Lappeenrannan teknillinen yliopisto

School of Business and Management

Sofware Development Skills

Sirpa Karhu, 2095573

LEARNING DIARY, FULL-STACK MODULE

**LEARNING DIARY**

15.6.2022  
  
I familiarized myself with the course. I am eager to get started with the course as I wish to develop my skills in software development. It is nice that there are first exercise projects as a “warm up” and then I can learn more and be creative with my own project. I chose Git for my version control. I have used it before this very little, so it is very useful for me to learn to use it more with a “real” coding project. I had previously used Gitlab as a repository hosting service but this time I decided to try Github. For my code editor for this course I have chosen VS Code as I have already used it so it’s easy to get started with that. I had previously already installed some extensions to my Visual Studio. From the link provided in the course material I realized it would be useful to get to know more with the extensions as there seems to be many extensions that make the developers work easier and faster.

I had to do some revision in basic functions in using Git. I had some problems with making a first commit and pushing it to Github. One of the problems seemed to be that my branch was named “master” locally but “main” in Github. I learned that “main” is nowadays the recommended branch name so I changed the local repository to main. I somehow struggled also otherwise in pushing but finally with the step by step advises from the documents in github site it finally succeeded.

Before starting to learn more about Node, I studied little bit (as recommended in the beginning of the tutorial video) about the http response status codes and headers, MVC ( Model-View-Controller) design pattern and REST API’s.

20.6.2022

I started to learn about what Node is through the node.js -youtube courses / tutorials. I learned that Node is an Javascript runtime. Through node, we can use Javascript as a serverside language. Node is fast and efficient and it can be used almost with anything that is not CPU intensive.

Through the tutorials I learned for example that:

* through Node REPL (Read-Eval-Print-Loop) it is possible to run JS through terminal.
* NPM (node package manager) is essential tool for running Node.js. It provides countless open source packages, which can be installed and interacted in node.js project through npm cli.
* Node module types are: core modules, local modules and third party modules.
  + Node core modules are for example: http, url, os, querystring, path, fs, util, event
* nodemon is a dev dependency tool that automatically restarts the node application when the file is changed
* a simple HTTP server can be started with Node.js
* through Heroku it is possible to deploy an application to the internet

First the syntax of Node.js seemed little difficult for me, especially as I haven’t much used serverside languages. There seems to be anyway good documentations available and after a while it felt more clear.

I did not have any bigger problems in accomplishing the exercises in the tutorials. Of course there were many typo problems but they were easily solved.

28.6.2022

I had problems in pushing to Git remote repository. I did not manage to push all of the contents of my git folder to remote repository, a subfolder could not be opened and the contents were not accessible in remote repository. I searched help from stackoverflow and other internet sources and tried many things. Apparently the problem was that I had a .git folder higher up in the folder hierarchy. After removing that folder I could finally successfully push all of the contents to github.

29.6.2022

I watched the MongoDB crash course videos. I was unfamiliar with MongoDB before this but it was quite easy to catch up with it through the videos. I learned that MongoDB is a no-SQL database. It has advantages compared to relational databases, it is for example easy to scale and faster. The data is stored in collections. JSON -like syntax is used with MongoDB.

I followed the tutorial and successfully created, read, updated and deleted the data in my own Mongo database.

3.7.2022

I started watching the first Express crash course video and doing the exercise project. I learned that Express is a minimalistic web framework for node. It is a server-side framework. It is fast and most popular node framework

Compared to creating a server with pure node.js in the node crash course, with express it seemed that much less code was needed to build a server and web application.

Totally new things for me were for example middlewares, creating routes, express router function, and template engine express-handlebars.

I had couple of problems with implementing the express application (members app).

I could not register the handlebars view engine with the code provided in the video. From stackoverflow I found an instruction to update the path app.engine("handlebars", exphbs({ defaultLayout: "main" })) to app.engine("handlebars", exphbs.engine({ defaultLayout: "main" }));. By doing as advised there, solved the problem.

I also had problems with adding members through the browser, after confirming that adding members succeeded through postman, I realized the problem was in index.habdlebars -file. The problem was one extra quotation mark around POST in HTML-code.

In the project an API was build, where it was possible to create, update and delete members and render views.

5.7.2022

In the older Express tutorial same kind of application was build like in the newer tutorial. A database (mongo) was also used here, so it was a bit more like a “real” application compared to the application build in the newer video. It was also good rehearsal with Mongo here.

The template engine used in this project was called “ejs”. Different kind of syntax was used compared to handlebars but otherwise it seemed to work somehow same way.

I ran into a problem that I could not get the ajax call to work and get the users deleted from the database in Firefox browser, in Chrome it worked fine. In the video tutorial’s comments someone else had the same problem but there were not any solutions proposed. Unfortunately I did not manage to find a solution to this.

In the index.ejs -file I had problems to get it working, problem seemed to be in the syntax. Through stackoverflow thread I found out that I need to change the code a bit in “include-call”. I am not sure if the syntax used in the video was deprecated or if the problem was something else. After changing the line I got it working anyway.

6.7.2022

I started to study Angular, which was totally new to me. I started going through the Angular “Tour of Heroes” -tutorial but found the Angular syntax difficult so instead started to study first the syntax a bit for example through w3schools tutorials. I learned about the use of modules, directives, data binding, controllers, scopes, filters, http and services in Angular. I learned that Angular is an open-source front-end framework and is typically used to build single-page web applications. I also tried to learn little bit more about TypeScript, as I had not used it before, and TypeScript seems to be used with Angular mostly.

13.7.2022

I have gone through the Angular Tour of Heroes and got the application working coding along with the tutorial. I learned to use Angular command line interface, which is a tool create angular applications and which helps to generate and configure components, services, dependencies etc.

I am still struggling quite a bit with learning Angular. I find it complicated to put pieces together with Angular, for example file structures are hard to perceive for me.

18.7.2022

I started to watch the MEAN Stack series and learned from the first video that this project will be covering for example topics like Node.js & Express REST API, Passport,js, MongoDB and Mongoose, JWT Authentication, Angular 2 and Angular-CLI, Angular2-JWT, Angular 2 Flash Messages and MEAN Deployment. It is nice to get to put thing together and make one project, where all technologies that have been studied in this course will be used. Hopefully this will make things more clear to me.

In the second video it was advised on how to start build an authentication app back-end using node and express. There was useful repetition eg. on setting up Express server and creating routes. Connection to MongoDB database was made with Mongoose module. Also middleware for the required dependencies were set up (cors, body parser..).

20.7.2022

In the third MEAN stack video an user module was set up, mongoose Schema was created and back-end registration functionality was build. Password encryption was implemented with NodeJS bcrypt library.