Lappeenrannan teknillinen yliopisto
School of Business and Management

Software Development Skills

Johannes Hölker, Studentnumber: 609983

LEARNING DIARY
SOFTWARE DEVELOPMENT SKILLS
FULL STACK
MODULE

Contents

1 20.10.2020

Dear Diary,
After reading the introduction of the course, I replicated the Word Template for the diary in LaTeXusing Texmaker as a GUI for writing. I managed finding my old Github Account then and installed Git on my Ubuntu system. I decided using Fork for my version control because I used it before and I am familiar with the program.

2 22.10.2020

I wanted to download Fork but I realised that Fork (and SourceTree) don't have a Linux Version. During watching the Introduction to Version Control I saw that the code editor Atom has an implemented version control GUI, so I decided to take that one.
Afterwards I watched the video about Node.js. Because I planned improving my Python skills this semester I realised that the course is made for JavaScript, which I didn't use before so it is a whole new challenge for me. Regarding the point that JavaScript is useful for sending/getting data from a server, I am looking forward to it.
I already have an idea for a JS application. With a friend of mine I was planning to make a simple ToDo-list application using the Python. For storing (and checking) the entries of the list on a server, I am thinking about using JavaScript.
Now I am setting up my Git-Repository and placing the Texmaker-Workspace in there. I tested the first commit of the folder SD_FS in the command line and the second commit directly using the extension of Atom. During this I am getting familiar with the code editor Atom and the Github implementation in it.

3 26.10.2020

During watching the video about Node.js I am creating my Exercises Workspace and the Project Workspace in my project folder. After I tried isntalling Node.js by using the tar.xz folder, which i didn't managed, I nstalled it using the command:
sudo apt install nodejs
After doing so I realised that I installe the version 10.19. So I deinstalled the program and installed it using PPA. It didn't work either. So I used the version manager by tyoing the following commands:
$curl-sLhttps://raw.githubusercontent.com/creationix/nvm/v0.31.0\,install.sh-oinstall_nvm.sh$
$bashinstall_nvm.sh$
nvm install 12.19.0
And setting the version to default.

4 27.10.2020

My environment during the Node.js Crashcourse contains now Atom and my Linux Terminal and it works all fine while the exercise till I wanted to use quotes in a String. The quotes with the dollarsign were not recognised as a quote. I started looking it uo and also started the JavaScript course on Codecadamy for learning the Basics of JS.
I figured out that I used the wrong quotation marks. Using the ones from the video my quotes are working fine.
In the end of the Youtube video I made myself a Heroku account and got my first Website online. It was really exciting and I am looking forward to my project now. You can see the result following this link:
https://blooming-fjord-46718.herokuapp.com/
After finishing the video I made a Commit on Github. Here I had the problem that the subfolder of SD FS was not committed. I looked it up.

5 31.10.2020

In the git documentary I saw that you need submodules for committing to two public repositories. So I decided to avoid these circumstances by making two project folders. One for Heroku and one for Github. I made myself a more serious Github account and seperated the folder for Heroku from it. Before committing I am making copies of the Heroku project into the Github project for sharing the code. I will also exclude the LaTeXcontent and will just make commits with the Diary pdf file in it.
After I setted up my new project folder I started the video about MongoDB. I installed the Linux version and eberything worked fine till I wanted o use MongoDB compass. It said to me the connection to the default server was refused. I shutted down the computer and quitted for today.

6 01.11.20

After the startup of my computer I tried the same thing as the previous day and eventually MongoDB was running. I didn't make any change. So I can continue with the MongoDB crash course with setting up the databases.

7 02.11.20

During the MongoDB Crash Course I experienced some very useful tools which are perfect for a ToDo list application that I thought I can build. In the end of the video I saw that Brad Traversy did a To Do entry and I definitly thought about doing this in my application.
I took a small look into the Example project Video series and I realised what the application should be about. I planned to do the example project too for

getting a Login page and then to implement a To Do list for every user.
It is especially really useful to store the ToDo entries in a MongoDB server for accessing and changing them from several devices.
Short after I started the Express Crash Course and setting a local server I made a commit on Github with all the tutorial stuff.
I continued with the video and everything becomes clearer for me. Especially the part of Express that you can just make .html files in one folder and a new page is opening amazes me.

8 03.11.20

I continued with the Express Crash Course and everything worked fine. After finishing the video I made a commit of my work and pushed it to Github.
I am now thinking about my own project and I searched for some inspiration in an Ted interview of Linus Torvalds. He says that his purpose is to make applications he needs. Here is therefore a collection of ideas of a garden/plant organizer. The functions that the user should be able to do on the website are:

- creating and logging into an account
- accessing a list of vegetables which is public and which he can extend
- adding vegetables from the public list to hiw own plant list
- creating hiw own list of vegetables he wants to plant next
- receiving a list of activities (like a ToDo-list) he has to do next
- also the weather should be displayed on the front page/Dashboard

By using the given Tutorials the application should be able to do the following:

- storing the entries for vegetables on a seperate server (MongoDB)
- the user profile should be protected with a password
- the Dashboard should change depending on the weather (also the weather of the last days) and the special season (written in JavaScript)
- having a handy and easy to use surface (Angular)

The vegetable Object should have the following:

- Caption
- description
- harvest time
- plant time
- pouring frequency
- optional Notes

9 04.11.20

I started the Angular Tutorial. While setting up my environment I decided to directly code in the git controlled dircetory this time. I setted up my environment again while using Atom (which I got failiar with and with which I am making my Commits and Pushes) and my bash Terminal. I got the default application from angular prompted in my browser.
I faced a problem while in the first section. I was not able to refresh the browser that it showed the name of the hero. I restarted the whole Tutorial project and tried it several times while I tried to understand what the Angular framework created there. I understood that it used Node.js and it wants to simplify the creation of a good looking website but I needed some time to get what the Initialisation created.
One thing that I experienced was that the .css files changed to .scss files. During the whole project up for one more time I realised that I chose the SCSS option in the initialisation.
In the third try the application of the first chapter of the Angular Tutorial everything worked fine. I had to restart the "serve" command because before restarting the input was not displayed above. I couldn't figure out why. After finishing the first chapter I made a Commit of my work.

10 05.11.20

I continued with working on the Angular Tutorial now doing the fourth chapter about how to add a service. While doing this I checked (way too late) what the language TypeScript is. I experienced like I guessed before that it is nearly the same a Java Script and that every JavaScript file can be compiled as a .ts file. During the process I made a safety Commit.