



Technical Report

Semester Project 1

Egil Johnsen Dahle

Word count

Summary: 250 | Main text:1000



School of technology and digital media

Table of Contents

1. Summary	3
2. Body	4
2.1. Introduction	5
2.2. Main section of report	6
2.3. Conclusion.....	14
3. References	15
4. Acknowledgements.....	Feil! Bokmerke er ikke definert.
5. Appendices.....	Feil! Bokmerke er ikke definert.



1. Summary

This report is about “Semester Project 1”. This is the first semester project, and contains three courses: Design, HTML&CSS, and Project Methodology. The last of these required us to use Agile methodologies in this project. I did not really achieve this, but I believe I learned a lot anyway.

Agile is – as I did not fully realize the scope of, until the last week – about small, measurable features being produced fast. My plan did not do this, even though I believed it did. My plan segmented the design part from the html/css part completely. A more natural way would be to make some big decisions, like typography and colour, and then do small parts of the page. These parts could have been “design and make header” or “design and make footer”, or more specific like “design and implement hamburger/navigation menus.

Another thing that did not work properly was my motivation. Instead of spending all the time on this, I instead did other projects, like a full stack (node, express, sqlite3, typescript) quiz system that displayed new questions every school day in December and a database that stored (and gave me an overview) the answers given by students and teachers. I think better planning could have prevented this, as one of the problems the scope of the tasks I set myself and how boring it is to not have a programming language involved.

It should be evident that my efforts in this project weren’t at the level it should have been.



2. Body

Links to project files:

Figma Dev Mode

<https://www.figma.com/design/I0rxruWJ8oIUIvPo6eujS/Community-Science-Museum?node-id=o-1&m=dev&t=ereXtW94yBHYIFFP-1>

Figma prototype desktop

<https://www.figma.com/proto/I0rxruWJ8oIUIvPo6eujS/Community-Science-Museum?node-id=12-618&t=ereXtW94yBHYIFFP-1>

Figma prototype phone

<https://www.figma.com/proto/I0rxruWJ8oIUIvPo6eujS/Community-Science-Museum?node-id=111-907&t=ereXtW94yBHYIFFP-1>

GitHub Pages

<https://lektordahle.github.io/semester-project1-egil-dahle/>

GitHub Repo

<https://github.com/LektorDahle/semester-project1-egil-dahle>

Github Project

<https://github.com/users/LektorDahle/projects/7>



2.1. Introduction

Community Science Museum is an open and accessible museum. The first two tasks to complete was logo and typography.

I chose the font “Poppins”, and below you see the logo.



Figure 1. Logo of Community Science Museum

The museum targets many demographics – teachers, kids, researchers, and the community at large are some examples. I made it a priority to not make the site too fancy. Movement and many colours are a nightmare if you aim for accessibility.

As a teacher I have experienced visiting sites for museums and science exhibitions aimed for education. They have to balance promoting themselves and giving practical information. I wanted to make the site as clean and consistent as possible, to make the experience of the user better.

I aimed to make this as accessible as possible, in regards to colours. I think (from testing) that all text-background relations are at AAA WCAG, but using opacity in the header means that this might not be true there.

2.2. Main section of report

Typography

The typography was the first thing that was decided. Below are the html-tags and the sizes they should have.

Name	Paper	Screen	Line Height
h1	24 pt	2rem	1
h2	17 pt	1.4rem	1.2
p	12 pt	1 rem	1.4
:root{font-size: clamp(14px, 1.5vw, 18px)}			
a, button, input, textarea	12 pt	1 rem	
blockquote	13 pt	1.1rem	1.6
small, figcaption, caption	8.5 pt	0.7rem	

Table 1. Table of the different HTML-tags and their typographic profile

This served as a guide to create consistency in the html code.

The geometric font “poppins” was chosen for the typography in this project. It is legible and looks both professional but can be quite playful at the same time.

Poppins Bold used for h1

Poppins Semi Bold used for h2

Poppins Regular used for paragraph, input and textarea

Poppins Medium used for a and button

Poppins Light Italic used for blockquote

Poppins light used for small

Poppins ExtraLight Italic used for figcaption and caption

Figure 2. Display of the different typographic profiles of Poppins.

Since I had to use screen-sizes to check that the clamp makes sense, I include this here as well. The dynamic break-



points for the page will have to be around 1000 px - 1200 px and around 600 px - 900 px - giving two @media rules for screen sizes.

The clamp reaches its minimum at 933 px view width and below, and its maximum value at 1 200 px view width and above.

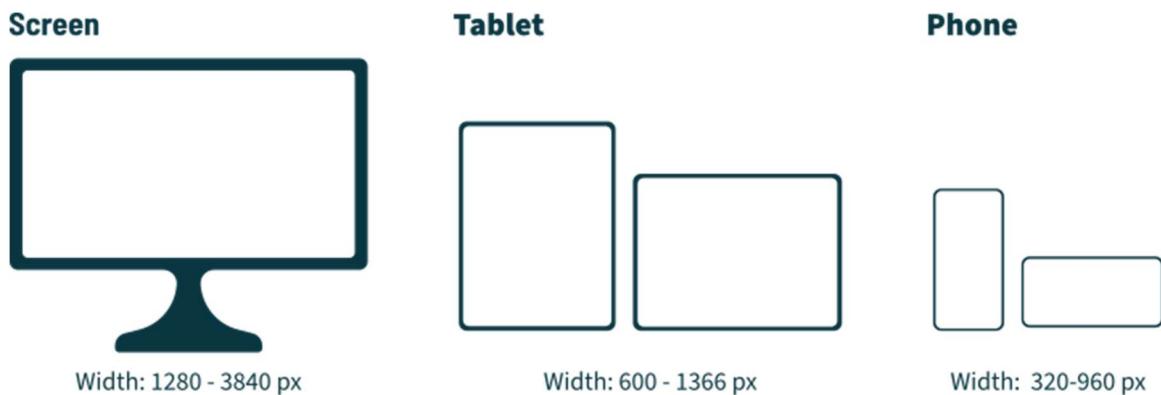


Figure 3. Different screen-sizes.

@media breakpoints

Choosing the width breakpoints, or hinge-points, can be difficult. This section is not to limit or determine what is to be used, but to assist and guide instead. I hope that by putting some numbers, with its reasoning, on paper, will help me make better decisions regarding all the @media rules used in the project. In the end, making the page and testing, is the only way to know what works, and what doesn't work. Sometimes higher fidelity mechanisms might be wanted in elements, requiring the use of more @media breakpoints.

The main goal of this section is to say when the menu - if needed - must move into a hamburger menu.

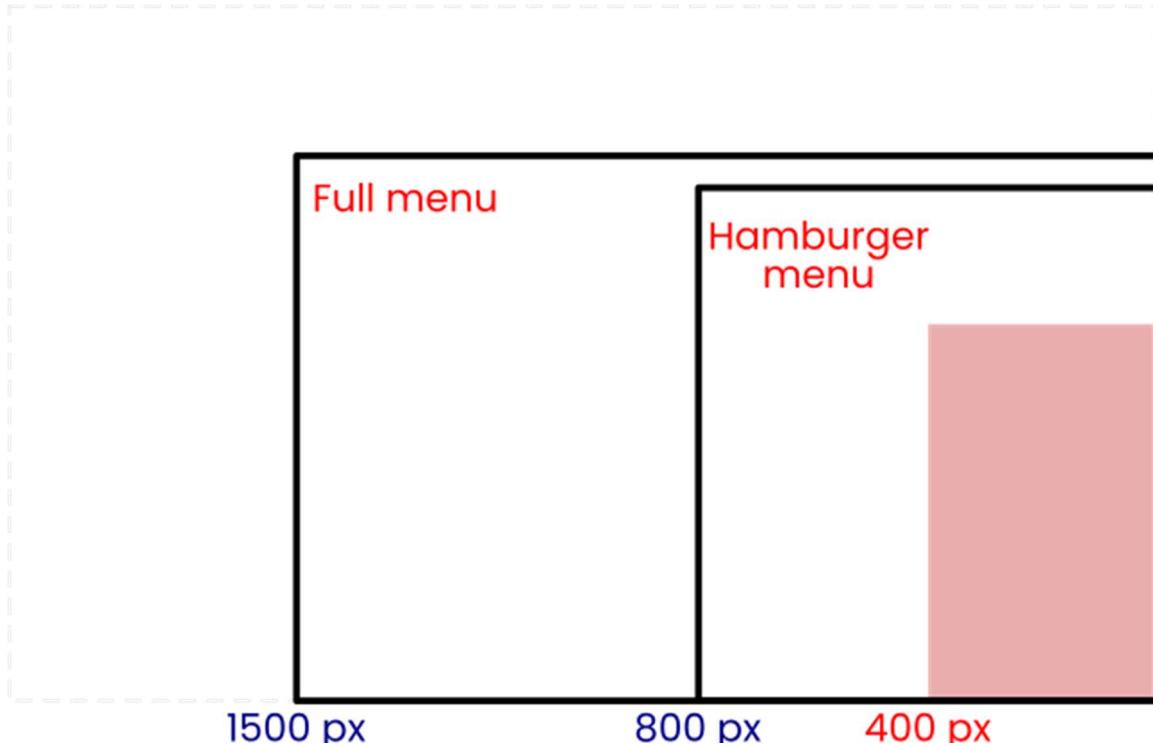


Figure 4. A guide for when the hamburger should be considered

From an early draft in figma, it became apparent that the header should look something Figure 5.

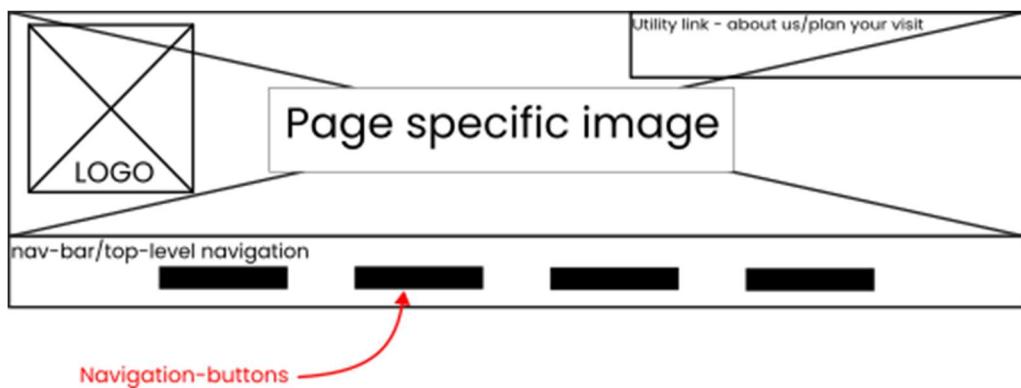


Figure 5. A draft version of the header

In Figma, these four navigation-buttons each took 177 px width. This means that we have about 20 px left-right margin on these at 800 px. See Figure 6. The Figma-prototype did not use clamp for typography, so the end result will have more space for margins, as the text will be smaller in size.

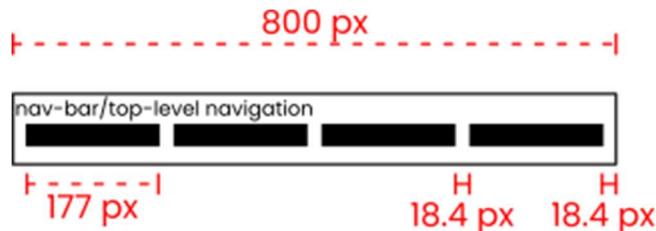


Figure 6. Top-level navigation space requirements

Why not use 1200 px as max size, as written in the typography-section? The logic from this section stands, and 1200 px can be a hinge-point. I will add that the text should have reached its largest before we reach the @media breakpoint, so 1200 px is too small for this while 1500 px might be to large. Only testing and building will tell.

After 1500 px the page should not grow the content to size, there is no reason to try and take advantage of this much screen-width on a site for a museum. If we do try to take advantage of all the screen width this can cause information overload.

This site will not be tested (or made) for screens below 400 px in width. No sane person will use such a small screen size to look for any other information about the museum other than opening hours - which will appear at the top of the site. At the museum, the information should be easily available, and visitors should be encouraged to not use their phones.

Colour Palette

First I wanted to make the site the same way I did for the first html/css submission, with a dark and a light mode. I also taught that I definitely had a good palette, but from designing in figma, it became apparent that I needed better contrast. AA was not good enough for what I wanted. In the end I ended up with the palette in Figure 7. Some of the old

names have remained though, these are not accurately describing their colours.

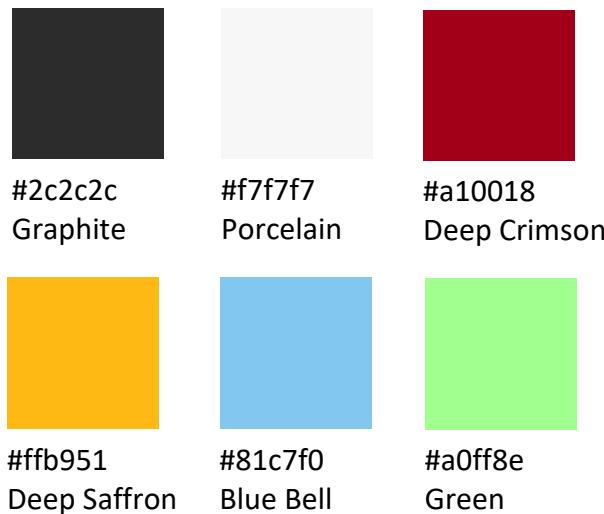


Figure 7. The colour palette

Intended Information Architecture

Below is a list of how I imagine the webpage at this moment. This is after the low fidelity wireframes, NOT the high fidelity wireframes. As you see from the Figma wireframes, these are different from the first sitemap.

- Top-level navigation
 - Home (Logo)
 - Explore (dropdown)
 - For Teachers
 - For Researchers
 - Shop
 - About Us (Apart from navigation menu?)

- Explore (dropdown)
 - Activities
 - Exhibitions



- Tickets and Admission
- Plan Your Visit
- For Teachers
 - Resources
 - Information about visits (prices, lunch/food, activities, link to accessibility page)
 - Booking visit
- For Researchers
 - About our projects
 - Labs and facilities
 - Access to archives and collections
- Shop
 - Merchandise
 - Science kits
 - Opening Hours
- About Us
 - Sponsors
 - Who are we?
 - Contact information
 - Mission and vision
- Plan Your Visit



- Accessibility
- Opening Hours
- Address
- Directions
- Facilities
- FAQ

Images

I set my mind to test using squircles for image shapes. Every image on the site should be a squircle. A squircle is between a circle and a square. It is neither. And it can look much better than a square with rounded corners (square with quarter circles for edges).

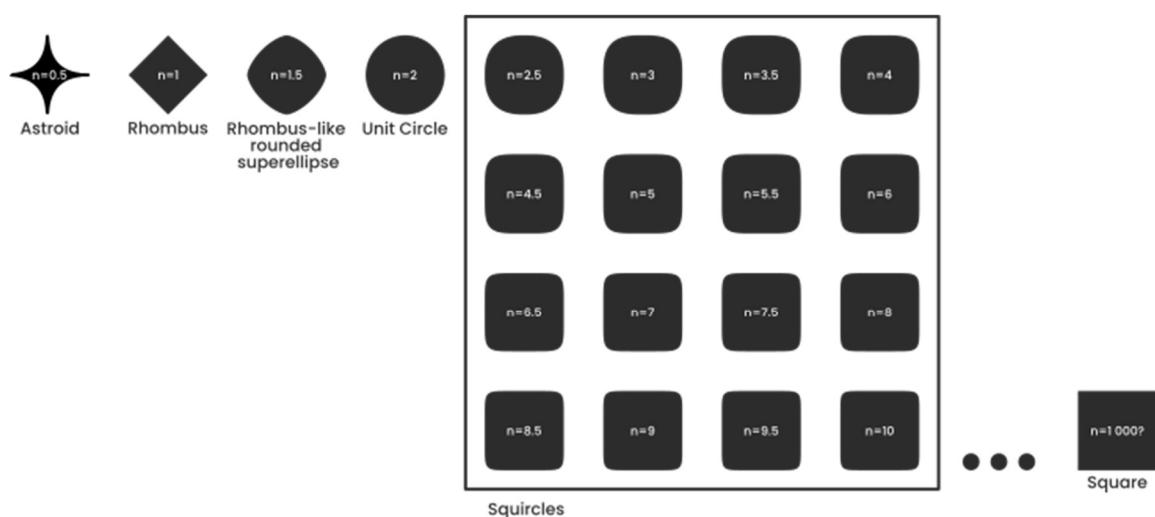
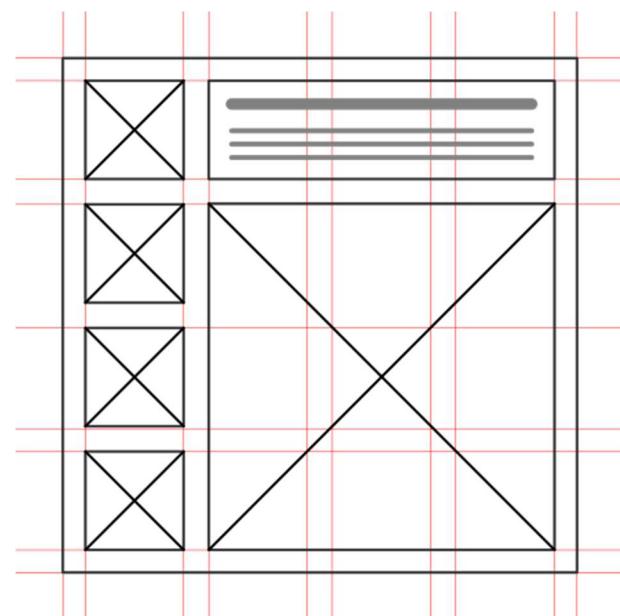
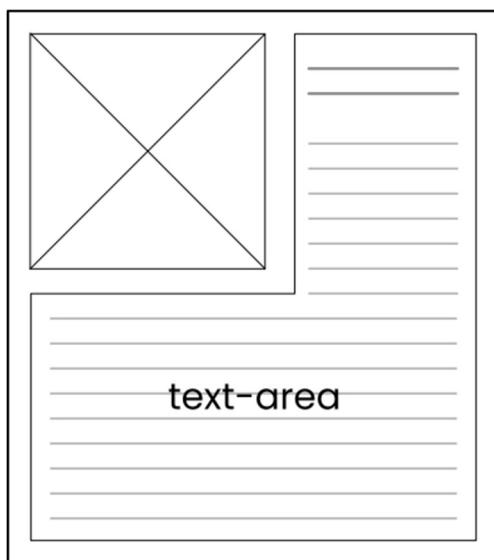


Figure 8. Squigonometry

On the next page are other sketches (unsorted and not numbered) I used in this project.



header-desktop



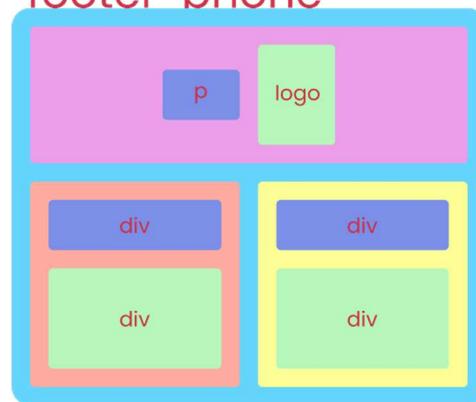
header-phone



footer-desktop



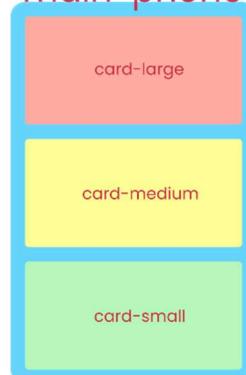
footer-phone



body-1-desktop



main-phone



2.3. Conclusion

I am not satisfied with the outcome of my work. Mostly from my ability to not do the work and do side-projects like image-converters to view images placed in HTML-canvas as people with colour-impaired vision would see it. The JavaScript/TypeScript made these project much more motivating. Also, by accident, these projects ran closer to agile methodologies than this, the one project I wanted to use agile in.

There are aspects I am satisfied with, and I might revisit this and improve this site later – using local storage, JavaScript and actual smart ways of creating hamburger-menus, not hacks.



3. References

Book: Color Accessiblity Workflows by Geri Coady. A book apart. ISBN-10: 1937557561

I either used work from my previous submissions, the report files from these can be found in the _docs folder of the repo.

ChatGPT and Gemini was used as code-reviewers and did help on this project. I did write all the code myself, and mostly without outside assistance.

The one problem I did not solve myself or steal from my previous work, came from

<https://stackoverflow.com/questions/10468554/dropdown-menu-without-javascript>

