

Semester Project 2

Monica Kjærvoll

Design

What went well on the project

The design process of the assignment all began with finding inspiration around the web. I wanted to make a grasp of the assignment and to find the right information and inspiration in order to find the best web design solution for my website.

I was determined to make a minimalistic web design, as I found out (after some research) that many e-commercial websites have a minimalistic design and I think it is more appealing to the eye - and last but not least it makes the navigation more user friendly.

The website that I have created is modern and simple. The web design may seem too simple because I did not have much time left to add colors and elements to the page. The most “stand-out” detail on the page is the body background of the site. I chose to implement a textured background (concrete wall) to make the site more appealing to the eye, than just having a plain white background. The texture that I chose for my web design is collected from the site www.transparenttextures.com.

What would you do differently next time

As I began to develop the site using VSCode, I noticed that I spent more time on the design process than the development process of the assignment. I almost forgot how time-consuming it is to develop the actual site by coding. I faced a lot of technical issues along the development process that I lost track of time and had little time left to make the web design more playful and appealing, according to what I had envisioned in my mind. So the only thing that I could do differently next time is to get better at managing my day and time when working with different assignments and to choose the most difficult and time-consuming tasks first before other simpler tasks.



Technical

What went well on the project

Even though I struggle the most with coding with JS, the development process went well at the very beginning of the assignment, due to that I was “familiar” with some of the coding that was used in previous assignments. I also faced some technical solutions along the way, and I was actually proud of myself for not giving up the JS part of the assignment because there were some frustrating hours when I could not understand what I was doing and everything seemed hopeless. I am still a little bit rusty working with JS. By making time and effort into learning more about coding with JS, I know that I get to become a better version of myself as a future frontend developer.

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

The development process of the assignment was a lot harder, as I faced a lot of technical issues when hosting the API locally on my computer. I used a lot of time trying to figure out the different issues I faced and unfortunately, it was time-consuming and affected the development process of the site. After some back and forth I got help from MJPhilip and was told that I had to create my own API by using Heroku and Cloudinary. I have no experience using either of the websites, so I had to get comfortable with each website and figure out how I was going to develop my website by making my own REST API.

After some frustrating hours, I finally figured things out and started to develop my website. The only thing that I knew for sure was that the “original” API contained sneakers as products for the e-commercial site. I then found pictures of different sneakers via Footlocker.com and implemented them in my API via Heroku.

What would you do differently next time

The only thing that I could do differently is make more time and effort into this assignment, due to that I did not have much personal time to make the website look the best possible way according to my preference and what I had envisioned in my mind. Sometimes I struggle to make the ends meet and juggling between my personal life, job, and online studies can be



hard. And sometimes I work best under pressure, but this time I found out that it takes time and effort to make things work. So I have been very disappointed in myself that I did not put more time into making the assignment the best possible way, and fulfill all the criteria of the assignment.

The only thing that I did not have time to develop was the cart page. The final result of the cart page does not fulfill the criteria of the assignment, since it is not nearly finished. I also struggled with making the cart page, as I have minimal experience with coding a cart page.



Herokuapp

Heroku URL = <https://sneaker-lounge-strapi-api.herokuapp.com/>

Login credentials

Username: admin@admin.com

Password: Pass1234

Strapi API

The Strapi API contains two collection types:

1. Products
2. Users

Products

I have created twelve products to be implemented on the website as e-commercial products. Each product in the collection type contains the following six fields:

- Title
- Description
- Price
- Image
- Image_url
- Featured

Users

The collection type users contain only one user with the username admin and the following user information:

- Username: admin
- E-mail: admin@admin.com
- Password: Pass1234

This user is the only user that is confirmed as an authenticated user. This user can log in, add, and edit products on the website.



Sneaker Lounge

Navigation

Each page of the website contains a global nav, navbar, and footer. The main homepage also contains a fourth navigation (highlight nav) where there are three elements to highlight “free shipping on orders > 500 NOK”, “free returns”, and “30 days open purchase”.

Global nav

The global nav contains a nav link element that users can click on to log in to their account.

Navigation bar

The navigation bar contains the following nav link elements:

- Home
- Products
- About
- Contact
- Search bar (redirect click to products.html)
- Heart icon (redirect to wish.html)
- Cart icon (redirect to cart.html)

Footer

The footer contains a four-column section with different elements. The first column contains different customer care solutions, like returns, shipping, etc. The second column contains a list with different links to which users can click and read more about the company. The third column contains a link to job openings in the company.

Index.html

The main homepage contains the following elements/sections:

- Hero banner section
- Featured section
 - Three featured products (redirect click to products.html)
- Brands section (to highlight popular shoe brands the site sells)



Products.html

The products.html page contains the following elements/sections:

- Search input (filter products by brand)
- List of products

Each product contains a redirect link to details.html page by clicking on the title or the image of the product. The products are placed in a card element using Bootstrap CSS. I have also included an add to cart and edit product button under each product. Even though I have added an edit product button on the products page, the only user that can edit and add products on products.html is the authenticated user *admin*. The user has to log in to their account in order to edit and add products.

Details.html

The details.html page contains two columns. I have sectioned the image and details of the product in order to make the page more appealing and user-friendly. The first column contains an image of the product on the details page, and the second column contains the title and description of the product. Under the description field, I have included an add to cart and heart button. Add to cart button will redirect the user to the cart.html page and the heart button will redirect to wishlist.html. Due to time limits, I did not have time to create a wishlist.html for the website.

About.html

The about.html page contains one row with two columns. The first column contains the title of the page H1: "The Company" and H2 title: "Sneaker Lounge". I have placed a lorem ipsum description of the company under the titles. Here users can read more about the company. The second column contains an image of a shoe - this is to emphasize the concept of the e-commercial site.

Contact.html

The contact.html page contains a row with two columns (like the about.html page). I find it more appealing to create a horizontal section, instead of a vertical section. Personally, the



eye flows more naturally when placing elements horizontally, and I find it more “easy” to navigate through the page as you can notice all the information on the page at once.

The left column contains a title and a brief lorem ipsum description where people can read more about how to contact the company. The right column contains a contact form where users can fill in the form with their name, e-mail, subject, and message to contact the company. The form has to be filled correctly in order to be successfully sent, if not an error message will appear to inform the user to fill in the form correctly.

Cart.html

The cart.html page has not been created and does not fulfill the criteria of the assignment.

Login.html

The login.html page contains a form section in which the user has to fill in their username/e-mail and password in order to log in to their profile. As mentioned earlier, the only user that can log in to this page is the *admin*, since it is the only authenticated user on the site. When logging in the user will be redirected to the products.html page. A new global navigation bar will appear when a user is logged in. The global bar contains two elements the user can click on: add product and sign out: name. When clicking on the add product page the user will be redirected to the add.html page and by clicking on the sign out the user will be signed out of the account. By clicking on the sign out a pop-up message will appear to ask you the following question “are you sure you want to sign out?”.

The only technical issue that I faced with this page was that I had to remove the search element in the navigation bar, because the form would not work if it was implemented on the login page. Therefore the login page seems simple and minimal because I had to remove some elements.

Add.html

The add.html has an identical design to the login page. Here the user can add a product to the products page with the title, price, description, and image URL. All the inputs have to be correctly filled in to add a new product - if not an error message will appear.



I also faced the same technical issue as the login page. The search element in the navigation bar had to be removed in order to submit the form.

Edit.html

The edit.html page has an identical design to the login.html and add.html. The form contains a title and four inputs where users can edit the product that they have clicked on from the products page. On this page, the user can either edit the product or delete the product from the API.

Here I also had to remove the search element in the navigation bar in order to edit or delete the product.

