Lappeenrannan teknillinen yliopisto

School of Business and Management

Sofware Development Skills

Olli Saaristo, \*\*\*\*\*\*\*\*\*\*

LEARNING DIARY,

Anytime-course: Software Development Skills: Full-Stack 2020-21 MODULE

LEARNING DIARY

30.11.2020

(link to my GitHub) <https://github.com/Osaaris/FITech-Full-Stack-LUT-2020>

**Node.js**

I was familiar with Git and many text editors, but I decided to use good old VS Code, because I'm most experienced with it. I prefer VS Code over Atom, Notepad++, Editra, SublimeText, etc. because it's simple and easy to use and has a lot of great extensions. I have also decided to use GitHub over Bitbucket, mainly because I want to keep my repositories in one place.

I learned a lot of new things today, despite the fact that I've done Full Stack previously. For instance, the fact that you can write node codes directly to CMD was new to me. I really like the choice of this course's [YouTube video tutorials](https://www.youtube.com/watch?v=fBNz5xF-Kx4); they are much better quality when compared to some other tutorials I've watched.

shortcuts for VS Code I learned from the video:

* Toggle line comment //
  + Ctrl+'
  + Ctrl+K+C (undo Ctrl+K+U)

I created Heroku user, but I'll rather use my GitHub to store the codes.

I watched the video entirely, and learned a lot of new information about how the node modules work and how to create them as on your own. All I can say is that thank god we don't have to create them, when we can download them with npm.

30.11.2020

**MongoDB**

I learned few new things about MongoDB and installed MongoDB Compass.

I copy and pasted the useful commands from GitHub document to my local notes.

<https://gist.github.com/bradtraversy/f407d642bdc3b31681bc7e56d95485b6>

I have also enrolled to another course by another organization, and I have access to FrontEnd Masters web-site. I have watched similar videos there.

1.12.2020

**Angular**

<https://angular.io/tutorial>

This is my first time when I'm using Angular. The CLI was easy to install. I haven't worked with .ts files before and the syntax is different I've used to.

Following the tutorial (Tour of Heroes) is a nice way to get familiar with Angular.

I had some difficulties with tutorial, mostly because I put the code parts into wrong section. Although Angular seems powerful tool, I'm not very good to use it, yet at least.

Creating components with CLI seems handy and useful feature.

All of the commands used in CLI were new to me (with Angular).

Most important ones are

ng serve --open

ng serve

ng generate service ####

ng generate component

After completing the tutorial I can say that it's not my favorite tool... But maybe I'm not seeing the full picture of its benefits. I read an article that said it's good when making dynamical websites, and it's from Google, which is a plus.

The tutorial itself was nice, but there's so much new information to me that I may have to read it again at some point.

2.12.2020

**ExpressJS**

<https://www.youtube.com/watch?v=L72fhGm1tfE>

Dealing with APIs is not my strong suit, but I learned new information about the structure and how to make API calls. Backend in general is something I need to study more.

What did I learn from the video? First, I learned couple shortcut tricks in VS Code, second I learned more about how the folder and file structure should be done.

I also learned a bit more of npm libraries and got good practice.

I struggle with API things in general, so it's something I need to work on.

*Course goal*

*This course aims to give you an edge in the jobmarket by providing tools for creating unique projects and to help you find your passion as a software developer.*

*This is the full-stack course.*

*You can find the back-end, mobile (android) or front-end development courses in the Moodle by searching "Software Development Skills" or by their topics ("back-end" etc.) The courses will likely be at the bottom of the search results.*

*The course has 3 mandatory assignments for a student to pass this course.*

*1. Material from the exercise projects that you perform, following the tutorial series (Reflect your learning in the diary)*

*2. Learning Diary to keep track and reflect your learning. (Template found at the bottom of this page)*

*3. Project to demonstrate how to use the course material in action.*

*How to actually complete the course?*

*1. Get yourself familiar with Git and choose a code-editor*

*2. Follow the steps provided in this Moodle course*

*3. Remember to commit your work often, so it's easier to keep track of your work.*

*4. It's recommended to commit your coursework in a different directory in your git-repository apart from your project, label the directory ex. Coursework*

*5. Remember to write your learning diary side by side with every time you work with course material, this is very highly recommended!*

*6. Return your work to Moodle.*

*7. Go to the "Course completion" tab and select that you want your course to be graded.*

*How to receive a grade?*

*Have you completed a module and want it to be graded? Go to the "Course completion" tab and select that you want your course to be graded. That will start the grading process for this course and your grade will be delivered to the study office.*

*How does the evaluation process work?*

*You will submit a file to moodle containing a link to your public git repository.*

*In your git repository there should be the following:*

*1. Material from your exercise projects*

*2. Learning Diary*

*3. Your Project*

*4. ReadME how to run your project*

*5. File including a link to a video of your project running*

*It is optional to explain the project in the video or add sound. The video can be uploaded to YouTube and captured with OBS studio for example.*

*Note that you are granted a permission to reuse any code given in the webcourse example project.*

*You are also allowed to use code from the internet, just remember to comment in the code ex. a stackoverflow thread where the used code was from.*

*Help from friends is welcome, but when you solve code problems by yourself you learn a lot more.*

*Also in worklife it's a huge bonus to be able to solve problems independently.*

*Still asking for help is recommended if you feel like you're unable to solve your current problem.*

*Using tutorials as a source for learning*

*1. Tutorials are just one way to address the given problems. They are NOT the only single solution to handle the presented problem.*

*2. Note, that some tutorials may have outdated information. The core learning objectives are still valid.*

*3. If there's problems with tutorials with outdated libraries please post a question to course community.*

*4. Don't just copy and paste code blocks. Try to understand the reason behind decisions.*

*Information about the moodle-course Project*

*1. The project needs to differ from the example project, but it still can be an extended version of the given example project.*

*The difference must be something, that fits the definition: "your project".*

*2. You need to fulfill the specific project needs given in each module*

*You may add some other technologies in addition.*

*3. Remember to make your project code as clean as possible.*

*4. Try to avoid bad variable naming and generally "bubblegum solutions".*