



Noroff

School of technology
and digital media

Semester Project 2

Pål Nicholas Bakken



School of technology and digital media

Design

What went well on the project

I wanted to keep the design simple for this one, and I think the end result works quite well. I'd normally like to include more purely decorative design elements and trim, but I've pretty much stuck to only including the required content this time and the website feels cleaner for it.

The photographs ended up working really well. I was able to borrow a well-lit corner of a local florist's along with some of the inventory, and had an hour more or less to myself to get what I needed. I got the image for the hero banner one afternoon outside in my garden. I feel like there's a cohesive feel to the images in terms of light and colours which feeds in to the overall theme of the design.

What was difficult/didn't go well on the project

I still don't have a good grasp of bootstrap. I can see the value in it, and I think I will have to conclude it made certain things easier, but I can't escape feeling like I'm in a constant battle with it. If I finally chase down the one hidden class property of any given element that needs changing, something else breaks because of the change. As a result I've pretty much mostly used it for the navbar, and I can't say I'm entirely happy about that either. There's a spacing issue that comes from one of the support classes around the search input that prevents me from aligning everything along the baseline, for example. While I do appreciate the ready-made hamburger menu, trying to change the nav link colours left me with an invisible menu toggle, which is why the nav-links use the Bootstrap-default black and grey.

What would you do differently next time

At first I was planning to reuse the concept of my first year project exam, and just sell photography prints. After some thinking and discussing with a tutor, I came to the conclusion that it would be better to have a variety of project



themes for my future portfolio. This left me with a problem, as I was a week and a half in and suddenly unsure of what content I would use. I left my photography design drawings as they were, and got to work on building the basic html and javascript modules. I carried on this way until the underlying systems were basically feature complete, but I was still lacking a theme and had virtually zero css or extra structural html.

I was eventually able to mix in parts of my original design plans, and I'm pleased with the end result, but it was unusual to the point of almost making me feel uncomfortable (with regards to time management) to design and style the site around the underlying code as opposed to fitting the code in to an existing design.

I say that I would do it differently, but now having finished the project, I could see a case being made for working this way. I suppose it all depends on what best fits the current project. (I probably will do it the usual way around on my next project though.

Technical

What went well on the project

Every part of the code works, as far as I can tell. I'm sure things could be split up even more but at the same time the programming is sufficiently modular that it's given me a noticeable benefit in being able to slot different parts in and out as I was going along, or making changes now at the end. While I was working I was aware of people struggling with the file upload for product images, so when I came to it I immediately went for the image_url string and never looked back. It adds an extra step to the end user, so I can see the benefit of using an image upload. In terms of benefits to me working on my semester project the url string was definitely the way to go though.

Overall the coding went reasonably fast. Ran into some unexpected results underway, but I had time to sort those out, as well as time to hunt down some



more obscure bugs I became aware of as I adding to the display code and using the system over and over.

What was difficult/didn't go well on the project

When I started coding I had in the back of my mind a piece of feedback from an earlier project about my codebase being too large. I began working with this in mind and tried to keep things more to the point. But by the end of it I feel like I still have a massive labyrinth of project folders and overall I'm unsure if I should have made things even *more* modular and reusable or if I've missed the mark completely.

What would you do differently next time

I'd like to better learn project management strategies with regards to the architecture and planning of such a large project (relative to our previous course work).

The base idea I've had for the javascript modules is to split things up roughly along the lines of one group for systems and underlying computations ("utils" in the modules folder, not the best name but I'm stuck with it until the deadline for this), and the rendering of the results of those computations and interactions with them ("display") — the admin system got a section of its own. I stayed more or less within those bounds the whole way, but there are parts where they blur together and I would like to be more confident in feeling like I know what I'm doing.



References

<https://freesvg.org/1326364747> blue flowers .svg used for logo



