Lappeenrannan teknillinen yliopisto
School of Business and Management

Software Development Skills

Marta Xiaoyang Moraga Hernández, 001101267

LEARNING DIARY, FRONT-END 2022-23 MODULE

27.09.2022

I accessed the course for the first time and read the Course Overview, General Course Information and the Environment Setup tabs. Briefly, I looked over the other tabs as well.

I decided to work on VS Code on Ubuntu, as I have already done some basic version control there.

The diary is being written on google docs, then being exported and transferred to the project as a pdf before important commits.

24.11.2022

Using GitHub Docs and the tutorial on the course I set up the repository.

Part 1

To follow the instructions from the video, I installed nodejs, using the instructions provided on the Advanced Web Applications course.

25.11.2022

In the last step of the video they create the .gitignore file before initialising the repository. Since I already had the repository ready and I was working in the Coursework directory inside the already created Front-End repository I created .gitignore inside Front-End, and added the route *Coursework/node_modules* inside.

I struggled a bit because I made a spelling mistake, so node_modules ended up on the github repository anyway. Therefore, I deleted it and did several test commits to check for the problem.

Part 2

I pondered for a moment if I should use my name and data to fill the portfolio example. I decided I will use the data from the tutorial, because in the future it might link things like twitter or linkedin, or add images, and I lack those.

To use the icons I signed up for fontawesome.com.

So far, scss is quite similar to css. Variables, nesting and functions truly facilitate styling.

Sometimes, when trying to use a variable or id, messing up the spelling is quite easy, so I downloaded Code Spell Checker extension to help with that.

As I assume everything on the tutorial is going to work I am not committing very often.

Part 3

This part of the tutorial was quite simple, just following along. Most of the difficulties were differences in class names. For example, I named the navItems class as ‘nav-items’, while the tutorial mentions that they name it ‘nav-item’ in singular.

I had the same problem with sass as the tutorial and the solution was the same Ctrl+C and run the script again.

Part 4

I had some trouble with translate3d, but it was quite easy to find the problem, I mixed up the values.

As I continued with the videos, I started to see there are lots of options for the front-end. With clever use of css and js many things are possible.

Using the option to simulate different mobile screen sizes when inspecting a web on Mozilla I found that on slim smartphone screens a scroll bar does appear. Scrolling down it is possible to see the end of the transparent grey box that should cover the image.

Adding *overflow: hidden* to the body on the main.scss or changing the height from 100% to 100vh does solve the issue. Both with the problem that some parts of the web will not show because the screen is too small, however, I suppose this is expected behaviour, as a couple of times in the tutorial *overflow: hidden* is used. Also, to be fair, screens of this size are unlikely and this is only for the sake of experimenting.

Part 5

I reverted the 100vh back to 100% and followed the tutorial. Everything worked as intended.

When adding a comment to the commit I mistakenly wrote “Part 4 II: About page” instead of “Part 5 II: About page”. Checking GitHub docs it seemed that if the commit was already online they do not recommend force pushing an amendment. The reasons why did not apply (no one had cloned my repository), therefore, after a merge (I tried to fix the last commit not remembering it was already online), I decided to just let it be.

Part 6

Doing the work page was pretty quick as we had been using grids for the other pages.

Flex seems like it is better at automatically adapting to the display, as _mobile.scss did not have to be touched.

Part 7

As the exercises material is inside Coursework, the steps from the tutorial did not work. If, when I try to deploy my project, it does not work, I will come back to it. In the meantime, it seems like a normal result.

Project

Starting:

Following again what was done in the first video, I established the folder structure for the project. Everything worked as expected.

This time, the steps from the last video worked as intended and the website was deployed at https://martaxm.github.io/Front-End/. Obviously empty at the moment I was writing this.

Website plan:

Before starting, I decided on the theme: an aquarium website. When I signed up for the course I already had some topics in mind (maybe a museum or a history website) and some ideas.

For an aquarium, I thought that it would be cool to scroll down and see the background change to reflect a deep ocean. After seeing the Basic Parallax Website With HTML & CSS video that was linked on the front-end course page, it was decided.

The site would have a similar menu to the one in the tutorial, but on wide screens it would have a fixed navigation bar.

There would be 4 pages:

- Home: a bit about the place and photos
- Exhibition: about the exhibition and kind of animals you can find (the parallax design)
- FAQs: Frequently Asked Questions.
- Tickets: see tickets prices and timetable

All images are from https://www.pexels.com/ website shown on the Parallax tutorial.

Working on it:

I started out with the structure, to have something on there I copied a lot of the code from the example. I will delete everything that does not get used in the future.

26.11.2022

For the fixed navigation bar, searching online I found a short tutorial https://www.w3schools.com/howto/howto_css_fixed_menu.asp, which I implemented.

When on media XL or mediaLg, the links are visible and the menu button stays hidden. While in any other case, the links of the fixed navbar disappear and the menu button becomes visible.

The navbar's display is set to flex. I had some problems adjusting where everything was, but after fiddling with the font size and the margins I got it to look how I wanted.

I made the title a link, because it is very intuitive that the title brings you back to the home page.

After a very basic home page I started working on the exhibition page with the parallax video.

I changed a couple of things to adapt it to the page, like the colours or images. Also, each section has a colour from lightest to darkest. I contained the parallax on main id="exhibition" and set its z-index to -1, so that the navigation bar showed over it.

_navbar.scss was created for better organisation.

Next I did the tickets page using a simple grid and committed. There were not any major problems apart from the padding and margins that I had to adjust until it looked good enough.

Before starting the frequently asked question page I had to fix the nav-btn and the navbar links. They were not working properly. After a while I realised that their z-index was not set correctly and I changed it. I also had to change the z-index of the close button and change the nav button position to fixed.

While working with divs I had some trouble.
https://stackoverflow.com/questions/5703552/how-can-i-center-text-horizontally-and-vertically-inside-a-div-block the 2020 update of the first answer was useful when doing the faqs page.