 98.8%;
181 }
182
183 .box-container{
```

---

```
184 border-radius: 5px;
185 margin: 5px;
186 background-color: #F0F0F5;
187 overflow: hidden;
188 }
189
190 img{
191   max-width: 100%;
192 }
193 span{
194   white-space: nowrap;
195 }
196
197 button:hover {
198   background:-webkit-gradient( linear, left top, left bottom,
199     color-stop(0.05, #1e62d0), color-stop(1, #3d94f6) );
200   background:-moz-linear-gradient( center top, #1e62d0 5%, #3d94f6
201     100% );
202   filter:progid:DXImageTransform.Microsoft.gradient(startColorstr='#1
203     e62d0', endColorstr='#3d94f6');
204   background-color:#1e62d0;}
205 }
206
207 button:active {
208   position:relative;
209   top:1px;
210 }
211
212 #news img{
213   width: 100%;
214 }
215
216 #news td:first-of-type{
217   width: 20%;
218 }
219
220 .rss-large-screen td:first-of-type{
221   min-width: 35px;
222 }
223
224 .search-button{
225 }
226
227 #shortcuts button{
228   width: 48%;
229   margin-bottom: 10px;
230   margin-right: 5px;
231   height: 40px;
232 }
```

---

```
228 #information button{
229     width: 30%;
230     height: 70px;
231     float: left;
232     margin-bottom: 10px;
233     margin-right: 5px;
234
235 }
236
237 .shortcutsButtons{
238
239 }
240 .informationButtons, .shortcutsButtons{
241     width: 90%;
242     padding-left: 5%;
243     padding-top: 30px;
244     padding-bottom: 20px;
245     text-align: center;
246     padding-top: 10px;
247     margin: 0 auto;
248     text-align: center;
249 }
250 i{
251     padding-top: 10px;
252     margin-left: 10px;
253     /* font-size: 1.5em;
254     */
255
256
257
258 .section-seperator{
259     border-bottom: 1px gray solid;
260 }
261
262
263 /*
264 Media Queries her og nedover!
265 */
266 @media screen and (min-width: 1200px){
267
268     #mainWrapper{
269         width: 80%;
270         margin: 0 auto;
271     }
272 }
273
274 @media screen and (min-width: 768px) and (max-width: 928){
```



---

```
275
276 #shortcuts button{
277     width: 40%;
278 }
279
280 .shortcutsButtons{
281     padding-top: 20px;
282 }
283 }
284
285 @media screen and (min-width: 768px) and (max-width: 1050px) {
286     body {
287         font-size: 80%;
288     }
289     table{
290         font-size: 80%;
291     }
292 }
293
294 #schedule{
295     width: 98.8%;
296     overflow: hidden;
297 }
298 /*#topRightWrapper{
299     height: 24%;
300 }*/
301 #rss-large-screen{
302     width: 98.8%;
303 }
304 }
305
306 /*
307 Liten tablet og telefon
308 */
309 @media screen and (max-width: 767px) {
310     #rss-large-screen{
311         display: none;
312     }
313     #rss-small-screen{
314         display: inline;
315     }
316 }
317
318 /*
319 Small TABLET
320 */
321 @media screen and (min-width: 481px) and (max-width: 767px) {
```

---

```
322 #menu-icon{
323     display: block;
324 }
325 #menu-links{
326     display: none;
327 }
328 #search-field{
329     margin-top: 10px;
330 }
331 #schedule{
332     display: none;
333 }
334 #findRoom{
335     width: 100%;
336     height: auto;
337 }
338 #findCourse{
339     height: 60px;
340 }
341 #shortcuts{
342     height: auto;
343 }
344
345 #shortcuts button{
346     height: 60px;
347 }
348 .shortcutsButtons{
349     height: auto;
350 }
351
352 #topRightWrapper{
353     float: left;
354     height: auto;
355 }
356 #news{
357     float: right;
358 }
359 }
360
361
362
363
364 /*
365 TELEFON
366 */
367 @media screen and (max-width: 480px){
368     body {
```

---

```
369     margin-left: 0px;
370     margin-right: 0px;
371 }
372
373 #menu-icon{
374     display: block;
375 }
376 #menu-links{
377     display: none;
378 }
379 #search-field{
380     margin-top: 10px;
381 }
382 .box-container{
383     width: 100%!important;
384     margin-left: 0;
385     margin-right: 0;
386     margin-top: 0;
387     margin-bottom: 5px;
388 }
389
390 #findCourse{
391     width: 40%;
392     float: left;
393 }
394 #schedule{
395     display: none;
396 }
397 #findRoom img{
398     display: none;
399 }
400 #findRoom{
401     height: auto;
402     float: left;
403 }
404 #findCourse{
405     height: auto;
406 }
407 #shortcuts{
408     height: auto;
409 }
410 #shortcuts button{
411     width: 47%;
412 }
413 #findRoom span, #findCourse span{
414     float: left;
415     margin-top: 10px;
```

---

```
416     margin-right: 2px;
417 }
418 }
419 #news{
420     display: none;
421 }
422 #topRightWrapper{
423     width: 100%;
424     height: auto;
425     margin: 0;
426     padding: 0;
427 }
428 #kontakt p{
429     font-size: 12px;
430     padding-left: 20px;
431 }
432 #menu{
433     width: 94%;
434     padding-right: 0;
435 }
436 #findcourseMobile span{
437     float: left;
438     margin-top: 10px;
439     margin-right: 2px;
440 }
441 #findCourse input, #findRoom input{
442 }
443
444 #search-field{
445     padding-right: 0;
446 }
447
448 .shortcutsButtons{
449     padding-bottom: 10px;
450     padding-top: 20px;
451     height: auto;
452 }
453 #shortcuts button, #information button{
454     font-size: 1em;
455     height: 55px;
456     margin: 2px;
457     width: 46%;
458     font-size: 0.8em;
459 }
460 .informationButtons{
461     margin: 0 auto;
462 }
```

---

```
463 | #information{
464 |     padding-bottom: 10px;
465 | }
466 | #findRoom, #findcourse{
467 |     width: 100%!important ;
468 | }
469 | }
```



# Appendix D

## Implementation with framework

### D.1 index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title> NTNU - Det skapende universitet </title>
5
6   <meta charset="utf-8">
7   <meta name="viewport" content="width=device-width,
8     initial-scale=1.0">
9   <link rel="stylesheet" href="css/font-awesome/css/
10     font-awesome.min.css">
11   <link href="css/iconic-stroke/iconic_stroke.css" rel="stylesheet"
12     type="text/css" media="screen">
13   <link rel="stylesheet" type="text/css" href="css/bootstrap.css">
14   <link href="css/index.css" rel="stylesheet">
15
16   <script type="text/javascript" src="js/jquery-2.0.3.min.js"></
17     script>
18   <script type="text/javascript" src="js/bootstrap.js"></script>
19 </head>
20 <body>
```

```

20 <div class="container">
21   <div class="row">
22     <div class="col-lg-2 col-md-2 col-sm-2 col-xs-2">
23       <div class="visible-xs">
24         <button type="button" class="btn btn-default btn-lg" style=
           "margin-top:20px;">
25           <span class="glyphicon glyphicon-align-justify" ></span>
26         </button>
27       </div>
28     </div>
29     <div class="col-lg-10 col-md-10 col-md-offset-4 col-sm-10
           col-sm-offset-3 col-xs-10">
30       
31     </div>
32   </div>
33   <ul class="nav nav-pills navigation hidden-xs">
34     <li class="active"><a href="#">Hjem</a></li>
35     <li><a href="#">Informasjon</a></li>
36     <li><a href="#">Om Ntnu</a></li>
37   </ul>
38
39   <div class="row" >
40     <div class="col-lg-6 col-md-7 col-sm-7 col-xs-12 well">
41
42     <h3><span class="glyphicon glyphicon-road"></span>
43       Finn rom</h3>
44     <form class="form-inline" role="form">
45       <input class="form-control" placeholder="Søk på rom">
46       <button type="submit" class="btn btn-info">Søk</button>
47     </form>
48   </br>
49   
50
51   </div>
52
53   <div class="col-lg-6 col-md-5 col-sm-5 col-xs-12 emneinfo" id="
           emneinfo">
54     <div class=" well">
55       <h3><span class="glyphicon glyphicon-book"></span> Finn
           emne</h3>
56       <form class="form-inline" role="form">
57         <input class="form-control" placeholder="Søk på navn
           eller fagkode">
58         <button type="submit" class="btn btn-info">Søk</button>
59       </form>
60
61     </div>

```



```

62     <div class="well snarvei">
63         <h3><span class="glyphicon glyphicon-hand-right"></span>Snarveier</h3>
64     <div id="snarveier">
65         <button class="buttons btn btn-info btn-large col-lg-6
66             col-md-4 col-sm-4 col-xs-6"><i class="icon-white
67                 icon-home"></i> Innsida</button>
68         <button class="buttons btn btn-info btn-large col-lg-6
69             col-md-4 col-sm-4 col-xs-6"><i class="icon-white
70                 icon-user"></i> Studweb</button>
71         <button class="buttons btn btn-info btn-large col-lg-6
72             col-md-4 col-sm-4 col-xs-6"><i class="icon-white
73                 icon-envelope"></i> Webmail</button>
74         <button class="buttons btn btn-info btn-large col-lg-6
75             col-md-4 col-sm-4 col-xs-12"><i class="icon-white
76                 icon-book"></i> Bibliotek</button>
77     </div>
78 </div>
79 </div>
80 </div>
81 <div class="row">
82     <div class="col-lg-6 col-md-12 col-sm-12 col-xs-12 well timeplan
83         hidden-xs ">
84         <h3><span class="glyphicon glyphicon-calendar"></span>Timeplan<
85             /h3>
86         
87     </div>
88     <div class="col-lg-6 col-md-12 col-sm-12 col-xs-12 nyheter">
89         <div class="well"><h3>Nyheter for studenter</h3>
90         <div>
91             <i class="icon-white icon-envelope"></i> Studenter lager
92                 matapp<br />
93             <i class="icon-white icon-envelope"></i> Innsida 4.0 nå
94                 ute<br />
95             <i class="icon-white icon-envelope"></i> Husk
96                 eksamensoppmelding<br />
97             <i class="icon-white icon-envelope"></i> NTNU student vant
98                 pris<br />
99             <i class="icon-white icon-envelope"></i> EIT landsby
100                 hjelper eldre<br />
101             <i class="icon-white icon-envelope"></i> Mange søkere til

```

```

93     neste års kull<br />
94     <i class="icon-white icon-envelope"></i> Psykolog om
       eksamenstress<br />
95     <i class="icon-white icon-envelope"></i> Mange vil ha
       bedre middag<br />
96     <i class="icon-white icon-envelope"></i> Høyt snitt på
       masteroppgaver<br />
97     <i class="icon-white icon-envelope"></i> NTNU i tet<br />
98   </div>
99 </div>
100 </div>
101 <div class="row">
102   <div class="col-lg-6 col-md-6 col-sm-6 col-xs-12 infokont">
103     <div class="well info">
104       <h3><span class="glyphicon glyphicon-bullhorn"></span>
         Informasjon</h3>
105       <div id="informasjon">
106         <button class="buttons btn btn-info btn-large col-lg-6
           col-md-4 col-sm-4 col-xs-6"><i class="icon-white
             icon-home"></i> Timeplan</button>
107         <button class="buttons btn btn-info btn-large col-lg-6
           col-md-4 col-sm-4 col-xs-6"><i class="icon-white
             icon-user"></i> Klage på karakter</button>
108         <button class="buttons btn btn-info btn-large col-lg-6
           col-md-4 col-sm-4 col-xs-6"><i class="icon-white
             icon-envelope"></i> Eksamensinfo</button>
109         <button class="buttons btn btn-info btn-large col-lg-6
           col-md-4 col-sm-4 col-xs-12"><i class="icon-white
             icon-book"></i> Søk etter studenter</button>
110         <button class="buttons btn btn-info btn-large col-lg-6
           col-md-4 col-sm-4 col-xs-12"><i class="icon-white
             icon-book"></i> Søk etter ansatte</button>
111         <button class="buttons btn btn-info btn-large col-lg-6
           col-md-4 col-sm-4 col-xs-12"><i class="icon-white
             icon-book"></i> IT-hjelp</button>
112       </div>
113     </div>
114   </div>
115
116   <div class="well kontakt">
117     <h3><span class="glyphicon glyphicon-comment"></span> Kontakt</
       h3>
118
119     <div id="kontakt" class="box-container">
120       <i class="icon-phone section-title"> Kontakt</i>
121

```

```

122     <div class="section-seperator">
123         <p>
124             <b> Spørsmål om studier </b><br>
125             <a href="mailto:studentservice@adm.ntnu.no">
126                 studentservice@adm.ntnu.no
127             </a>
128         </p>
129
130         <p>
131             Samfunnsfag og humanistiske fag
132             Tlf Dragvoll: <span>73 59 67 00</span>
133         </p>
134
135         <p>
136             Realfag, ingeniørfag, medisin
137             Tlf: Gløshaugen: <span>73 59 52 00</span>
138         </p>
139     </div>
140     <div class="section-seperator">
141         <p>
142             <b> Spørsmål om IT</b><br>
143             Epost, brukernavn/passord, programvare:<br>
144             <a href="mailto:orakel@ntnu.no">
145                 orakel@ntnu.no
146             </a>
147         </p>
148     </div>
149     <p>
150         <b> Web </b>
151         <a href="mailto:web@info.ntnu.no">
152             web@info.ntnu.no
153         </a>
154     </p>
155
156     </div>
157 </div>
158 </div>
159
160     <script type="text/javascript" src="js/rammeverk.js"></script>
161 </div>
162 </body>
163 </html>

```

## D.2 index.css

```
1 body {
```

---

```
 2     overflow-x:hidden;
 3 }
 4 .container{
 5     max-width: 90%;
 6 }
 7
 8 .bannerimg{
 9     max-height: 120px;
10 }
11
12 .navigation{
13     padding-bottom: 5px;
14     margin-top: 5px;
15     margin-bottom: 5px;
16     border-bottom: 1px solid gray;
17 }
18 .banner{
19     padding-top: 15px;
20 }
21 .bannerrow{
22     border-bottom: 1px solid gray;
23
24     display: inline-block;
25 }
26
27 .btn-info{
28     padding: 5px;
29 }
30
31 h3{
32     margin-top: 5px;
33
34     border-bottom: 1px solid gray;
35 }
36 .well{
37     margin-bottom: 15px;
38 }
39 #informasjon .buttons, #snarveier .buttons{
40     margin:5px;
41     width:45%;
42 }
43
44 .well{
45     overflow: hidden;
46 }
47 }
48
```

---

```
49 .form-control{
50     width: 70%;
51     display: inline;
52 }
53
54 .meny{
55     float:left;
56     width: 15%;
57 }
58 .col .well{
59     margin-bottom: -99999px;
60     padding-bottom: 99999px;
61 }
62
63 /* bottom row */
64 .col-base{
65     margin-top: -15px; /* cut off top portion of bottom wells */
66 }
67 }
68 snarvei{
69     height: 400px;
70 }
71
72
73 @media (max-width: 992px) {
74     .emneinfo{
75         padding: 0px;
76         padding-left: 15px;
77     }
78     .infokont{
79         padding: 0px;
80         padding-right: 15px;
81         padding-left: 0px;
82     }
83     .nyheter{
84         padding: 0px;
85     }
86 }
87
88 @media (max-width: 768px) {
89     .emneinfo{
90         padding: 0px;
91     }
92 }
93
94 .infokont{
95     padding-right: 0px;
```

---

```
96     padding-left: 0px;
97   }
98   .form-control{
99     width: 80%;
100  }
101
102  .nyheter{
103    padding: 0px;
104  }
105
106
107 }
108 @media (max-width: 1271px) {
109
110   .nyheter{
111     padding: 0px;
112     padding-left: 15px;
113   }
114   .info{
115     padding-right: 15px;
116   }
117   .infokont{
118     padding-right: 15px;
119     padding-left: 0px;
120   }
121
122   .nyheter{
123     padding-left: 0px;
124   }
125 }
```