Noroff
School of technology
and digital media

Project Exam 1

Linn Melberg

Word count:

1104 (excl.references)

In this report I will go though my process of making my exam 1 project from ideation to finished product.

My main thought during this assignment was to keep in mind all the things we have learned this year, and to improve upon the things I felt I could have done better on previous projects (semester project 1, cross course project).

First I started with ideation and design. I wanted to pick something I am interested in. I chose to do an e-sport news blog. I created the name, then my colour palette and my makeshift logo. I also tried to find images online. I used mostly unsplash, but it was hard to find good images. I was unsure if official team photos from organizations were free to use so I opted out of using those. The wordpress posts / API took some time to make, but I made a post template that sped up the process. I converted the images using cloudconvert.com before uploading them to WP. I used One.com to host my API. I personally find One.com to be difficult to use and too complicated for my intended functionality, but I bought it so no point in using something else, but I will probably check out flywheel's hosting at a later time as it works well with the localByFlywheel app.

I created my design with simplicity in mind. In pervious projects I feel that my designs have been cluttered or too busy. I wanted to keep the design and code as clean as possible so it would be easier to add or modify the site later without worrying about breaking things around the site.

In order to deploy my site though netlify.com I had to transfer ownership of the github repo to myself. I therefore made it public so you (noroff) can access it.

I plan when I started making the code was to first create the reusable objects and making them scalable. I wanted to have them all in one place so that if I had to make any changes to, say the navigation, I would only have to update dynamically it in one place, and not on several HTML pages. I the global features (header, footer) in my script.js file. All HTML pages have links to styles.css, mediaqueries.css, and script.js which contains global and reusable elements. The pages also have page specific links such as about.css and about.js. Mediaqueries for page specific items are at the bottom of the page specific css file. This is how I wanted to organize and create pages, after having learned more and then seeing old projects and how they were organized.

I liked having one common and one specific js file for each page. For example the breadcrumb const was declared on the common file, so that I could use it in the specific js file to create the correct content for the page.

During the coding process I met several issues I didn't think I would be a big deal, which turned out to take hours longer than I anticipated. I had trouble finding out how to get the blog image from the API. I spent a long time trying to find the answer online, and was honestly annoyed at how hard it was to find the right information in the wordpress documentation. Finding out that I was missing "?_embed" at the end of my URL took me about 1.5 hours. That plus the modal (especially how to close it) and carousel probably took up most of my time in terms of functionality. Lots of things I am not sure of I was able to

solve in the "best and most correct" way, but it does what it needs to and I am happy with the project.

I had some of my roommates test the site and used their feedback to change elements. The feedback was mostly on design and font size. Other comments were related to the contact form, for example that it should not be necessary with 15 characters in the subject field, but that was irrelevant feedback. I tested the site on mobile every once in a while when making the pages. I would push changes and open the website on my phone which I found useful so I can see why developers have several devices with the site open on their desk. I checked the site for any colour-blindness issues using toptal.com and ran it though the validator on w3.org and accessibilitychecker.org and fixed some issues with my H3 and H4 on the home page.

All in all I am really happy with my result. I feel like I am getting the hang of things and I see the progress. Especially in my code. Some of it I still feel like needs some work, but it is fun to find new and easier solutions to things that were super challenging 5 months ago. I am also learning more abo9ut what I enjoy doing. JS is the favourite so far, even though I am not the best at it. HTML is cool, but CSS is the worst. It is necessary, but tedious and boring work. I really enjoy making functionality and making my life easier in the code, even though it looks the same on the outcome of the page.

Summary shorthand:

Design
What went well on the project – I like the simplicity of my site, it looks clean and organized with enough space to breathe.

What was difficult/didn't go well on the project – coming up with new ideas, I feel like no matter what I somewhat end up with similar designs.

What would you do differently next time – make the design less boxy.

Technical
What went well on the project – I managed to make the modal, and slider which is a new skill. Also, the view all posts on the posts page was pretty cool to see in action.

What was difficult/didn't go well on the project – closing the modal did not turn out exactly how I wanted. View all posts could probably be done differently in JS.

What would you do differently next time – not spend too much time on a problem. Work on something else for a bit, and the answer might be more clear after some time away.

WCAG guidelines, content management and SEO

What went well on the project – pretty happy with outcome.

What was difficult/didn't go well on the project - I had to change h3 and h4 positions.

What would you do differently next time – be more conscious of headings, it took some time to clean up the css after swapping them out.

References:

- Noroff lessons, modules, and projects
- Roommates for testing and feedback
- Noroff social discord / classmates

- "Ada & WCAG Compliance (Free Scan)." *Accessibility Checker*, 10 Oct. 2022, www.accessibilitychecker.org/.
- Abu, Jemima. "How to Build a Simple Carousel with Vanilla JavaScript (14 Lines of Code!)." *Web Design Envato Tuts+*, Envato Tuts, 2 June 2022, webdesign.tutsplus.com/tutorials/how-to-build-a-simple-carousel-with-vanilla-javascript--cms-41734.
- *Adobe Inc. - Libraries*. www.color.adobe.com/.
- "The Collaborative Interface Design Tool." *Figma*, www.figma.com/.
- "Colorblind Web Page Filter." *Colorblind Web Page Filter Testing Tool & Simulator | Toptal®*, www.toptal.com/designers/colorfilter.
- "Connect to Flywheel." *Local*, localwp.com/connect-to-flywheel/.
- "Develop & Deploy the Best Web Experiences in Record Time." *Netlify*, www.netlify.com/.
- "Fast, Secure Managed WordPress Hosting." *WordPress.com*, wordpress.com/.
- "File Converter." *CloudConvert*, cloudconvert.com/.
- "How to - Modal Images." *How To Create Modal Images*, www.w3schools.com/howto/howto_css_modal_images.asp.
- *MDN Web Docs*, developer.mozilla.org/en-US/.
- "One.com Web Hosting - Domain • Hosting • E-Mail ." *One.com Web Hosting - Domain • Hosting • E-Mail* , www.one.com/.
- "Rest Api Handbook." *WordPress Developer Resources*, developer.wordpress.org/rest-api/.
- Temani Afif on Dec 2, et al. "Tricks." *CSS*, css-tricks.com/.
- Unsplash. "Beautiful Free Images & Pictures." *Unsplash*, unsplash.com/.
- *W3C*, www.w3.org/.
- "W3Schools Free Online Web Tutorials." *W3Schools Online Web Tutorials*, www.w3schools.com/.
- "Website Heatmaps & Behavior Analytics Tools." *Hotjar*, www.hotjar.com/.
- "Where Developers Learn, Share, & Build Careers." *Stack Overflow*, stackoverflow.com/.