



Technical Report

Holidaze

Emma Haavik Tønnessen

Word count

Summary: 99 | Main text: 1501

Table of Contents

Holidaze	1
Emma Haavik Tønnessen	1
Word count	1
Summary: 99 Main text: 1501	1
Table of Contents	2
Summary	3
Body	4
2.1 Introduction	4
2.2 Main section of report	4
2.2.1 Planning	4
2.2.2 Design	4
2.2.3 Technical	5
2.2.4 Styling	6
2.3 Conclusion	7
References	8
Acknowledgements	9
Appendices	10

1. Summary

My holiday website is running with the functionality required by the assignment repo, however, there is always more that can be done with a website, and I hope to add more functionality to the site for my portfolio at a later stage. I catch myself thinking, as I always do after I “finish” a project, that I could start anew, knowing what I know now. I guess that’s not entirely a bad thing, though. It shows that I have learned new things while working on this assignment that I will bring with me to new projects, and into worklife.

2. Body

2.1 Introduction

First, I don't have word, and my online microsoft account I have access to through my school account won't let me pass in my username and password, so the template is a bit off, though I have tried to make it as similar to the template as I could.

I started this project by reading through the repo, making a plan using the gant layout and gathering things like fonts and colors I would like to use for this project. I then went on creating the more technical aspects of the site, before I styled it using Sass and had a fellow student look through my code, as well as having friends and family look through the site. Both live, and with hotjar installed. In the following, I will discuss more about what went well and what I could have done differently in the different aspects of this project.

2.2 Main section of report

2.2.1 Planning

The assignment said that we should use the gantt chart method to plan our project, so I went ahead with making that. On reflection, I wish I had made a trello board as well, as this is what I usually use, and I have found it works well for me. It fits me better because it's more dynamic, and I did end up making one on a weekly basis for the last two or three weeks, anyway. The Gantt Chart is added within the zipped files, and as you can see from it, my involvement with it in the end is lacking.

Other than that, I did not read through the repo thoroughly enough. In my prototype, which is linked in the delivery, I named the brand "Herbergen", because I did not see that it was supposed to be named Holidaze. Fortunately, both started with the letter H and I did not have to make too many changes, but it could definitely have been a bigger issue than it was.

2.2.2 Design

My website deviates from the prototype (a link to which can be found in the appendices), quite a bit. The prototype was a note quite my style. The colors were too strong, also the contrast was, of course, better with the white background, so I decided to change it to the way it looks now. I suppose I probably should have been more accurate with my prototype, but I

find I like to experiment when I start doing the styling anyway, and like to use the prototype more as a guide, rather than a strict set of rules to follow.

I hope to improve my logo, and icon making skills in the future. As I would have liked to play around with it a bit more, but didn't want to spend too much time on it, as there would be plenty of work to be done.

2.2.3 Technical

Most of the things I was required to do for the project, I had done before, and I was confident I could do them fairly quickly, so I decided to add more functionality when I prototyped the design. That was a mistake.

Sure enough, the things weren't actually so complicated, but I ran into so many hiccups along the way, and spending time trying to figure out how to add additional functionality is something I should have saved to the very end. I think I would have been able to implement more of them, if I had done that, as I would have been more focused in my work, and maybe would have been able to have my code reviewed a lot earlier, which was super helpful. This is the feedback I got from my fellow student, Tommy Johnsen:

Filstruktur: Dette kan være/er et spørsmål om preferanser, men etter min mening ville det gi bedre oversikt dersom alle routes/sider var samlet i en felles mappe som f.eks. heter pages, i stedet for å ha hver side alene i sin dedikerte mappe. Så ville jeg ha prøvd å samle øvrige komponenter med felles karakteristikker, f.eks. alle forms/logo/common-components i egen mappe.

Layout komponent: Denne komponenten kunne med fordel vært arrangert slik at header og footer kun ble importert til denne komponenten for så å implementere det varierende hovedinnholdet som en children-prop. På denne måten kan layout wrappes rundt alle sider med fiks ferdig header og footer. Dette vil spare mange unødvendige importter og gi ryddigere kode

Enquiries form: Hvis du vil gjøre det umulig å velge en from-date og to-date som er tidligere enn dagens/morgendagens dato, er det mulig å gjøre dette med å teste opp mot `reacts new Date()`. Jeg brukte forresten moment formattering i kombi med `new Date()`, for den gir vel klokkeslett osv også.

Roughly translated:

File structure: This might be about preferences, but in my opinion, it would be a better overview if all the routes/pages were gathered in one folder, called “pages”, for instance. I would try to gather other components with similar characteristics in their own respective folders.

Layout component: This component could be rearranged so that the header and footer only were imported to this file, and a main section with a children prop, where the content on the pages can go. Then the layout can be wrapped around all the pages with finished header and footer. This will save you many unnecessary imports and result in cleaner code.

Enquiries form: If you want to make it impossible to choose a from-date and to-date, which is earlier than today's date, it is possible to do that by testing it against react's new `Date()`. I used moment together with new `Date()`, by the way, as it also allows for time.

I let the pages be in their own separate folder, as this is what was done in the lessons, but I knew that the structure would have to change already after making a few components. I rearranged them in a way that makes sense to me, but I still don't think it's optimal, but I do not know what is the best solution. I need to work on that, moving on.

I changed the layout component to the way he described it, and it makes so much more sense. I'm glad he noticed that. Overall, getting someone to check my code was a nice experience, and I wish we could have done it for other projects as well.

Regarding the dates, I had previously used another way to display the “publishedAt” property, because I had forgotten about moment, but moment needs fewer lines and gives the code a cleaner look, so I changed to using that. I also made it so that it was impossible to pick an earlier date than today's date, like Johnsen suggested. He also reminded me of Lodash, which made me able to make some sorting functionality on the accommodation page in time. This was also something that had been requested by my friends, who had tested the site.

2.2.4 Styling

I used Sass and BEM to style my site, as this is the way I like the best and am most comfortable with. Next time, I will make sure to make more of the global styles first, so that I follow DRY more. There are some places that I repeat myself, where I could have not done so if I had planned a little bit better prior. For instance with the buttons. Of course this is possible to change afterwards, but it takes more work, and I decided to focus my effort elsewhere.

After testing with people on hotjar and in real life, the thing I got most feedback on to change was the font. My friend, who is a web-designer, also recommended lowering the shadowing, so I have done that as well. This would have been much easier, had I made the shadowing a variable, so I made sure to do that this time around.

2.3 Conclusion

I more and more get the sense that a site never really is finished as there is always something that could be improved upon. Going forward, I want to focus more on one thing at the time, and do it well, before moving on. And I want to be more strict when prioritizing the order in which I choose to do what needs to be done.

Besides using the lessons as a guide on how to do different things, I have also used other resources. I reference the ones I've used as inspiration for a particular piece of code inside comments right above where that code is being executed. None of them is straight up copied, though, except for the `UseLocalStorage.js` file (I know that hook file names should be written in camelcase, but I made a mistake when I made them, and when I tried to change it, there was a problem with netlify, even though I changed my imports). I'll also list the resources down below.

3. References

- <https://stackoverflow.com/questions/36606105/lodash-orderby-on-nested-property>
- <https://stackoverflow.com/questions/59835355/get-textarea-character-count-using-reactjs-with-help-of-usestate-hook>
- <https://ibaslogic.com/how-to-add-hamburger-menu-in-react/>
- <https://www.youtube.com/watch?v=x7niho285qs>
- <https://usehooks.com/page/3>
- <https://stackoverflow.com/questions/58810127/how-do-i-validate-if-a-start-date-is-after-an-end-date-with-yup>
-

4. Acknowledgements

Thanks to Tommy Johnsen for reviewing my code!

5. Appendices

XD Prototype:

<https://xd.adobe.com/view/9281849d-e9b8-4718-b020-55356c601505-c7d2/>