Lappeenrannan teknillinen yliopisto
LUT university – School of Business and Management

Software Development Skills: Full-Stack, Online course

Martin Bárta, 002525033

LEARNING DIARY, COURSE OVERVIEW

Date : 26-28.11.2024
Activity : Moodle page

I first checked out the General Course Information tab to understand what the learning outcome and requirements of the course will be. I looked through all the tabs and read what it is about. Since I already done Introduction to DevOps course and Front-End Software Development course, I understood the template for completing this course will be similar. I already had some experience with databases, connection strings and the overall functionality, I knew about different HTTP header methods and stuff used in REST API (just really basics, but even that makes it easier to understand when you are following a video tutorial at quite a fast pace). I needed to learn more about React and Redux, since I didn't have any practical experience with that beforehand. I also needed to learn more about node.js and express, but the overall functionality made sense after some time.

LEARNING DIARY, IDE SETUP

Date : 26-28.11.2024
Activity : Video tutorial

I started following the MERN stack tutorial and I wanted to grasp what will need to be done first. I immediately created a new GitHub repository, and I cloned the repo to my laptop. I prefer to use Visual Code Studio so I already had that setup and I am mostly familiar with git, so this part was without a problem.

LEARNING DIARY, MERN stack

Date : 26.11.2024 – 3.11.2024
Activity : Video Tutorial

After my IDE and Git were set up, I could start following the video and do the things myself. In the first part I reminded myself about REST APIs and how they work. The base folder structure was created at this part, and I started working on my backend code. Since the whole video tutorial is long and full of many different information, I will point out the most interesting parts for me and what I learned there. The first interesting thing that I learned to use was the Postman. I knew that it exists and that it is somehow used for calling the API and getting responses and I have seen some API communication using swagger on some of our applications at work, so it made sense pretty quickly and I was glad I could try it. It is a useful program and also simple to use, I quickly tried the created GET, POST etc. methods and I could verify that the created goal routes are functional. Also, the use of Bearer token and authentication in general later was a welcomed new thing.

Another thing that was explained well and was nice to learn was the use of environmental variables. The creation of controllers was pretty interesting as well and I think that Brad explained it really well, since it made sense, and I could follow what was being done. The error middleware was a tricky part to grasp for me, because I understand what is being done, but it took me some time and thinking. The MongoDB creation was a fun part and even though the process was a bit different for me due to the video being a bit older, there was no bigger issue. I also liked that I could connect to the database through Mongo Compass, because I sometimes struggled with picturing the tables/collections. The Schema creation was a bit harder at first to understand, but it made sense eventually.

In the JWT part I understood what was being done, but since it is more like a “third-party” coding and also its brand new for me, it was quite a hard part. The logic - that user needs to be verified, and I want to see the goals only of the logged in user makes sense though, so it was ok in the end.

The frontend part was interesting to see, especially how much “easier” it is to create a nice-looking frontend with those frameworks, than just using HTML and CSS and some JS like I did before. It is more complicated at first (and it still is now, for me), but I see how it can be really good and useful when you have more experience with React etc. After doing all the pages and components I encountered some problems though, because I couldn’t login after registration and the whole app kept crashing sometimes or loading indefinitely and so on. After some troubleshooting, I saw there is a network issue, and the proxy wasn’t really working. If I added the proxy to package json, the frontend wouldn’t compile properly, and I didn’t really know what to do with it. It took me some time to troubleshoot it and finally the problem was cleared when I added DANGEROUSLY_DISABLE_HOST_CHECK=true to the frontend package start script. At this point I realised how important are frequent commits and comments and those “best practices”, because as the project was much more complex than my previous projects, I got lost in troubleshooting really quick and I didn’t know what I changed, tried and what the

problem might me after a while. Like I said it eventually worked out and after testing I could move on to the last part.
In the last part the form and items were created and that went pretty well at first. I encountered a problem with js index at one point and it wouldn't communicate properly and I wasn't able to logout from accounts and it would get stuck etc. I found the problem though and it worked in the end. I had also problems when trying to deploy to Heroku, because I tried to do it first the way like Brad in the tutorial with Heroku CLI. It was more complicated in my opinion than just connecting GitHub account and deploying it that way.

Also I encountered a database problem, that would lag and loop my app and I couldn't communicate with the DB. This problem was because of whitelisted IP addresses and even though mine was there, it just didn't work. After I allowed connection from all IP addresses in MongoDB Atlas, it worked. This isn't a good practice probably, but I think that for the purpose of this course it is not a problem. After this fix, the app was working and was deployed on Heroku.

After this I wanted to add some additional functionality as part of the assignment. I wanted to add flagging and edit the goal "tile" visual, because it looked really bad in my opinion. The flagging was added with some updated frontend as the last part of this project. After this I deployed it again and did the video demonstration and uninstalled the app, because I wasn't sure how much am I being billed for it. I hope it doesn't have to be up for the assessment, but I can deploy it in that case anytime.