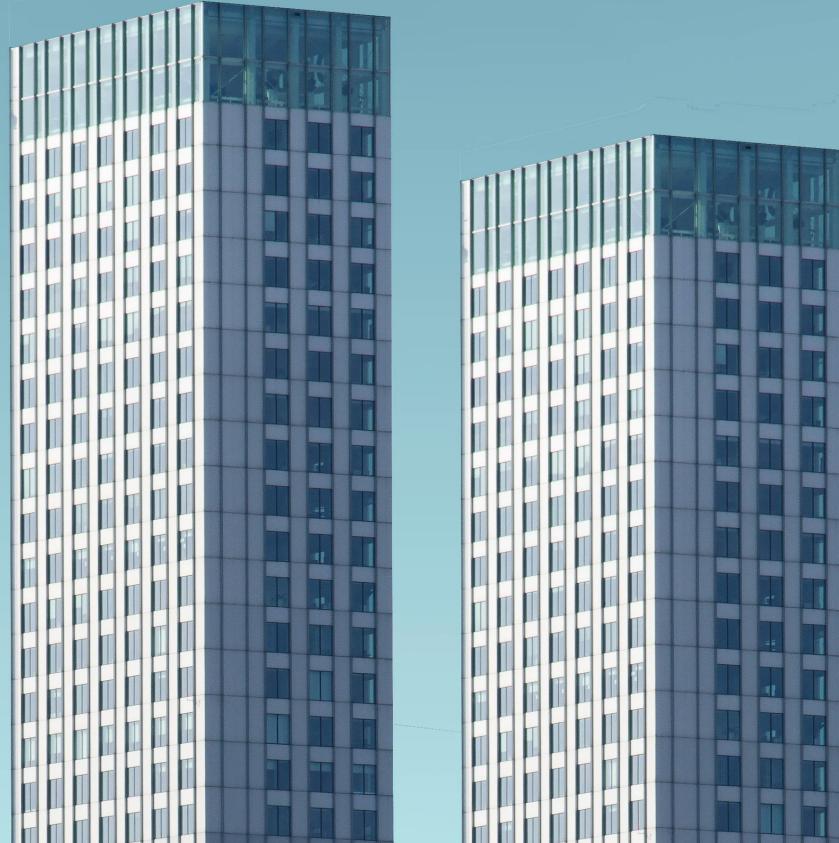


HOLIDAZE

Report Holidaze

By Bjørn-Ivar Skuggen



Content list:

1. **Introduction**
2. **Planning**
- 2.1 **Functional Specification**
- 2.2 **Gantt Chart**
- 2.3 **Trello Board**
3. **Design**
 - 3.1 **Style Tile**
 - 3.1.1 **Colors**
 - 3.1.2 **Icons**
 - 3.1.3 **Buttons and links**
 - 3.1.4 **Boxes**
 - 3.1.5 **Typography**
 - 3.2 **Prototype and Layout**
4. **Development**
 - 4.1 **HTML / CSS**
 - 4.2 **Dependencies**
 - 4.3 **Fetching and searching data**
 - 4.4 **Posting data**
 - 4.5 **Handling errors**
 - 4.6 **Issues along the way**
 - 4.7 **Errors**
 - 4.8 **What if the servers are down?**
 - 4.9 **Extra validation**
5. **Responsive**
6. **Code review**
7. **User feedback**
8. **Conclusion**
9. **Links and documents**
10. **Stock photo creds**
11. **Github repository**

Introduction

At 27th of April 2020, we got our final exam task. This task was to create a website where the users could book hotels, B&Bs and guesthouses. The user of the website should be able to search for an establishment, read more about it, book it, and contact the Admin of the website if he/she has any questions or feedback. This data should write out to a JSON file to be used later in the project.

We would have to make an Admin-part of the website. This would be where the Admin could read the enquiries sent by user, read contact messages, and create new establishments. To enter the Admin-sites, you would have to log in.

The process was divided into four parts; planning, design/prototype, development, testing/bug fixing. At the end of the project, the user sites and admin sites should be working towards each other to get the best user experience for both parts. I have tested a few different dependencies and techniques throughout the project, and I think I ended up with a good result.

All documents and links will be provided both throughout the document where it is relevant, while also be listed at the bottom of the document.

Planning

Functional Specification

A part of the planning-task was to create a functional specification. As this was something I had never done before, I used the setup given to us, and tried to fill this in with information about my project. I created some Data Flow Diagrams to show how the user and admin sites should operate, and how the trails should be when using these. This gave me insight in how to set up the website in a logical and usable manner. It also made me think about how the user would view, understand, and use the website, so I got a better understanding of how it should be structured.

Download or view the functional specification [here](#).

Gantt Chart

As a part of the planning, I made a Gantt Chart to keep track of what had to be done, and when it had to start or finish in order to stay on track. This included the four different stages mentioned in the introduction; planning, design, development, testing/bug-fixing.

For the planning and design part, I had two weeks. This included creating the functional spec, Gantt Chart, design style tile, and to create a complete design and prototype of the project.

For the continuing three weeks, I worked with the development, which included creating the entire project in React, and making sure it worked on all viewports.

The last week was dedicated to be for testing and bug-fixing, and to just clean up the code to make it easier to read and understand.

Note: I am aware that the Gantt Chart is created with an additional week. This was a mistake I made and realised it first after I had submitted it. However, this has not caused any issues as I have worked according to the plan, only with three weeks of development instead of four.

Gantt Chart for HOLIDAZE

Week 1	Week 2	Week 3	Week 4
PLANNING AND DESIGN	PLANNING AND DESIGN	Development	Development
Create functional spec	Design website for Desktop	Create Project using React	Start VISITOR sites with backend data
Create Gantt Chart	Design website for Tablet	Setup project using Sass	Replace placeholder data
Create Style Guide	Design website for Mobile	Start developing layout for the website	Post data from enquiries and contact form
Check out competitors	Make working prototype of these	Setup Backend connection	
Guidance with Cameron	Guidance with Cameron	Try to get some data from backend	
	Delivery of functional specification, Gantt Chart, Style Guide, and Adobe XD Design / prototype		
	Lesson about PHP installation with Cameron		
Week 5	Week 6	Week 7	
Development	Development	Bug fixing & Delivery	
Start ADMIN sites	Finish all technical on sites	Remove all warnings / errors	
Receive data from VISITOR sites	Style all sites	User testing to see if everything works	
	Early user testing	Device / Viewport testing	
		Refactor code	
		Student checkup	
		DELIVERY 05.06.20	

Trello Board

To make sure I would not forget things along the way, I created a Trello Board. This was not something the assignment required, but it is something I do for all my projects. Doing this, I can easily keep track of what I should do, am doing, and going to do. Not all ideas can be pursued right away, so to keep them in mind, I always put them in a backlog to check out later.

The Trello Board is also great to put in small reminders of bugs to be fixed, details that should be re-done, etc. By using this tool and technique, I have reduced time on bug-fixing and testing by solving many of these small issues along the way in my “clean-up” periods (usually before I made significant changes to the project).

Check out the Trello Bord [here](#).

Design

“Make it simple, but significant” - Don Draper, fictional character on Mad Men.

This pretty much sums up what I wanted to do with my design for this project. My two main goals were to design and create a website that was easy to use, while also making the users navigate the site after a particular and planned pattern. With clear and consistent navigation, not only through the navigation bar, but also through call to action buttons, the user will know what to do and where to click.

Logo and the use of Style Tile

The Holidaze-logo is made with simplicity, readability, and professionalism in mind. It is clean and understandable, and contains a hidden meaning in plain sight – the A in Holidaze is made to look like a roof, as this is a reliable firm that provide a roof over your head during your holiday.

To make sure I would use the same elements throughout the website, such as colors, images, text, boxes and buttons, throughout the website, I made a Style Tile. In the following section, I will present some of these elements, and the work and thought I have put in it.

Color Palette

I made a color palette where colors for the user-site, admin-site, and shared colors are displayed. These colors are tested for contrast with the color of the text inside.

On the visitor-site, I wanted to try a few colors that I usually don't see on booking websites.

The purple color is made for all buttons and clickable links. This is done so the user eventually knows that “purple is clickable”. I tried to limit the number of buttons and links to a minimal to make the user click to proceed through the website instead of clicking all around and never getting where I want them; to book.

The blue is for the navigation-bar, and remains the same on all user-sites.

The orange is for HotelSpecific icons and price color to give the site more happy feeling.

The dark gray is for smaller icons such as maxGuests on allHotels and the “H” icon on typeahead.

#B43DBD

#293338

#EC4F1E

#002A5C

The admin-site uses a lighter blue for navigation and other details. This is done as blue is a color more admin-related as it is considered a “tech” color.

#2560AC

#000000

#F1F1F1

#FFFFFF

Icons

In my opinion, the best solution regarding icons for this kind of website is to keep it simple. Flat icons are easily understood by the users, and Font Awesome had some perfect icons for this project. I decided to make use of these as they worked well with my design.

Buttons and Links

As mentioned earlier, all of the buttons and links on the website are purple. The buttons have sharp edges, an aspect that is intended to make the site seem professional and secure. To keep a clean look, the buttons have the same design on every site, except at the search-bar on the landing page.

Boxes

To match the professional appearance of the buttons, the different boxes on the website also contains sharp edges. I have highlighted the boxes by using shadow, making them appear more clickable than without this effect.

Typography

For large headings I decided to go with Proza Libre. This font is used to make the website a bit more welcoming. Although it is..., a lot of sharp-edged boxes and strong colors might get too much for some users, and this round, kind typography helps smooth this out. If the user cannot use this font by any reason, Helvetica Neue is the second choice. Paragraphs is in Open Sans for easy readability of larger chunks of text.

HOLIDAZE

Style Guide

COLORS

Visitor Site



Admin Site



Both Sites

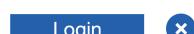


BUTTONS & LINKS

Visitor Site



Admin Site



ICONS



IMAGES



BOXES



SUBMISSION FIELDS

Name	<input type="text"/>
E.g Sunset	<input type="text"/>
<input type="button" value="Search"/>	

TEXT

This is a headline
(Proza Libre or Helvetica Neue)

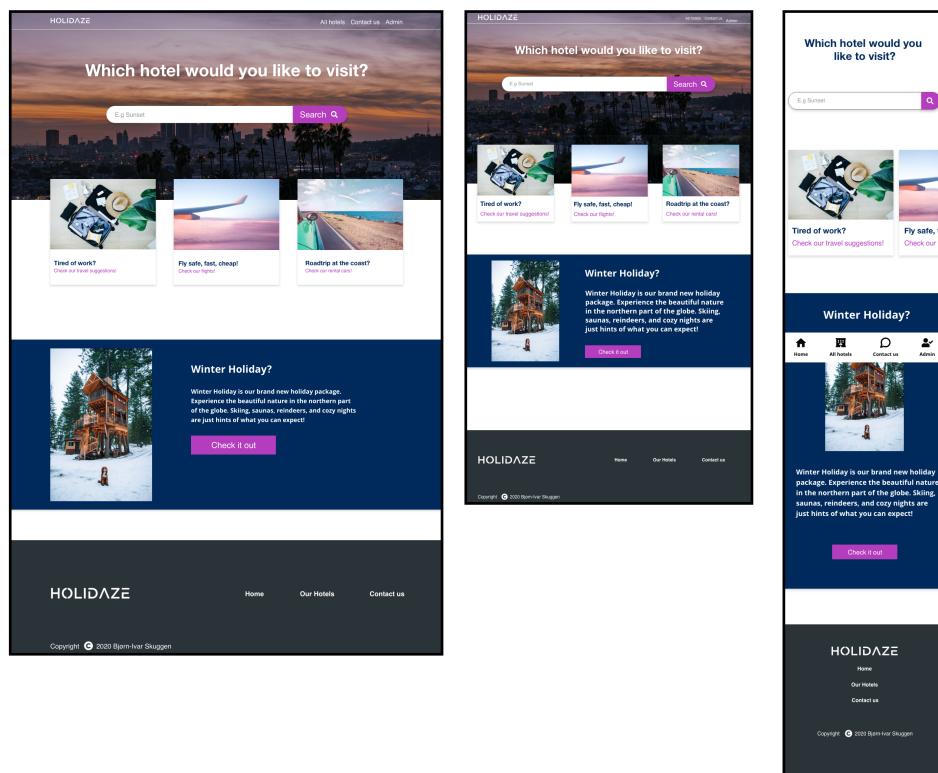
This is paragraph text so when I am using more text,
Open Sans REGULAR is the font I will use for this.
I was not able to get the Regular font for the style
guide, so this is the bold version.



Layout and Prototype

I used Adobe XD to plan and lay out how the website should look without any details. After deciding where the different elements of the page would fit, I continued to fill in the boxes with colors, images, buttons, icons, and text from the Style Tile.

I did two complete designs, and I ended up using the last one, as I found that to be more suitable and looked more authentic than the first one. Further, I created the entire user- and admin-sites in desktop, tablet, and mobile view. I also made them functional and clickable, so the prototype would look and work just like the finished product.



Link to Adobe XD prototype is [here](#).

Development

As the assignment required, React was the JS library used to make this project come true with a XAMPP server to make use of the PHP and JSON files we were going to send and retrieve data from.

HTML / CSS

As this project left almost everything up to each individual, I had the choice of using Bootstrap or other CSS frameworks to simplify the process of creating different elements on the website and make it easier to be responsive. However, I have used Bootstrap enough in my previous projects, and wanted both a challenge, as well as a project in my portfolio that shows that I am not dependent on using frameworks to make things work and look nice on different screens. I therefore made all my HTML and CSS from scratch. The website is based on a 12-column system and Flexbox.

As a CSS-preprocessor I used SaSS, and of course used BEM methodology throughout the website.

Dependencies

During this project, I have tested a few different dependencies or tools to create different parts of the website. Google-maps-react is used for displaying the location of a hotel, React-hook-form to validate forms both for enquiries and for contact form, and simple-react-lightbox for modal of the image on hotelSpecific. These are tools that have simplified some of the work.

Fetching and searching data

To fetch all the data and display the different locations on the website, I used Axios. The reason for this is because Axios performs automatic transforms of JSON data. This means that I can skip a step I would usually have to do with `fetch()`. When finally getting all the results back, I used `handleFiltering` to filter the establishments that contains the value from the search field. This is the technique I applied for both the typeahead and the filtering on `allHotels`.

Posting data to the PHP

To post data such as contact form, enquiries, or creating a new establishment, the choice again fell at Axios. I let the react-hook-form handle the data, and to display errors if something goes wrong. If everything looks good, the data is appended to a FormData object, and posted to a PHP file that creates a JSON object of it.

Handling errors

On forms the react-hook-form library is handling the errors and displaying them if the value does not meet the requirements I have decided. However, there are some errors that had to be done from scratch to get the full experience of validation.

In the enquiry form, the data is validated through react-hooks-form, however, the maxGuest is not validated as this is not something that this site needs to post to the JSON. I therefore created my own validation, so if the hotel has room for 4 guests, and there are 5 people booking in, the error will display, and stop the form from posting.

In the login page, the error is handled based on sessionStorage.

What if a user presses a link that is no longer in use, or types in the wrong url? I have made a simple 404 - Page not found site that shows if the link is broken. There is also the user-navigation so the user easily can press any of those links to get back on the original site.

Issues along the way

All projects have their issues, and this was no exception at all. However, I did expect to have bigger issues that would be very time consuming, but fortunately I only met small bumps in the road. One of the bigger issues I had was that the app refreshed on every form submit after I changed the path in the PHP files. I spent quite some time trying to figure out why, but I never got closure. The fix I found was to put the JSON files inside src instead of the Public folder.

Other small issues were to make the Google Maps work and to display or not display certain information. One issue I discovered in the last second was that sometimes, on mobile view, the simple-react-lightbox did some funny things when opening, making the image pushed to the bottom of the screen. This does not happen every time, and I had no time to look more into this.

Errors

One console error I found while using the website will probably show for you as well. I will justify this here:

```
✖ ► Uncaught TypeError: Cannot read property 'type' of undefined
  at setFieldValue (onloadwff.js:71)
  at HTMLFormElement.formKeydownListener (onloadwff.js:71)
```

onloadwff.js:71

This is an error that will happen if you are using the LastPass Plugin for Chrome. Disable this, and the error will no longer be there. Source for the issue [here](#).

There is also a few warnings related to cookies and DevTools, however, this is out of my reach for now.

What if the servers are down?

If the servers are down, or something is causing the data not to display, there is a safety-net for this. If there is no data, the loading gif will appear, and the search fields will no longer handle filtering or displaying dropdown menus. All forms will not post any data, but will tell the user that something went wrong, and ask the user to try again later. There is not a lot to do on the website if the servers are not running, but at least the user can navigate around without the site crashing.

Extra validation

Even though react-hook-form does a lot of validation by standard, I had to implement some more to make everything work as i wanted it to.

The default regex expression for the email validation did not validate more than test@test, but I wanted it to validate test@test.com and other email addresses with dots, hyphens, etc so the user can not type an invalid email.

Other validations I did was checking if the amount of guests the user types in is higher than the maxGuests, the user has to enter fewer people. You can not pick a date before today as checkin, and the checkout date has to be after the checkin day. This stops the user from booking dates in the present.

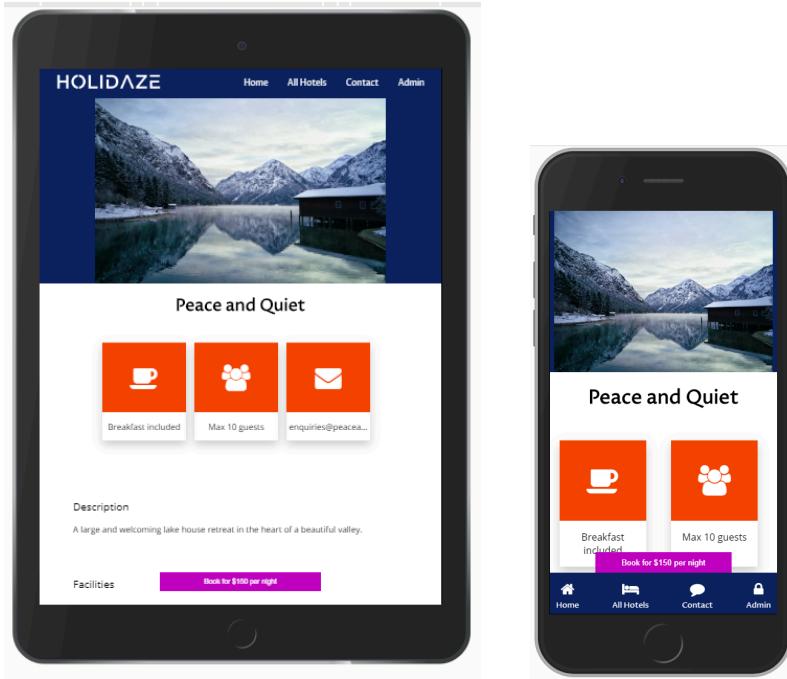
Another feature is that I calculate the total price of the people booking in and days they will stay. This is done just to make the site feel more realistic.

Responsive

With device compatibility in mind, all the different sites should fit to desktop view, tablet view, and mobile view, so you can book your holiday wherever, whenever.



The screenshot shows a dark blue header with the word "HOLIDAZE" on the left and navigation links "Home", "All Hotels", "Contact", and "Admin" on the right. Below the header is a large image of a snowy mountain landscape with a wooden cabin reflected in a calm lake. Underneath the image, the text "Peace and Quiet" is displayed. At the bottom of the page, there is a purple footer bar with three orange boxes containing icons and text: a coffee cup icon with "Breakfast included", a people icon with "Max 10 guests", and an envelope icon with "enquiries@peaceandquiet.no". Below these boxes is a purple button with the text "Book for \$150 per night".



Code review

“

Code review of Bjorn-ivar Skuggen, by Remano Victor.

Bjorn has structured his code in a way that makes it easy to read. His use of BEM makes for an easy flow of code and I can see that should there be an error he is able to quickly identify and fix the bug.

Bjorn displays a clear understanding of React hooks and also displays a clear understanding of the code he writes.

Bjorn has built a website that is both technically sound as well as beautiful to look at. This goes for both design as well as the code. Overall a job really well done

”

User feedback

I was supposed to have more than just one user test, but because of little time, and few people knowing how to set up the project, I ended up with one user testing. Frode is not the normal user. He is a developer, and has experience with creating websites, however, his main field is backend. As he is experienced with using websites and how they should navigate and work, he is often more picky than other users.

“

User feedback by Frode Siem

The website greets you with an inviting homepage. Great choice of color palette. It's easy to start searching for hotels to book, very intuitive. If you don't really know where you are going i liked that you can easily just scroll down and get suggestions to where to go to and how to get there. When ordering a trip i don't enjoy the hassle by finding the best place to go, I just want to travel!

Viewing all the hotels with inviting pictures made it easy to choose. On the hotel page you get all the needed information to know if this is your next holiday trip. Easy, straight forward, and just a few clicks from ordering.

Accessing admin pages was a breeze and with the top menu you easily know where to click to manage the different hotels or enquiries.

Theres no question where to click if you want to contact Holizdaze or even the hotel, I liked that.

“

Conclusion

This project has been interesting. I was a bit terrified at the beginning when trying to grasp everything, and make sense of how I would make it all in time for delivery. I always strive for perfection, and this has a tendency of affecting the use of time throughout projects like this. However, I think I have managed time in a good manner using both my Gantt Chart and Trello board as help, and I feel comfortable with the result I have delivered, both in design, but also functionality. I have tried to follow my designs and Style Tile down to detail, and even though there are a few changes, the result stays very much alike. I think I have written readable and understandable code, and according to the code-review it does seem like other developers can agree. I am happy with the final result, and I feel comfortable that this is a project I can display on my portfolio.

Links and documents

Functional Specification:

https://github.com/bjornivars/ExamBooking_PlanningAndDesign

Trello Board:

<https://trello.com/b/QqtjHEr8/exam-feu2>

Adobe XD prototype:

<https://xd.adobe.com/view/44c758db-2dc8-4e08-615c-0fa2b3bb807a-32f6/>

Error link:

<https://github.com/KillerCodeMonkey/nginx-quill/issues/351>

Github link (old repo):

<https://github.com/bjornivars/holidaze>

Design repo:

https://github.com/bjornivars/ExamBooking_PlanningAndDesign

Github link (new repo):

<https://github.com/bjornivars/newHolidaze>

Stock photo creds

<https://unsplash.com/photos/4iXagiKXn3Y>

<https://unsplash.com/photos/TVIIxFyGaLEA>

<https://unsplash.com/photos/96A9UTFAMUM>

https://unsplash.com/photos/tpe8OBr_jNk

<https://images.unsplash.com/photo-1501426026826-31c667bdf23d?ixlib=rb-1.2.1&auto=format&fit=crop&w=676&q=80>

<https://images.unsplash.com/photo-1517016006573-2eefaa2f5b63?ixlib=rb-1.2.1&ixid=eyJhcHBfaWQiOjEyMDd9&auto=format&fit=crop&w=634&q=80>

<https://images.unsplash.com/photo-1457269449834-928af64c684d?ixlib=rb-1.2.1&ixid=eyJhcHBfaWQiOjEyMDd9&auto=format&fit=crop&w=1267&q=80>

https://upload.wikimedia.org/wikipedia/commons/thumb/8/8a/OSLO-NO-02_05_ubt.jpeg/500px-OSLO-NO-02_05_ubt.jpeg

https://upload.wikimedia.org/wikipedia/commons/thumb/6/62/Oslo_Nationaltheater-1.jpg/250px-Oslo_Nationaltheater-1.jpg

<https://ourwaytours.com/wp-content/uploads/Vigeland.jpg>

[https://tellusdmsmedia.newmindmedia.com/wsimgs/Den-Norske-Opera_Ballett_1_c_VisitOSLO_Thomas-Johannessen_8977006.jpg\[ProductImage\]\[4D037D0DBEC323D15D8FF16649BD\]](https://tellusdmsmedia.newmindmedia.com/wsimgs/Den-Norske-Opera_Ballett_1_c_VisitOSLO_Thomas-Johannessen_8977006.jpg[ProductImage][4D037D0DBEC323D15D8FF16649BD])

Github repository

<https://github.com/bjornivars/newHolidaze>