Working as a software developer at KPMG Finland

Diary Thesis

Anh Nguyen

Abstract

15.01.2021

Author(s)	
Anh Nguyen	

Degree programme	
Business Information Technology	

Report/thesis title	Number of pages and appendix pages
Working as a software developer at KPMG Finland	**60**

The main purpose of this thesis is to document the daily responsibilities and the development progress of a junior software engineer at KPMG Finland. The working period starts from 26.10.2020 to 1.1.2021.

The diary covers a period of 10 working weeks. Each weekly report includes 2 parts: daily observation and weekly evaluation. The daily will describe the main tasks and events that happened in a day, while the week evaluation analyses deeper new skills in the author's professional development progress. The technology will be diverse from Java Spring Boot, Node Js to Angular or React.

This thesis would be useful for undergraduates or anyone who inquire into what a junior software engineer does at work in general and in a global big four organization specifically. Therefore, they can well prepare while still at university and improve what skills are demanded in a working environment.

Keywords

Software engineering, backend development, docker, Jenkins, java, spring boot, KPMG, big four, React hook, typescript.

Table of contents

Table of Figures

1   Introduction

The thesis was written in the form of a working diary. The observation period started from the 26th of October 2020 to the 1st of January 2021. The main purpose of the thesis is to document the daily responsibilities and the development progress of a junior software engineer at KPMG Finland.

KPMG Finland is a member firm of KPMG International - one of the world's leading expert organizations. It is a global network of professional service firms providing audit and assurance, tax and legal, and advisory services. I am a part of the Application Technologies team, in the KPMG Lighthouse unit. KPMG Lighthouse unit develops intelligent solutions to help its customers utilize data efficiently and reliably. The solutions can be ready-made platforms from KPMG or their partners, or they can create a solution with their application development team to serve the customer's needs.

As being a software developer in this team, my tasks are various from the backend to frontend to meet the diverse demands from the customers. Depending on different products with different technologies' requests, the development team works with different tech stacks: Java Spring Boot or Node JavaScript in the backend, React or Angular in the frontend, and also Python Django/Flask.

On account of the extremely strict data privacy policy in KPMG, I am not allowed to discuss many of the business, detailed service solutions that I am working on, any piece of code or screenshots of the products are also not included. I will be mainly reviewing my daily tasks and further analyzing new technologies that I have been learning at work. It helps me to explore and have a chance to immensely understand the technologies that I quickly pick up at the working time.

My team is called Application Technologies, consisting of 12 developers working interactively on 8 projects. In one sprint, we divide into small groups and the group's members are changed on every sprint. So normally, 2-4 people will work on a project at once, and in a sprint, it is approximately 4 out of 8 projects are active. This system is quite new to me at first, however, I quickly adapted and got the advantage of working in this way. It allows me to work with different people,

and different technologies so I can improve a wide range of tech stacks and be more flexible.

The framework used daily in my working life includes:
1. Locally installed software: Java Development Kit for Java 8, Java Development Kit for Java 10, Eclipse (Java IDE), Python (2.7, 3.7), Visual Studio Code (Frontend IDE), Git, MySQL Workbench, Ansible (2.9.7), Vagrant (2.2.10), Docker (19.03.13)
2. Tools and services: Bitbucket, Jenkins, docker
3. Technologies and frameworks: Spring Boot Framework, Maven, Hibernate, Angular 9, React, Django

Most of the researches and studies that I have analyzed in this thesis are a combination of various sources for different purposes. The most significant sources for theoretical information are from the documentation pages (such as https://www.baeldung.com/ for Spring community, https://reactjs.org/docs/ for React), Medium or Hacker Noon blog posts. When I need to understand a concept thoroughly or know-how of a new framework, I learn from these sources about the general information, best practices, and practical advice. When I faced some bugs, it can be from the npm package, or my mistake, I often get references from Stack Overflow, which is a professional online community for developers. It is a place where other developers ask questions and get answers from other developers or just some discussions. For me, there are already the answers to most of the questions I asked. It can save a lot of my debugging time and effort.

About the necessary skills, the most essential skill that I need to prepare for this job is a solid knowledge of web technologies both frontend and server-side web-programming, strong programming skills in Java, JavaScript, REST API, and a general understanding of Cloud, Docker container. Those skills also include using the related frameworks, for example as Java, we use Spring Boot and Maven; for JavaScript, we use NodeJS in backend and React TypeScript in frontend.

I also learned more frameworks for testing during my working time. I have been writing and using unit tests, integration tests, and system tests such as JUnit and Mockito for Java, and Jest, Mocha, or Jasmine for JavaScript.

Besides programming and testing skills, Git version control is also an important skill that I need to deeply understand for this job. The most important and tricky part of using Git is solving conflicts. Git is a technique to control the working flow of a team, therefore, when 2 commits modify the same line of code, it is essential to analyze which part of code that should be selected. However, I have been using git for a year, I have become more confident to use Git command now.

Apart from those skills above, I am also familiar with Docker during my working period. I did not configure the container, but I have been working with it, therefore I was able to pick up some useful tips. In general, my work tasks always differ, so that I need to learn a lot of new skills during the development time.

1.1 Key professional concepts

Spring Boot	Java framework
Java	A class-based, object-oriented programming language
JavaScript	A lightweight, interpreted programming language
TypeScript	A superset of JavaScript
React	JavaScript library for frontend
Angular	Typescript frontend framework
Python	Programming language
MySQL	Database management system
OWASP	The open web Application Security project
VM	Virtual Machine
CMD	Command-line, text interface for computer
VPN	Virtual Private Network
OVPN	Open Virtual Private Network
CSS	Cascading Style Sheets
SCSS	Sassy Cascading Style Sheets
SASS	Syntactically Awesome Style Sheets

Jenkins	Jenkins Automation server
Jest	JavaScript testing framework
Myfinance	KPMG internal app to manage invoice
KYC/Esign	KPMG product for creating electronic signature
UI	User Interface

2 Framework

2.1 Analysis of your current work

○ In KPMG and my development team, our working process is defined based on the sprint period (2 weeks) and there is a Jira ticket of tasks that I need to complete in the sprint.

My main work tasks for 10 weeks working with 2 products are:
- Set up the local development environment for projects.
- Research and implement new features
- Refactor the codebase
- Review and test task of other developers

Set up the local development environment for projects

Since most of the projects that I have joined have been built more than 1 or 2 years ago. The projects are really large and complicated with many layers, security configures steps. Depends on each project, there will be a Wiki page to follow on the set-up process, but there is also a project, which is considered simpler, does not have any Wiki page. I have so much more difficulty setting up myself when I cannot follow any instructions. I need to ask step by step who already successfully setup. Also, with the projects which have the Wiki page, as a developer, I need to keep update the wiki page to catch up with the newest environment, for example, 2 years ago, the instruction will be compatible with Ubuntu 14 but now it will be different with Ubuntu 20.04.

Research and implement new features

The developments and specs of the projects are growing continuously to meet and fulfill the demands of end-users. The development teams have to implement many new functionalities to the current products. As a new developer in the team not only learn about projects but also research new technologies to apply for implementing new features. I realized that the time I spent on researching is more than the time I code during my development process. I can improve my JavaScript and Java skills which I learned in university. One of the important skills I needed in this work test is communication and comprehension skill. I need to understand the

ticket description thoroughly and communicate efficiently with other developers to succeed in my tasks.

Refactor the codebase

This task happens whenever I need to implement something new to the current application. This is an important task as well. A code refactoring is necessary and processing at the same time with implementing to make the code base more maintainable, clean, and easy to optimize. This work also helps to save time for the later developer to join the team, also saves my time to explain each line of code. Experience is the most needed skill for this task. All of the tips of writing clean code I learned are from my previous work. A clear coding convention for the whole development team is essential as well.

Review and test task of other developers

During my working time in KPMG, besides developing my task, I also need to review and test others'. In my previous company, there was a tester specialized in testing our features. However, in this work, I also work as a tester. To be successful as a reviewer, I need to have strong programming skills and coding clean to point out some mistakes in the coding style that others cannot see. As a reviewer, I need to make sure the code of other developers is clear enough to be easy to read. As a tester, I work as an end-user, try all the test cases, and expect to get the same result as the description. To be a good tester, I also have the ability to see things in many aspects and imagine many scenarios that may happen when the users use the products.

2.1.1 Evaluation and development plans

I have been received positive feedback from my colleagues about the tasks that I have completed. I believe I still need to try more to improve myself but as a junior developer at this point, I think I have done fairly. Before joining KPMG as a junior software engineer, I have six months intern in a startup company in Finland and then 5 months in a big software company in Vietnam. I also have gained many useful programming pieces of knowledge at Haaga-Helia, which makes me stronger with my background. However, to get to the level senior or mid-senior

developer, I need about 1- 1.5 years more to broaden my knowledge with Cloud, DevOps, and backend.

Spring Boot –I already learned Spring Boot at university, but it was 2 years ago, and I have been focusing on JavaScript and its framework only, therefore, it's a good time to get back the skill and enhance it.

React – the time I spent working on the project using React in this company is not so much, but it is enough for me to revise and catch up on some new updates of the language.

My development plan in the future is focusing more on backend development because I like the challenges in the backend more and I feel like I am comfortable with logic more than UI in frontend. To support for the backend development plan, I also plan to get a Cloud certificate in the near future.

2.2 Interest groups at work

My personal work can affect the following interest groups:

Internal:

- o My teammates who work on the same sprint;
- o Other developers in the Application Technologies team who may not currently work in same sprint with me but have knowledge about the product that I work with.
- o Product Owner who is a bridge between customers and development team;
- o (Optional) Company's partners ("operators") who distribute the product to end users;

External:

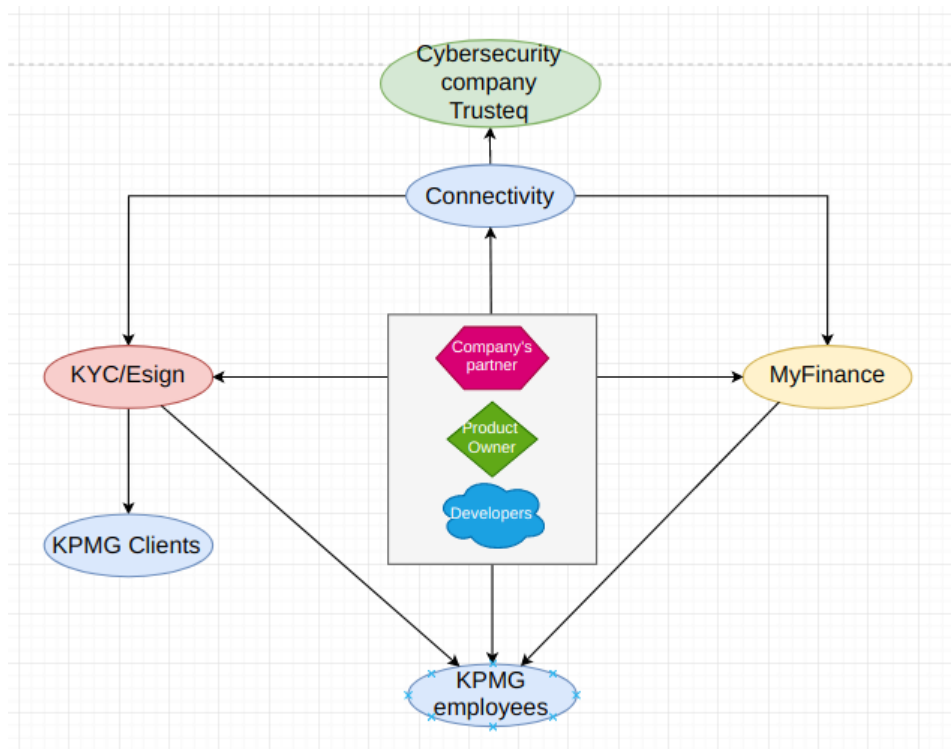- o End users who directly use our products

Figure 1. Interest groups possibly affected by my work at KPMG

Most of the products I have been building is for internal use (KPMG auditors and other staff), however the KYC/Esign (this is an electronic signature service) is popular used by KPMG Clients as well. Therefore, my work can potentially influence a wide number of people within the company and company's clients. I can see my influence clearly while talking with my friend, who works in one of KPMG's Clients, he also used the electronic signature service of KPMG. This has motivated me a lot and put more responsibility because my work can directly affect to people.

2.3 Interaction skills at work

It was interesting because the important interaction skill at work is the online meeting. Because of the coronavirus situation these days, most of my colleagues have not come to the office at all since March. Therefore, everyone is familiar and now more comfortable with an online meeting, online calling.

In KPMG and especially the Application Technologies team, we have a team weekly meeting for all the members in the Application Technologies (12 members) every Wednesday. For the Lighthouse department with almost 50 people, we have

a monthly meeting at the end of the month to inform about plans for the next month as well as a summary for the current month. However, I have interaction the most with the small team that I work with within a sprint. We have a daily meeting to report what we have done the day before, and the plan for a new day, also point out any issue or barrier if there is any. Besides daily calls on Microsoft Teams, our team discussion is happening mostly on the Telegram chat group or sometimes on the comment section of the JIRA ticket. We use Telegram as a means for communication in a development team because some of our developers including me are using Ubuntu OS, which is not supported by Microsoft to use their services.

I do not have any direct interaction with customer services. If there is any incident that happened in the product that I work with, mostly I will get the information from the Product Owner or the Scrum Master.

Personally, I think the most obvious challenge that I have faced is the inconvenience of online communication. Since we don't have to go to the office, people have more flexible time on how they start a workday as long as they complete their task. Therefore, it is really difficult to contact people if I need to discuss something because people are not always online while they work, or reply to the message in time. Especially if you need to show or explain something to the person, which is always easier to do face-to-face.

3 Diary entries

3.1 Observation week 1 (October 26 – October 30, 2020)

Monday 26th October 2020

Today is my first day at work in the new company. I joined the KPMG Lighthouse, in the KPMG Application Technologies team as a junior developer, at the same time with another senior developer. My tasks on the first day at work were mostly getting familiar with the new working environment, the new working devices, and be able to access the internal system.

The company provided each of us with a mobile phone and laptop with keyboard and mouse as well. My laptop is a Dell Latitude e5410 with Linux OS, Ubuntu distribution. As I have been only using MacBook with the macOS and sometimes using Windows in school, it is my first challenge to get familiar with the Ubuntu system. Compared to Window OS, the Ubuntu command line is more similar to the macOS.

Every new employee has one peer advisor to help them with setting up the work equipment and answering some working environment-related matters. My peer advisor has to work at KPMG for about two years and he is also a software engineer, with DevOps focused. It turned out easy for me because he supported me a lot with configuring the project environment and installing operation system.
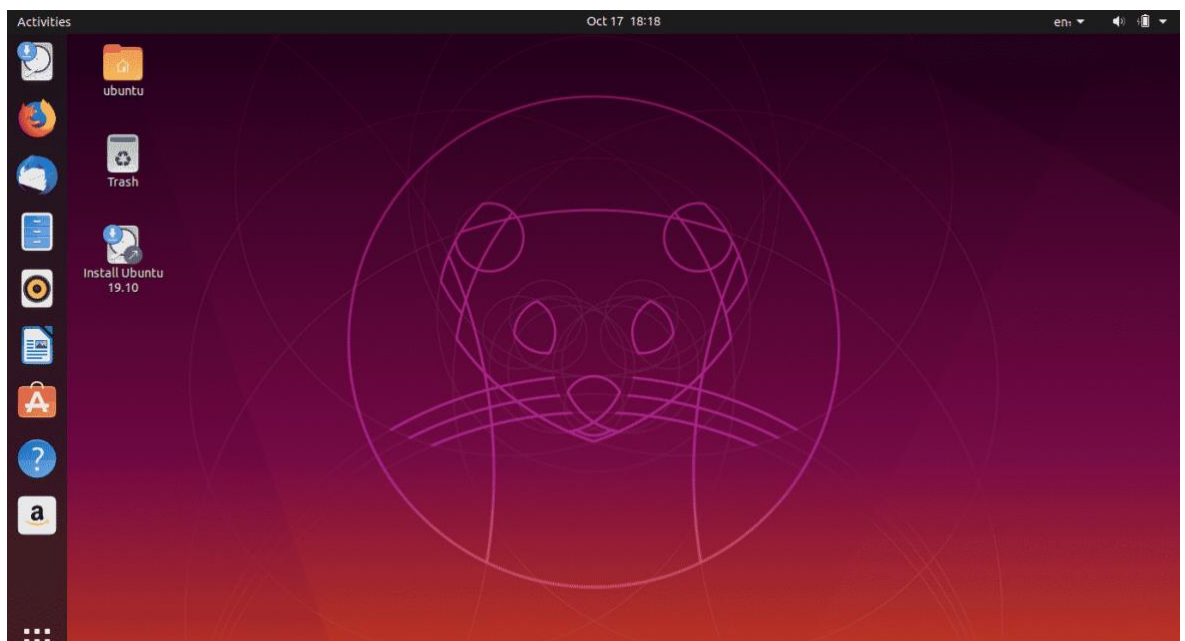

Figure 2. A screenshot from Ubuntu OS 20.04

With a brand-new operating system set up, I spent time exploring Linux OS to know more about what I will be working with. As I read, Ubuntu might not be the best Linux distribution, but it is the most popular one. It currently is the most popular operating system on the Cloud. It is the next generation of Debian Linux, which is one of the oldest and well-respected distributions (Memon, 2020).

About the user interface, I think Ubuntu is fairly user-friendly. It is a combination of macOS and Windows. If someone is totally new to Ubuntu and has experience with macOS or Windows, I do not think there is any big challenge for them to feel comfortable with Ubuntu.

Tuesday 27th October 2020

I have two meetings today and one tea break time every Tuesday afternoon. The first meeting with the Head of The Application Technologies team. He introduced all of the projects which the team is in charge of and delivered general information about their business for us to have a big picture of what they are doing. The meeting was quite informative and gave us a good first impression of what we will work on in the future.

The second meeting is with the KPMG Technologies team's assistant who is responsible for all HR-related matters in the team. She was also a student of Haaga-Helia University of Applied Sciences, Pasila campus. The meeting was all about onboarding-related information. We got to know about how to mark working hours correctly, all the benefits of the company such as travel card, learning resources, or company sports, events, ...
During the tea break, I and the other new employee had a chance to see other members of the team because since they are all remote working, it was the first time I can see all the members of the developer team.

Wednesday 28th October 2020

We have to build a Virtual machine in the background to run our projects. This is the first time I worked with vagrant open source.

In the afternoon, we had another meeting again with our Head of The Application Technologies team. This meeting was about the software development process. He introduced the team structure and all members of the development team and their responsibilities.

As the team is large and "asynchronous" working on different projects, they decided to use a Scaled Agile Framework for project management.

He also introduced about Jira - a famous project tracking software. In KPMG and in our team, we applied OWASP level 2. OWASP - The Open Web Application Security Project is an online community that produces freely available articles, methodologies, documentation, tools, and technologies in the field of web application security (IBM, 2015).

He also mentioned git style in the team. Normally, I only use git merge to merge my branch to the master branch when working in a team. However, they use git rebase here.
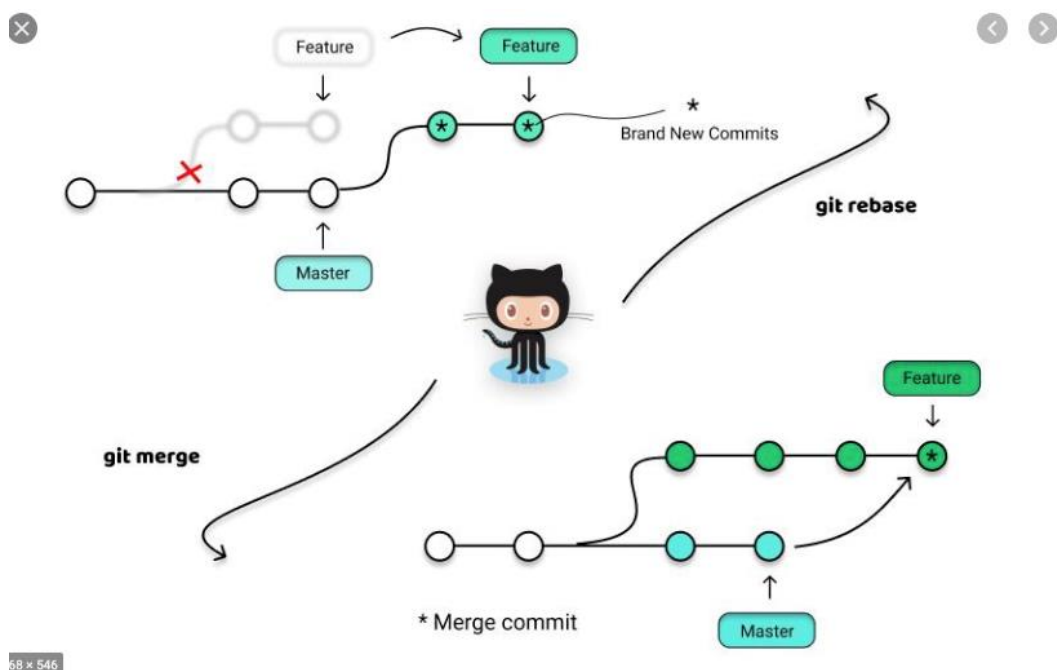

Figure 3. Compare merging and rebasing
Source: https://www.atlassian.com/git/tutorials/merging-vs-rebasing

He also mentioned code review in the team. We prioritize the quality of code, so we have a very strict coding review.

Thursday 29th – Friday 30th October 2020

After I had to reboot my laptop in the morning, I cannot run again the project what I managed to make run yesterday. I learned from my peer advisor that every time I turn off the VM, in the environment project, I need to run `vagrant reload` and `vagrant up` to run again my VM. Also, in the project that has a Docker container, I need to tear down the built file and build it again.

In the afternoon, we had a meeting with the Product Owner of one of the projects. She gave a demo of the service and discuss what is still missing and what is in the building progress.

After the meeting, we spent most of the time completing the mandatory courses of the KPMG internal system. There are quite many courses that every new employee has to train when working in KPMG. I had 6 totally mandatory courses to finish.

We learned about Data privacy, which is extremely important. We also trained sentinel requestors.

My colleague shared me a tip on Ubuntu to create new document faster.

`touch ~/Templates/Empty\ Document`

This is my right click before and after setting up:
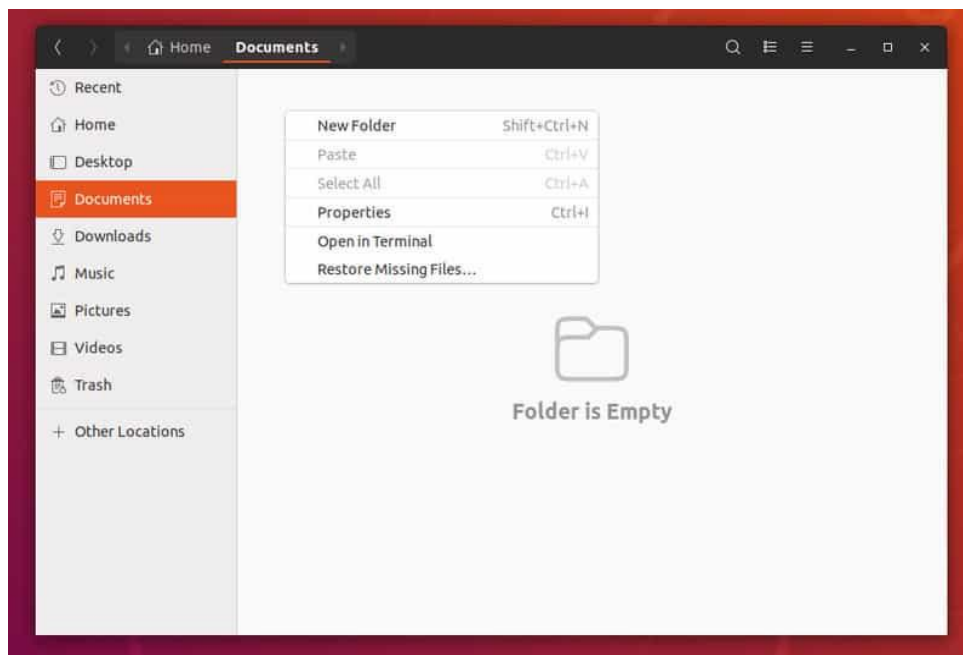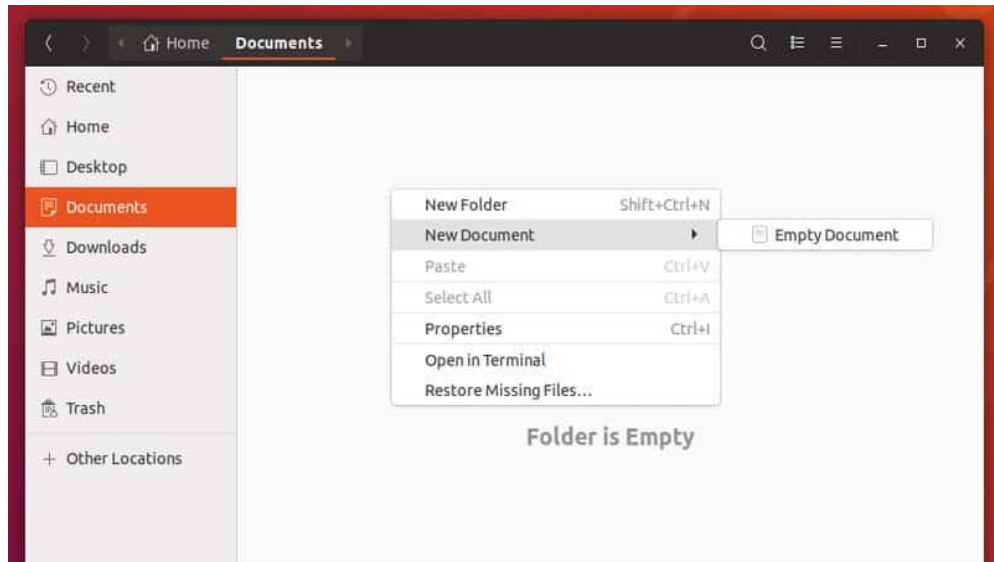

Figure 4. Screenshot before set up tooltip

Figure 5. Screenshot after setting up tooltip

Week 1 evaluation

It was the first week and my task was getting familiar with working space and working devices, manage to use the internal system for marking hours, and checking on company benefits. I cannot really experience the welcome environment on the first working week because everyone is working from home right now. In my office, there are about less than 10 people even normally it will reach hundreds of people.

In general, because it is a global company which has many rules and regulations to apply, I have to learn quite many new mandatory courses which are not much relevant to my main task and surely time-consuming.

A good thing from the first week is I got to work with the new operation system Linux which I can learn about many cool Linux command line syntax that I cannot find in macOS before. In my opinion, I still believe MacOS is running faster than Ubuntu and MacBook is a better laptop than Dell. In my own experience, Ubuntu seems slow sometimes and made significant noise when running. For Ubuntu 20.04, sometimes my screen got haywire and stopped, I have to reboot the laptop again which is quite inconvenient.

One of the things I also spent time setting up a lot this week is the Docker container.

This docker contains 8 smaller projects inside, that is the reason why whenever I run the docker, my laptop is getting too hot.
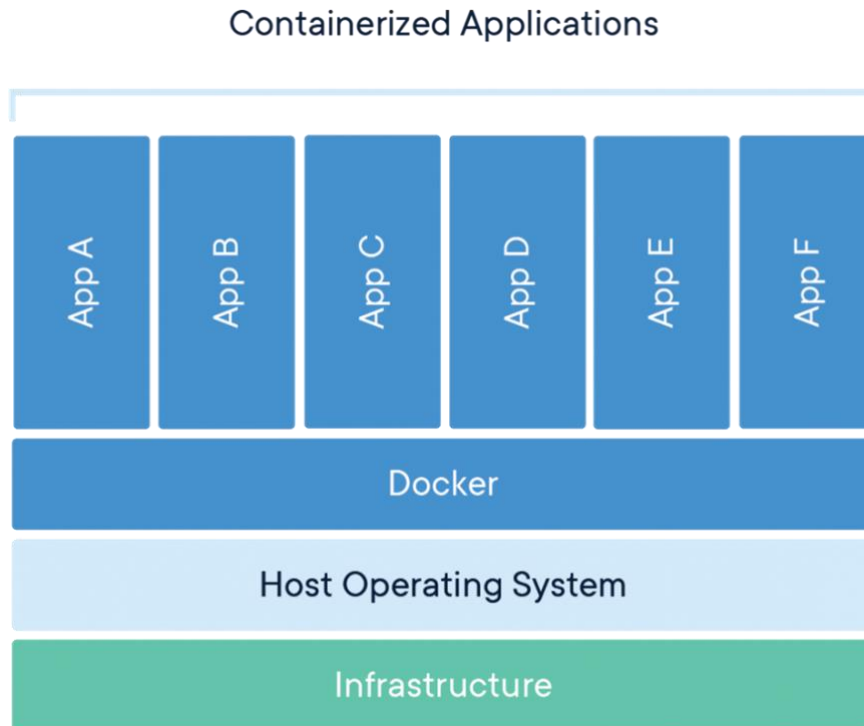

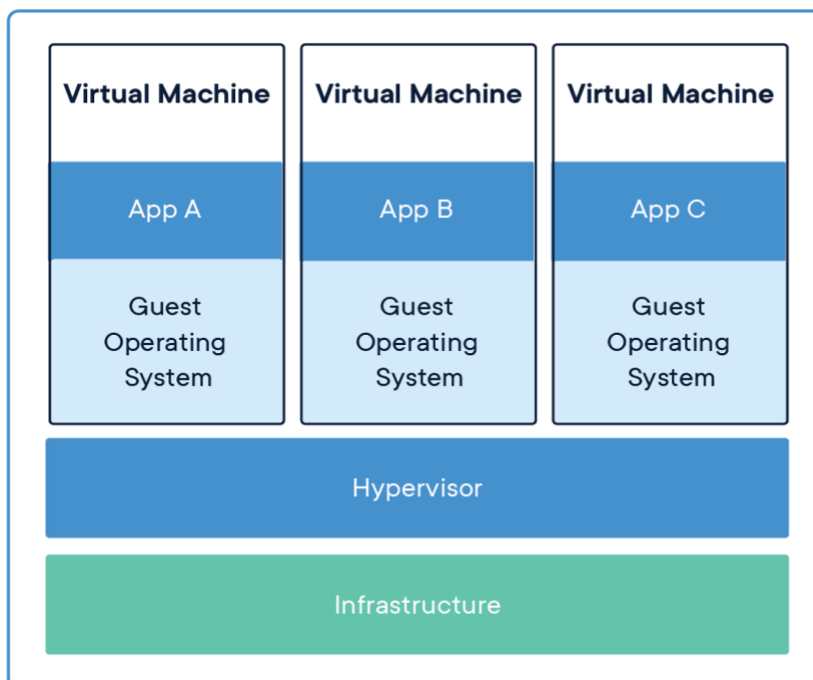Figure 6. Docker container illustration 1

Source: https://www.docker.com/resources/what-container

Figure 7. Docker container illustration 2

Source: https://www.docker.com/resources/what-container

I can develop my DevOps skills quite significantly because there is no specific DevOps engineer in my team, everyone is expected to be full-stack and there are some people who favored CSS and frontend more, and there is some favored backend more but everyone is supposed to know everything at a decent level. I like this way because I do not have to be fixed with one label "backend" or "frontend", but I can broaden my learning area.

Overall, it was such a challenging week but thanks to the support from my co-workers, the week was satisfying ending.

3.2   Observation week 2 (November 2 - November 6, 2020)

Monday 2nd November 2020

We are in the Sprint 4 of the Plan Increment Period including 5 sprints. Each sprint lasts 2 weeks. This Thursday, we will start the new sprint, so everyone in the team in the hurry to finish all the tasks left from this sprint and update the backlog for the next sprint.

After discussing with our team leading, we can take some simple and easy tickets on Jira to start to work on, in the meanwhile, reading and analyzing the code base as well. There are 2 tickets needed to be reviewed so I think it is a perfect time for us to review the code and analyze the code.

I set up the SonarQube on the website so we can scan the code on SonarQube server.

Add following plugin to pom.xml of the backend codes:

```
<plugin>
  <groupId>org.sonarsource.scanner.maven</groupId>
  <artifactId>sonar-maven-plugin</artifactId>
</plugin>
```

Run SonarQube in Docker container by executing following commands:

```
docker pull sonarqube
docker run -d --name sonarqube -p 9000:9000 -p 9092:9092
sonarqube
```

```
mvn clean verify sonar:sonar
```

The SonarQube interface is available at port 9000. http://localhost:9000/

We have a meeting to train about Scaled Agile Framework and where they are going to allocate us for the next sprint.

She also distinguishes between the responsibility of Scrum Master and Project Owner which is easily confused.
Project Owner is a communication gate between customers and stakeholders to the development team
Scrum Master is taking care of meeting and technical issues, also protect the Development team from the outside barrier.

Tuesday 3th – Wednesday 4th November 2020

Last week, I tried to set up the VPN to work from home but still need the admin to add my public key to the network to activate it. This morning, he response the email and asked me to check the network.

I checked the network with these command line:
Connect to the Wire Guard server:
```
wg-quick up wg0
When done:
wg-quick down wg0
```

I can successfully connect to VPN at home so I can work at home safely using the company's network.
Later on, in the day, I tried to download some useful extensions for Java and TypeScript on VsCode.

Thursday 5 th – Friday 6th November 2020

Since yesterday, when I tried to set up all the extensions for VsCode, I realize that there is no support for Java yet on the VsCode extension in our team. I searched around and happily to know that, from Java 11, we can compile Java codes on VsCode easily (normally they use Eclipse to compile Java code).

It was not easy as it has to be because our project is running by Java 8 and I have to update to Java 11 to enable all the VsCode's extensions. My solution is that I still download and run Java 11 globally to enable all the extensions but will keep the Java 8 in the java configure runtime.
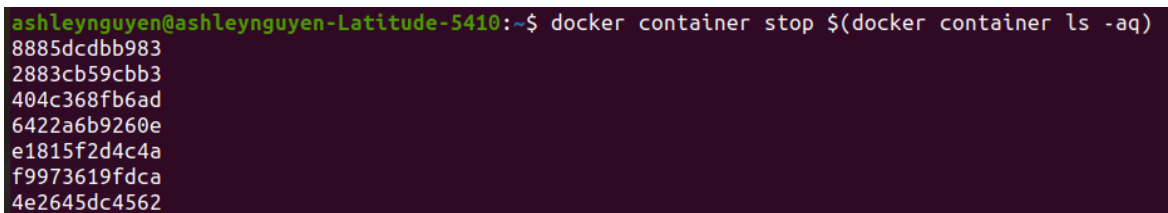
After spending hours reading a lot of articles and solutions on GitHub, also Stack Overflow, I can make it run on my project.
I also learn how to use snippets. If developers get familiar with the snippet, he can save a lot of time for coding.

The laptop has a very loud noise even I did not open any application, it was fresh. I searched on the internet about the cause. There are some answers from Ubuntu users that I should check if the docker is still running in the background or opening Citrix, the VM window computer. They are a few possibilities that cause the slow and the sound.
I first closed all Virtual Window Tabs, but the sound is still quite loud. I tried this command line to test if there is any running docker.
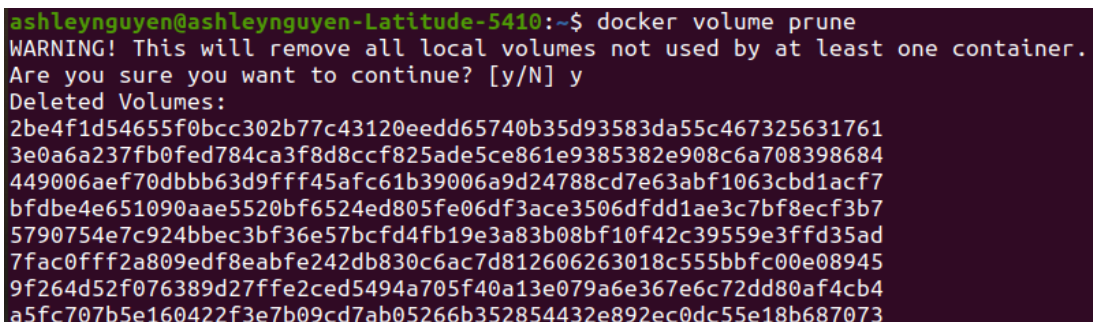
```
docker container stop $(docker container ls -aq)
```


I just run the CMD to stop all the running apps. Continue I run this command line to delete all the local volumes that docker has obtained:

```
docker volume prune
```


There was totally 12.16GB is removing from the local volumes from docker.

Week 2 evaluation

This week, I started to review the pull request for some tickets. Normally, one ticket can be done after reviewing and testing by one developer in the team. The other developer who also joined KPMG at the same time as I was not familiar with Java Spring Boot because his main programming language is JavaScript and he used React, so he asked if he could review and test the ticket together with me. I think it is a good idea because we are all fresh to the product, we may be confused so it is good to review together and discuss.

When we tested the features, we recognized 2 minor bugs in the product, and we reported them to the team. The product owner was so glad to hear that. She also gave us a compliment and encouraged us to give more feedback.

One of my new achievements this week is I managed to take advantage of VsCode. The project in the team started in 2018, and people mostly used Eclipse on the backend site, because at that time, Eclipse is the best compiler for Java. So, people in the team keep using it without any complaints. But with the fresh eyes on the team, I am more comfortable using VsCode and I tried to use VsCode for the backend as well. And luckily, Java 11 which is released in late 2018 can be compatible very well with VsCode. Also, here is a list of useful extensions on VsCode for a full stack developer (Angular in frontend and Spring Boot backend):

1. Angular Language Service
2. ESLint
3. TSLint
4. TypeScript Hero
5. Sass
6. SCSS Refactoring
7. IntelliSense for CSS class names
8. Bracket Pair Colorizer
9. Angular Files
10. GitLens
11. Code Spell Checker
12. Path Intellisense

13 Auto import

14 Java extension pack

15 Spring Boot extension pack

Here is the step by step to install and activate Java 11 plugin with Java 8 based project:

Start by installing all of these following plugins in VS code:

Java Extension Pack, includes the following plugins

1 Language Support for Java (TM) by Red Hat

2 Debugger for Java

3 Java Test Runner

4 Maven for Java

5 Project Manager for Java

6 Visual Studio IntelliCode

Spring Boot Extension Pack, includes the following plugins

1 Spring Boot Tools

2 Concourse CI Pipeline Editor

3 Cloudfoundry Manifest YML Support

4 Spring Initializr Java Support

5 Spring Boot Dashboard

(Optional) SonarLint

(Tested on Ubuntu 20.04)

VS Code Java plugins require Java 11 or newer to run. Fortunately, there is a way to tell VS Code to use Java 11 and still keep have project running an older version. Here's how to do it for Java 8.

Install required Java versions

Make sure you have both Java 8 and Java 11 installed by running the following commands in the Terminal:

```
sudo apt-get install openjdk-8-jdk
sudo apt-get install openjdk-11-jdk
```

Configure VS Code settings

Use the Terminal to check where your JDKs have been installed. Use the following command:

```
sudo update-alternatives --config java
```

The output will look something like this:

```
Terminal

→  ~ sudo update-alternatives --config java
There are 2 choices for the alternative java (providing /usr/bin/java).

  Selection    Path                                            Priority   Status
------------------------------------------------------------
  0            /usr/lib/jvm/java-11-openjdk-amd64/bin/java      1111       auto mode
  1            /usr/lib/jvm/java-11-openjdk-amd64/bin/java      1111       manual mode
* 2            /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java   1081       manual mode
```

Make note of the paths until the OpenJDK folder. You will need them in the next step.

Open VS Code settings from the top bar by navigating to File > Preferences > Settings. The settings you will have to configure are java.home and java.configuration.runtimes. You can search for them on the settings page and open the VS Code settings.json file. Paste in the following values for the required settings:

```
settings.json

"java.home": "/usr/lib/jvm/java-11-openjdk-amd64",
"java.configuration.runtimes": [
    {
        "name": "JavaSE-1.8",
        "path": "/usr/lib/jvm/java-8-openjdk-amd64/"
    },
    {
        "name": "JavaSE-11",
        "path": "/usr/lib/jvm/java-11-openjdk-amd64"
    }
],
```

Figure 8. Screenshot java configure runtimes in VsCode

The java.home setting tells VS Code which Java version you want it to use internally. The java.configuration.runtimes setting specifies the different Java versions for your project to find.

- o Restart VS Code
- o Restart VS Code for the changes to take effect.
- o Running tests

You should now be able to run your Java unit tests created with JUnit or TestNG right in VS Code. From the left side navigation bar go to Test, choose the tests you want to run and run them by clicking the green play button. You can choose a specific test or all tests in a folder. The test report opens in a new tab.

In conclusion, after this week, I can be ready to code a simple, less dependent feature in the project. The environment set up is ready as well. We also had a meeting to inform about all the internal codes for hour registration in the working hour management system.

For example, all hours that are spent to Scrum of Scrums, are reported to IO-code: **Software Development Management.**

If the time for studying is directly related to a project / service, the hours should be inputted to that set project's / service's Statistical IO code.

If the studying has been general studying for personal knowledge gain, the hours should be inputted to internal IO code: **Internal training and self-learning**

Thanks to that, the product owner will know exactly how many hours a developer spent on building some specific feature and allocate better for next sprint.

3.3   Observation week 3 (November 9 – November 13, 2020)

Monday 9th - Tuesday 10th November 2020

I worked from home for two days because I feel sick.

In the daily meeting, all tasks of the last week are cleared. There was one ticket needed to review an application written in React. As I have more experience in React than Angular, I am more than happy to take the responsibility to review the ticket.

There were not many changes to the ticket, so there are not many things to fix. However, this project was written by 2 developers who are learning React, so there are many rooms to refactor. I can see that she/he used actual DOM (document.createElement) in the React app, which is not a good coding habit in React because React mostly manipulates the virtual DOM.

Some small bad practices that I notice is the template literal is used with the single quote not with the back stick.

```
`example ${example}`
```

Some lines of codes with if else conditionals can be improved better is like:

```
if(errors[Id]){ (
            <span className="invalid-example">
                <FormattedMessage id={errors[Id]}/>
            </span>
        )} else
            {null}
```

can be replaced like this:

```
{errors[Id] && (
            <span className="invalid-example">
                <FormattedMessage id={errors[Id]}/>
            </span>
        )}
```

Wednesday 11th - Thursday 12th November 2020

I went to the office to fix the secure problem that I had yesterday when I cannot access to the database.

I still did not know what the original cause of the problem is but since I got it again whenever I work at home, so I think the network at home is not secured to reach the database.

Fixed the screen Ubuntu 20.04

One of the issues I got quite randomly at work is my flickering screen. I did not know why I got it and how to make it happen again to show the IT guy to fix but it just happened quite randomly. After searching on the internet, the answer got from most of the discussions is because of the newest Ubuntu 20.04 graphic card is not compatible with most of the screen so it is crashed at some point. There was one solution to fix it by using Wayland display server on the logging screen: click your profile > click the white wheel in lower right corner > choose 'Ubuntu on Wayland'.
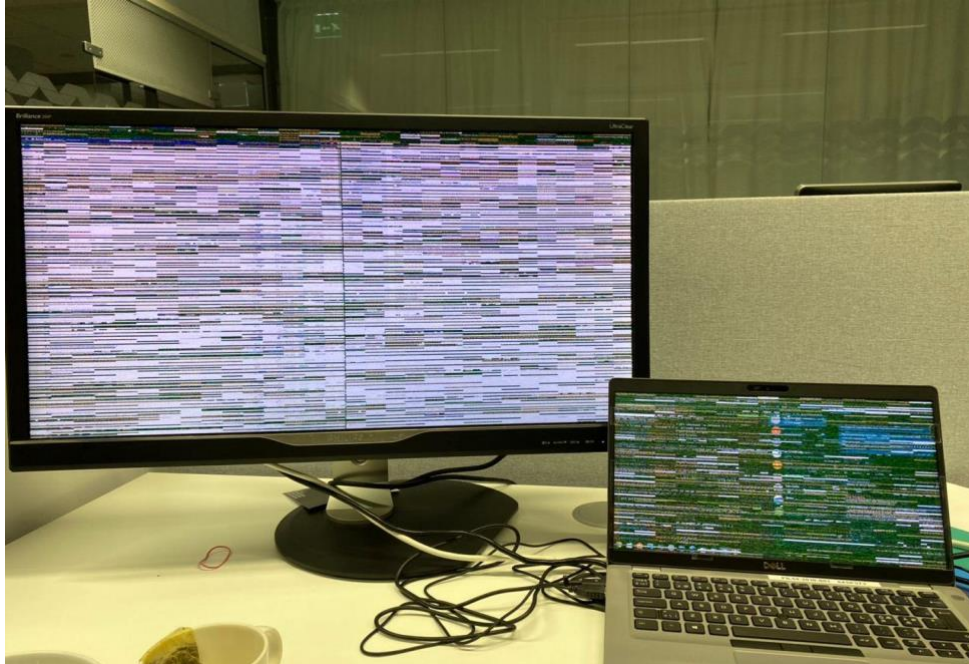
Figure 9. Flickering screen

The colleague who also joined the team at the same time, also got the same widget I had and also got the same problem. We both changed to Wayland.

Friday 13th November 2020
I was off in the morning.

In the afternoon, I had a Sprint Planning meeting with the development teams and product owner of one key product. We set the goal and story point (how many efforts needed) for each ticket in the back log.

Also, in this next Sprint (from 17.12.2020 – 31.12.2020), most of the developers will be on holiday, even the Scrum master. There will be only me working on the product, so they have to consider and replan the sprint to be more reasonable and feasible for me.

Week 3 evaluation

This week, I have learnt a lot about the product and the code base. In this week, I spent time analyzing the React product and the Angular product. For the React product, there was quite many discussion between me and the other developer about coding convention. As I mention before, the original developer team has the most experience with Angular so React is not a major language in the developer team, that's why the React app still needs to improve quite a lot.

We have one discussion about which one is good practice in styling for React app: material UI **makeStyles** or **SCSS** architecture?

Styling method for React app:

The makeStyles is a great way to use a theme-based UI. It allows you to use common properties and methods to generate style based configured variables.

Hook usage with theme:

```
const useStyles = makeStyles((theme:MyCustomTheme)=>{
  root: {
    background: 'linear-gradient(45deg, #FE6B8B 30%, #FF8E53 90%)',
    border: theme.myBorder,
    borderRadius: 3,
    boxShadow: theme.shaodws.boxShadowPrimary,
    color: theme.colors.primary,
    height: theme.size(1),
    padding: '0 30px',
  },
});

export default function MyCustomThemeButton() {
  const classes = useStyles();
  return <Button className={classes.root}>Hook</Button>;
}
```

Figure 10. Example of using makeStyles in React Hook

Source: https://stackoverflow.com/questions/59696613/for-a-react-app-theme-is-there-an-advantage-of-material-ui-makestyles-vs-scss

This is from material UI website:

Material-UI's styling compound is combined by many other styling libraries such as styled-components and emotion.

- It has the same advantages as styled components.
- It's impressive fast.
- It's expansible via a plugin API.
- It uses JSS at its core – a high performance JavaScript to CSS compiler which works at both runtime and server-side.
- Less than 15 KB zipped; and no bundle size increase if used alongside Material-UI.

(MaterialUI, n.d.)

If mixing styling method in one React project is a no matter to you, I think I still chose normal SCSS style for simple components.

Should I return HTTP 200 empty or 404 Not found?

When testing an API, I checked the response and realize when I asked for data which returns nothing, the message is shown 404. It was a little bit odd for me to look at the response code. 4XX mostly mentioned about Bad Request, but the request here is valid.

1xx informational response – notice that the request was received, continuing process

2xx successful – the request was successfully received, understood, and accepted, give good result

3xx redirection – need further action to be taken to finish the request

4xx client error – the request contains poor syntax or cannot be fulfilled

5xx server error – the server failed to complete an obviously valid request

(www.iana.org, 2011)

200 should be returned that has the body of whatever you successfully fetched. Not appropriate when the data you're fetching doesn't exist.

400 is used in response to a badly formatted HTTP request (for instance malformed http headers, incorrectly ordered segments, etc.).

Newer RFCs (Request for comments) now allow for 400 to be used for semantically invalid requests (Overflow, 2012).

In my point of view, I still want to return 200 with an empty array more, however, in the definition of 4XX message code, they mentioned "4xx client error – the request contains bad syntax or cannot be fulfilled", so 404 is acceptable in this case as well. Because they cannot return what is expected in other word "not fulfilled".

3.4   Observation week 4 (November 16 – November 20, 2020)

Monday 16th November 2020

In the morning, we have a long meeting about the Angular and Spring Boot project. There is one ticket that cannot be done in this sprint, so we all agreed to move the ticket to the next Sprint and split it into smaller tickets, so many people can share the responsibility.

In the afternoon, we have a PI Planning meeting for the whole big team. Last week, each project has divided into a smaller group of developers and a project owner group to discuss more the project and plan for the next sprint. In this meeting, we all gather and confront any blockers or objectives of all products. This meeting includes the developer teams, all product owners, and scrum master.

Tuesday 17th November 2020

I took a day off because I was sick.

Wednesday 18th November 2020

Today in the morning, I have a weekly meeting with the whole department team. We discussed about last 5 sprints because today is the last day of an increment period includes of 5 sprints which takes 10 weeks in total (2 weeks/sprint).

After that I spent my whole afternoon discussing with the IT infrastructure engineer about my VPN set up. I followed his instructions and changed many configurations but the application that I have been working on is still not working.

Thursday 19th - Friday 20th November 2020

These 2 days are filled with a lot of planning meeting of the coming sprint.
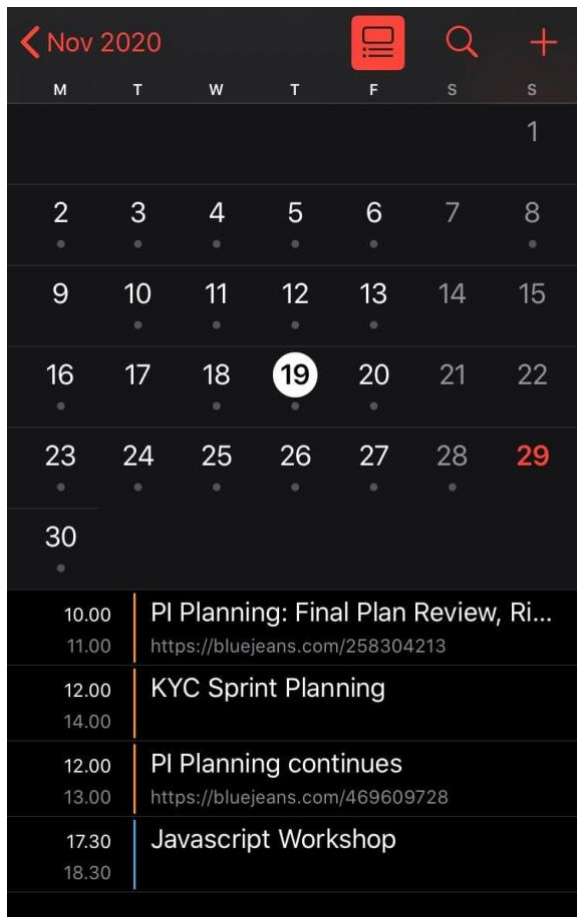
Figure 11. Calendar of a busy meeting day

The planning meeting. This is a time when the product owner and the team members negotiate about the tasks. During the meeting, the product owner presents the prioritized product backlog, explains almost every single task, and asks the team to select the reasonable tasks for the next sprint. The result of this meeting is a sprint backlog that is concurred by the team and clear to all team members (blog, 2020).

This is one of the 4 major Scrum meetings.

Week 4 evaluation

This week I can learn a lot about Scrum Agile Methodology in practical.

And in this week in my team, we were going through the Sprint Planning Meeting. We have to plan ahead 2 weeks so it was implemented carefully and seriously to make sure everyone can smoothly work on the tickets without any blockers or rushes later.
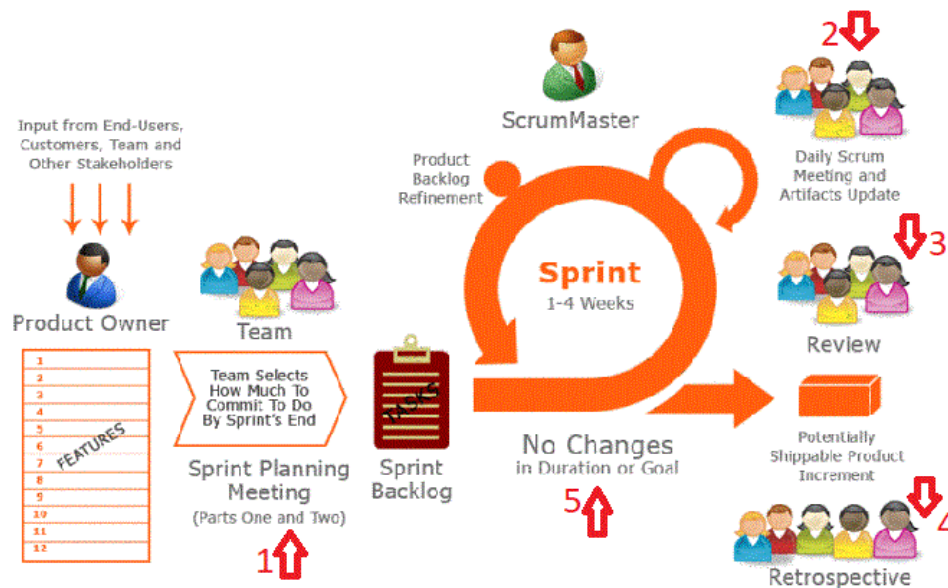
Figure 12. Life cycle of Scrum

Source: http://www.agiletroop.com/product/life-cycle-of-scrum/

This picture illustrates what I have learned from the university, but I haven't practically worked on any project using Scrum methodology.

In the meeting, the Scrum master will take out all the tickets and we voted to put them in order in the backlog from more important items to the fewer ones. We used a story point to measure how important the ticket. I did not know what story point before the meeting, and I researched it on the internet.

According to the Visual Paradigm article, a story point is a standard used in agile project management and development to estimate the complication of implementing a given user story, which is an abstract metric of effort required to work on it. In simple terms, a story point is a number that shows the team the difficulty level of the story. Difficulty can be related to complexities, risks, and efforts involved (visual-paradigm.com, n.d.).

To allocate the task precisely in the meeting, PO and the developer team make a rough estimation called product backlog grooming before the Sprint Planning to answer those questions:

Is the Sprint Plan is ready to be implemented efficiently?

Is there enough information to finish these issues?

Is the user story subtask divide practically?

And in my team, no one can individually decide the story point because the estimation can be different from developer level. We use an online tool to vote and decide the story point.
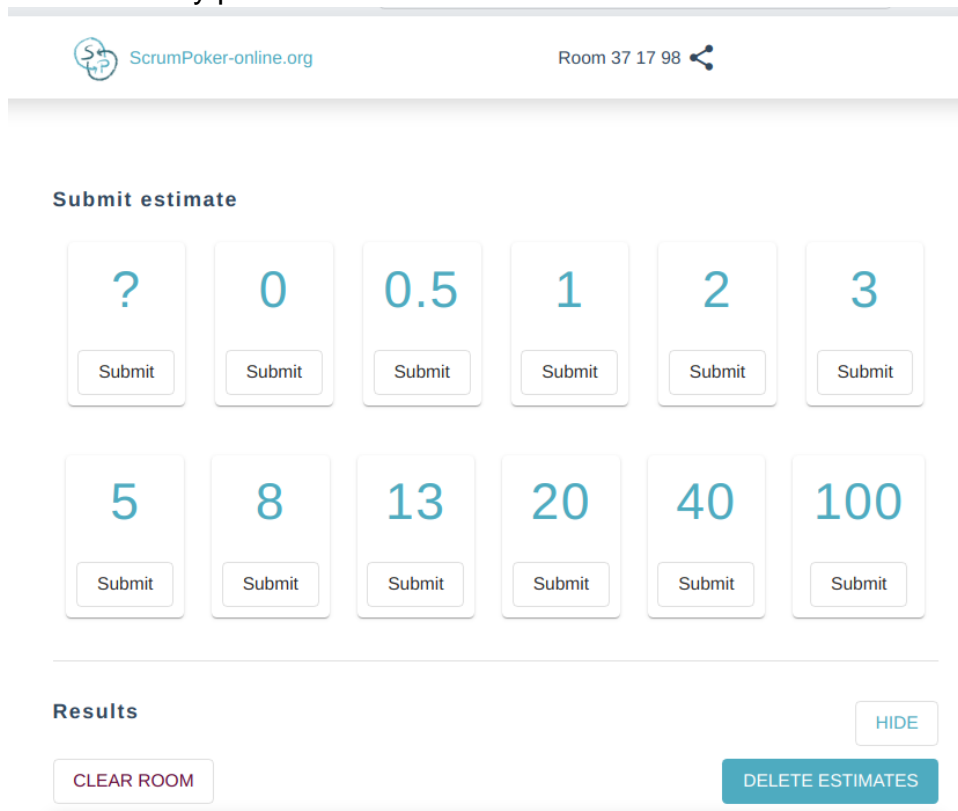

Figure 13. Screenshot from the app ScrumPoker-online

The result was hidden until the last developer vote, the scrum master will reveal the result and we will choose the most chosen story point. For example, we will pick 3 as a story point for a ticket after the voting below:

After the meeting, every ticket is listed in the backlog with the proper order. Every developer feels clearer what to do and ready to implement them all. I think our meeting was successful.

I can learn and understand a lot about Agile development in this week. Also, I get to know some tools to support the meeting.

3.5 Observation week 5 (November 23 – November 27, 2020)

Monday 23rd - Tuesday 24th November 2020

Today I start my new sprint. My task is generating Swagger for a feature in the application. Objective for today is reading and understanding the requirement in the ticket thoroughly. If it is not complicated, I can start to work on it.

The task is quite simple but the instruction for generating swagger script is a bit confused. Also, my VPN was not working properly since last week and I seek for help from IT department. The problems are predicted as my Wire Guard config and docker config for that project are crashed at some point which leads to the weird behavior.

After I completed the ticket that I has been working on, I need to write testing instruction so other developers can pick it up and test its feature.

Wednesday 25th November 2020

We all realized that the problem I cannot run the app properly is because Wire Guard and docker config. After many times trying, I have to use a different application, called OpenVPN as my VPN application (not Wire Guard).

In the daily meeting, I notified other developers that my previous ticket is ready to review so everyone can review it. At the end of the day, I see that my ticket was moved to the Ready for Test Deployment. It means my ticket was successfully passed all the tests and reviews as developer level. It will be deployed and tested by the Product Owner.

Thursday 26th - Friday 27th November 2020

Today I will work on the new ticket in the backlog. I need to create a method in the backend to add a pdf file to the current zip file. The ticket itself sounds difficult to me because this is a new feature that I haven't done before but luckily, there are some similar methods in the project that I can refer to and apply to my feature. I spent time reading the codebase and the connection between different methods so I can work and test the product.

It took me about 6 working hours reading codes, implementing, and testing to complete the feature. After that, I wrote testing instructions and moved it to ready to review column so other developers can pick up and test it.

On Thursday afternoon, we have a monthly KPMG Lighthouse meeting including all 25 members of the team. We did it virtually. The purpose of the meeting is to summarize the whole month, introducing new employees who are me and another senior developer, also the purpose of the meeting is planning for the next month. We plan to hire more data analysts and solution architects to the team.

Week 5 evaluation

As the coronavirus situation in Finland is getting more severe and people are recommended to work from home and I also want to work remotely but there was something wrong with my VPN set up that blocked me from connecting to the server. Therefore, I still have to go to the office to use the secured network in the company.

The docker itself and the VPN app (Wire Guard) itself were set up perfectly but in that specific application, it has been misunderstanding each other at some point that the Wire Guard only did not work with that specific application. In my team, there are only me and the new coming developer using Wire Guard, and he currently works with another project, so he basically doesn't have any blocker during his development period. Other members in my team are using OpenVPN and they also don't have any issue. I wondered why they don't let me use the

WireGuard for Trusteq Network Access

OpenVPN is rather outdated and working erratically. To remediate we are now using WireGuard for backup.
REGULAR USERS ONLY NEED TO READ THE SECTION CLIENT SIDE CONFIGURATION.
Server side configuration is provided for admins as some sort of documentation.

OpenVPN but spending 3-4 working days just to fix the issue which eventually cannot be fixed.

In the internal page showing how to set up VPN, they pointed out that the OpenVPN is old and outdated so they would like other new employees use Wire Guard.

I also research more about OpenVPN and Wire Guard to see how different of these two VPN applications:

- o Wire Guard is taking the VPN world by storm and promises big improvements over existing VPN protocols, such as OpenVPN.
- o However, currently, the gold standard of VPN (Virtual Private Network) protocols is OpenVPN. It is an open source protocol with good performance and is considered to be very secure and reliable, having passed numerous independent audits (Taylor, n.d.).

I have researched and read quite many articles and they conclude that for the time being, OpenVPN and Wire Guard have proven to the market that they stand out among the crowd and are worthy competitors.

If you don't think the lighter the programs are important then OpenVPN is the one to opt for.

On the other hand, if you are looking for a lighter, simpler VPN, it is recommended to opt for Wire Guard. Many organizations use two VPNs at times, so we could also give both of them a chance to identify which is a better match for ours.

Figure 14. A comparison Wire Guard and OVPN

Source: https://www.hitechnectar.com/blogs/wireguard-vs-openvpn/

I cannot see any clear difference technically between using Wire Guard or OpenVPN. For me, the set up for OpenVPN is easier and simpler and also it works well with my project so I can work remotely from now.

3.6 Observation week 6 (November 30 – December 4, 2020)

Monday 30th November 2020

Today I will continue with another ticket in the backlog. There are 9/23 tickets left in the backlog need to be finished before this Thursday. There are currently 4 developers in the team, and I think we can manage to complete them all in 3 days.

Currently I am fixing the ticket I worked on after receiving reviews from other developer. There is something I can note down is that I should write document

comment for each method. There is a problem I am facing is my Vs Code keeps generating the untracked file even I removed it. I realized it only happened in my Vs Code because other developers use Eclipse. I don't want to push that unwanted files to remote branch so I can use these command lines which only commit tracked files and ignored untracked files.

```
git add -u
git commit -a -m "Add your message here"
```

Tuesday 1 December 2020

Since yesterday I moved on to a new ticket, my objective today is finishing that ticket before the daily meeting so I can ask someone to review and test the feature.

Today my task is adding new information to a pdf file, it's a summary sheet. So, I need to modify the content of the summary file based on the actual result from other pdf files. For each pdf file, we have a "properties" file to contain keyword for multilanguage, in this case they are Finnish and English.

I am still new with the service that I am working on, therefore I spent most of the time to see the structure of this service, which files are linked and needed to modify to complete this feature.

Wednesday 2 December 2020

Yesterday, after going through and having a blast meeting with the other developer to discuss about some technical issues, today I continue to complete the ticket that I started yesterday. My goal today is completing this ticket.

From 10 to 11AM, I have an Application Technologies weekly meeting with my team. In this meeting, I was informed that KPMG will cancel all the celebration or event until next year because of the second wave of corona virus in Helsinki. They also recommend not to go to work because we should not gather a group more than 5 people. Also, other members in the team start to plan their holidays in order to keep the project still go well (avoid all the developers have a same period holiday so there will be no one for the sprint).

After the meeting, I am back to the ticket I was working on. There is one small issue that in the "properties" file for multilanguage, the pdf generate library cannot understand Finnish special character: å,ö,ä. I will give a solution of this more detail in the weekly analyse.

Thursday 3 December 2020

Yesterday I already completed the ticket. After that, I wrote the testing instruction carefully for the ticket before moving it from **In Implementation** to **Ready for review.** While waiting for other developer tests my ticket, I moved to another ticket.

Other developer is testing my feature. Because she is not working on this feature, so I need to explain for her and add more detail into the testing instruction so she can demonstrate again all the test cases. There is one issue happened. Her computer cannot generate the Finnish encoded, so it starts to show undefined characters. A wide guess is because of her Eclipse config which didn't work as expected so she asked me to clarify for her. We all agreed that it works for other computers except her so we will skip that part, and after testing other functionalities, my ticket is moved to **Ready for UAT Testing**, and now it is a job of product owner.

Friday 4 December 2020

Today in the morning, I have 2 meetings. The first meeting is daily meeting and the second one is Product Implementation Planning as we often call PI Planning. The second meeting lasts 2 hours and we discussed and analysed about all the ticket in the backlog because we are moving to the next Sprint now. There are only 3 developers left, because one developer is allocated to another project. However, he has not finished his ticket yet and in the middle of implementation. Our best solution is letting him finish his ticket first before getting new responsibility from other product.

After having lunch, I have a Demo presentation of a new product that the senior developer who was also on board the same time with me is a Scrum Master. In last sprint, he worked with this project and made some refactors, that's why he

wants to show the product demo to anyone in the developer team. This is the way to learn from each other.

After that, we have a Christmas party from KPMG. It's an online event.

Week 6 evaluation

During this week, my skills developed a lot. I got more familiar with the product and understand more what I am doing, also learning new tools, and knowledge of Java or git. I feel this was an efficient week and I am satisfied with my performance and learning progress.

First of all, after struggling how to not add untracked files to Bitbucket, I found solution for the case and also study some more new relevant command lines. Normally I will use the command line `git add .` for add all the changes in the working directory to the staging area. It tells Git that I want to include updates to a particular file in the next commit.

`git add -A` bring to stages all the changes
`git add .` bring to stages new files and modifications, without deleted files
`git add -u` bring to stages all modifications and deletions, without new files

`git add -A` is same as `git add . ; git add -u`

The highlight point about `git add .` is that it checks at the working tree and adds all those paths to the staged changes if they are either changed or are new and not ignored, it does not stage any 'rm' actions.

`git add -u` looks at all the tracked files and bring to stages the changes to those files if they are modified or if they have been removed. It will not add any new files, it only stages changes to already tracked files.

`git add -A` is a combined shortcut for doing both of those.
(Stackoverflow, 2009)

Git Version 1.x

Command	New Files	Modified Files	Deleted Files	Description
git add -A	✓	✓	✓	Stage all (new, modified, deleted) files
git add .	✓	✓	✗	Stage new and modified files only in current folder
git add -u	✗	✓	✓	Stage modified and deleted files only

Git Version 2.x

Command	New Files	Modified Files	Deleted Files	Description
git add -A	✓	✓	✓	Stage all (new, modified, deleted) files
git add .	✓	✓	✓	Stage all (new, modified, deleted) files in current folder
git add --ignore-removal .	✓	✓	✗	Stage new and modified files only
git add -u	✗	✓	✓	Stage modified and deleted files only

Figure 15. A table compare git version

Source: https://stackoverflow.com/questions/572549/difference-between-git-add-a-and-git-add

```
git add -A is equivalent to git add --all
git add -u is equivalent to git add -update
```

After `git add` the change to the stage, to commit changes and automatically add modified files (doesn't add new files)

```
git commit -am "commit message"
```

Normally, I just use `git commit -m` but if I want to commit only the modified files, not include the new files, I need to add the -a config before the message.

Also, this week I can also learn about how to print out Finnish special characters using encode.

ö = \u00f6

ä = \u00e4

38

For example: If I want the pdf gonna print out:

*Ilmoitetut poliittisesti vaikutusvaltaiset **henkilöt***

I need to write

*Ilmoitetut poliittisesti vaikutusvaltaiset **henkil\u00f6t***

3.7 Observation week 7 (December 7 – December 11, 2020)

Monday 7 December 2020

After having a daily meeting with the developer team, I have a discussion with the Scrum Master about which ticket I should pick up now. There are not many tickets left in the backlog, however, each of them is blocked by the other tickets which are currently implemented by other developers. So, to be short, I cannot start to work on those left tickets if the others have not finished their tickets.

In the meanwhile, there is a developer just finished his ticket, so my task is reviewing and testing the ticket. This ticket has a clear specs so I can review, and test quite fast based on the testing instruction. Also, this developer has experience with clean code so there is nothing needed to refactor in his ticket. He is also carefully when adding more mock data to support test cases and write unit tests for each method he created.

Tuesday 8 December 2020

The highlight of today is we have a 3 hours general PI retro for all developers in team. In a sprint, the Sprint Retrospective happens right after the Sprint Review and prior to the next Sprint Planning (visual-paradigm.com, n.d.).

The retrospective meeting is simply an "improvement" meeting happens to find ways and means to identify potential pitfalls, mistakes, and find out new ways to avoid those mistakes, which are attended by all – the product owner, scrum master, development team members, and optionally with the stakeholders (in our case, there is not stakeholder).

In the beginning of the meeting, the scrum master gave an anonymous link, and each of us will write down what we like and what can improve in the sprint. After

30 minutes, the master went through each opinion and talk about it. There are some complains about the unclear responsibility of a scrum master for a project. Because we don't have a separate scrum master, it is the tech lead of the project also a scrum master. Therefore, he/she spent time allocating tasks and meetings, also developing the project. They seem to do the extra job all the time. Everyone also mentioned about 2 developers just joining the team which includes me, they complement how fast I and other developer have been integrating.

Wednesday 9 – Thursday 10 December 2020

Today I moved to another ticket. It is not a ticket that needs to be built from scratch, this is a small feature which is good if can be improved, it's not urgent. I cannot start to work with other higher priority tickets because they are all blocked by other unfinished tickets, therefore I moved forward with the less priority ticket.

I need to make the hyperlinks which are generated from the pdf file clickable. Currently we are using the library PDFBox to create the pdf page. This library at this moment is not directly supporting the hyperlink. Therefore, in a discussion on Stack Overflow that I read, people tried to do other way around is drawing a blue line under the text to make it look like a hyperlink and make text redirect to website when we click.

I have tried to implement in this way but there are some issues. The pdf is not a static page, the content is based on what returns from the API, which is not static. I cannot know exactly the length of the text and the coordinates of the line, that's why I cannot draw the line.

There is other solution is using a library called PDFBox-Layout, there is a built-in function to add hyperlink so it will be faster to implement. However, to apply this library, I need to change the whole pdf file generation with more than 300 lines of code using this library. After discussing with the scrum master, I will replace the old library to the new PDFBox-Layout.

Friday 11 December 2020

My goal today is completing the ticket by applying new library to the file, if I managed to successfully apply the library, it will be a game changer for this

service. Because this library has so many built-in functionalities which support a lot for developing later.
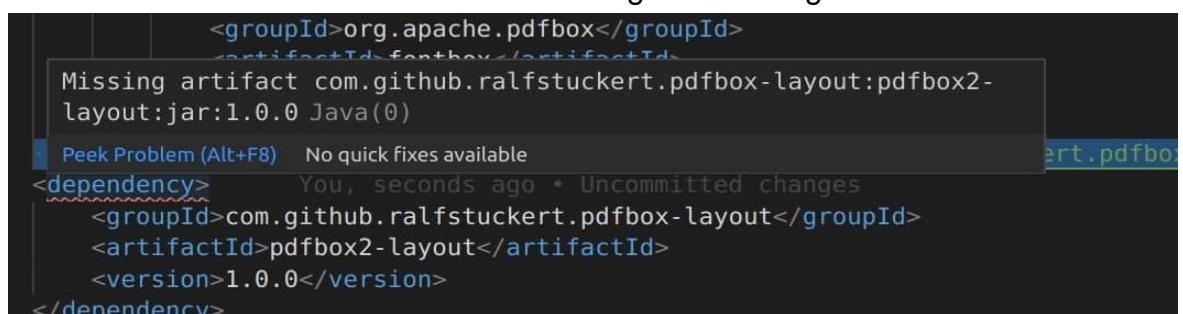
However, I first had problem when adding new dependency, I spent half of the day configuring it, after that I spent 3-4 hours applying the library and complete the ticket.

Week 7 evaluation

A week has ended, and I felt more satisfied than the previous one because, despite spending too much time researching and configuring things, my work has had fruitful results. I was just supposed to add only a small feature to the current service, but thanks to the new library, my result just opens a wide road for the new development later.

I confidently believed that I have the most experience with this library in the team. That is the reason why later on, other developers want to change their service using this new library, they always ask me to clarify. More than 4 hours researching all the solutions on Stack Overflow, going through the GitHub Page of the library, and checking all the examples gave me a reward.

One of the barriers I got when adding the dependency is not recognized by maven. Even I had teared down and build again but still get this error.

Figure 16. A screenshot of the dependency bug

After research, I can understand that I need to do more configure process to make it work.
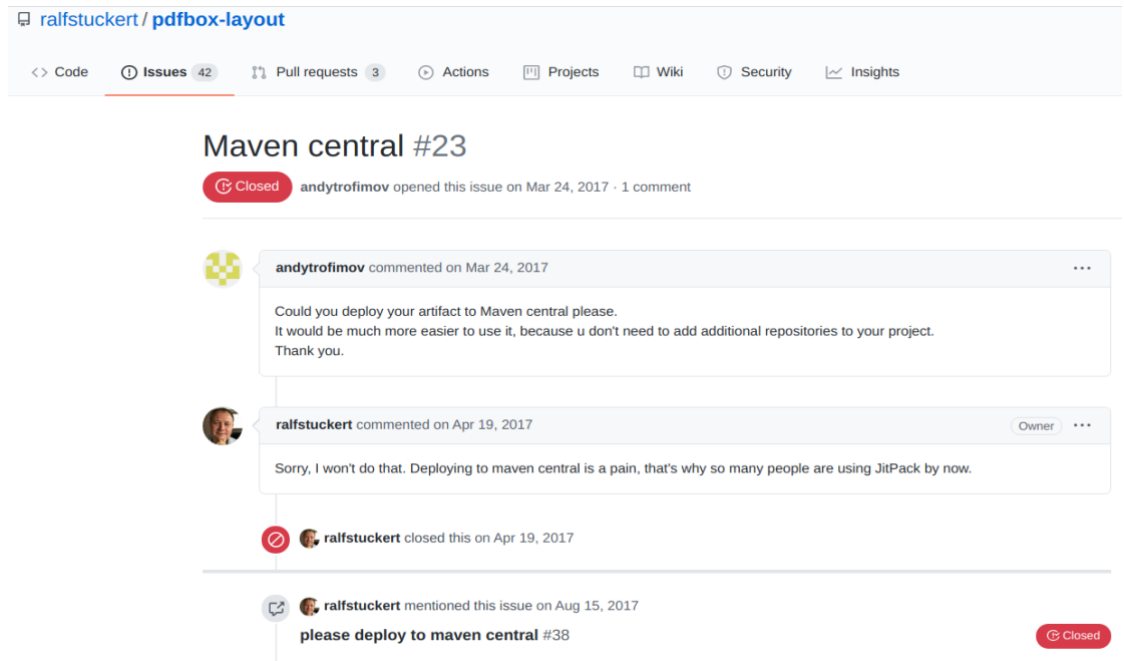
Figure 17. A screenshot of an issue on Github

The account ralfstuckert is the author of this library, and when someone asked him to deploy this to maven central, he said he is not going to do that.

I discussed with my boss, and I need his permission to upload this new dependency to Nexus repository before I can use it. He had to do it manually. For me, it was just full of confusion, like why we have to upload to Nexus repository but cannot use like how I always do with school project. I found some interesting information on the internet. "Nexus is called repository manager. It allows users to proxy, collect, and manage the dependencies so that users are not constantly juggling a collection of JARs. It makes it easy to distribute the software. Internally, users configure their build to publish artifacts to Nexus and the product then become available to other developers" (OBrien, 2010).

Therefore, for big cooperation like KPMG, we need a repository manager, and we use Nexus. After adding dependency to pom.xml file, I can import the library now and use it smoothly.

This is what I need to import to use for my service.

Figure 18. Services import from PDFBox Layout library

I will give quick introduction to apply the library. A tiny layout library on top of pdfbox. Main features are word wrapping, text alignment, paragraphs, pagination.

Figure 19. A list of services from PDFBox Layout library

These are a list of services this library can support from creating column for pdf, or adding links, margin, mark-up for the PDF Page (ralfstuckert, 2017).

3.8   Observation week 8 (December 14 – December 18, 2020)

Monday 14 December 2020

Last Friday, I completed a ticket and moved it to **Ready to review**. However, in the morning, the reviewer asked me to rebase from master again because there is conflict in my pull request. The reason is that last week, there is a pull request had been merged to master, I need to rebase from master to get the most updated codes and solve the conflict first before pulling the request.

After reviewing, she added some comments on my pull request, mostly about coding convention.

Here are some of the comments I got from the reviewer:

1. Always camelCase in Java
2. Fix indentation
3. Rename the variable to make it more meaningful

She also commented that I have some lines of code keep repeating, so she asked me to declare it as a constant and reuse it.
I fixed all the comment and let her review again.

Tuesday 15 - Wednesday 16 December 2020

My ticket is moved to **Ready to Test** from **In Review** section. While waiting for the testing, I reviewed other's ticket.

I realized there are some small things that he can improve better, for example this line of code:

```
if(customerAcceptance == null) {
        throw new InvalidObjectException();
    }
    if(personnel == null) {
        throw new InvalidObjectException();
    }
    if(searchResultsForMngtPersonnel == null) {
        throw new InvalidObjectException();
    }
```

What he can do is just add these in the same if clause

```
if (bundle == null || customerAcceptance == null || etc) {
throw new InvalidObjectException();
}
```

When he modified the method, adding more parameters into the method but forgot to fix the documentation part as well. Also, he needs to rewrite the test file because it looks confused.

The other developer who is testing my ticket cannot make it run to generate the pdf file, so she asked me to support her with the configuration. She got this message when try to build the docker.

```
Failed to execute goal on project esign-backend: Could
not resolve dependencies for project
com.kpmg.esign:esign-backend:jar:1.0-SNAPSHOT: Failed
to collect dependencies at
com.github.ralfstuckert.pdfbox-layout:pdfbox2-
layout:jar:1.0.0: Failed to read artifact descriptor
for com.github.ralfstuckert.pdfbox-layout:pdfbox2-
layout:jar:1.0.0: Could not transfer artifact
com.github.ralfstuckert.pdfbox-layout:pdfbox2-
layout:pom:1.0.0 from/to maven-central (http://
repo1.maven.org/maven2): Failed to transfer file:
http://repo1.maven.org/maven2/com/github/ralfstuckert/
pdfbox-layout/pdfbox2-layout/1.0.0/pdfbox2-
layout-1.0.0.pom. Return code is: 501 ,
ReasonPhrase:HTTPS Required. -> [Help 1]          15.37
```

Figure 20. Error message when generate pdf file

I checked if she had torn down the build file, `mvn clean install` the package and build again. She had done it all but there was a trick here. There are 3 `pom.xml` files in our project, one in frontend, one in backend and one in a common folder. Normally, we only do the command line `mvn clean install` in the common folder, and because I added the dependency in backend folder, so she had to install the maven package in backend folder as well.

After fixing all the configures, she tested successfully the ticket and also satisfied with what the new library can do. The ticket had been moved to **Ready to UAT testing**.

Thursday 17 - Friday 18 December 2020
In the early morning, I received an invitation mail from a DevOps developer in my team asking for a quick meeting about SonarQube. I haven't met him before because I haven't been in same project with him. However, my boss, who also directly interviewed me, has known that I used to work with SonarQube before, so he asked the DevOps guy, who currently set up the SonarQube in our Jenkin, ask

me to test about the new set up and share my experience working with SonarQube.

After the meeting, I continue to work on other ticket in the backlog. There are 3 tickets had assigned to only me. The reason is that there are some features in the product design which have been skipped to implement because of the complexity and time-consuming. However, since I applied new library, those features become possible to implement. I took the responsibility to implement those features.

During the development time, I faced some small problems with extract data from object array because the structure of Array and Stream in Java is more complicated than in JavaScript. I also applied some regEx pattern to get the desired data. In my previous company, I had worked same project with a colleague who is master of regEx, so I also got some reference from him.

Week 8 evaluation

This week has passed smoothly and productively to me. I feel more confident and feel more useful and engaged in the team after 2 months working here. I am also very thankful for what I have learnt in my previous experience, which helps me a lot during my current development time. The thing I want to share in this week is the SonarQube.

SonarQube is a tool to automatically static code review, inspect bugs, vulnerabilities, and code smells in the code. It can integrate with the existing workflow to enable continuous code detection across the project branches and pull requests (documentation, n.d.).

There are currently 4 editions for SonarQube available:
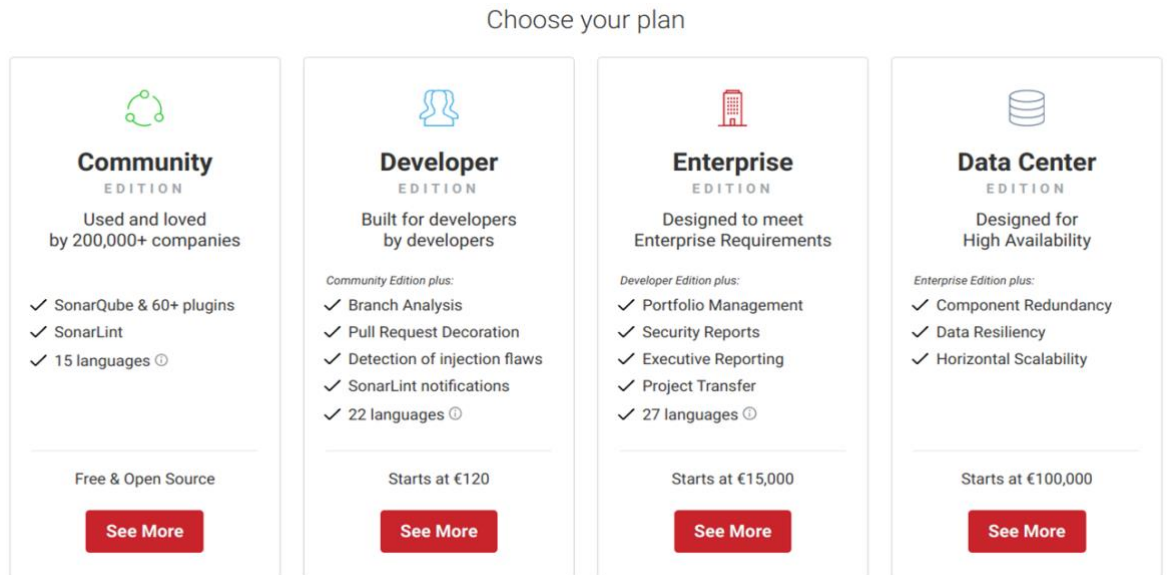
Figure 21. Four types of SonarQube editions

I was lucky because I had chance to use the **Enterprise** edition in my previous company.

Currently, in our team, we only set up and try to test with the free edition, the **Community** one.

I must say I was so fascinated about SonarQube when I first worked with it in my previous company. And I would love in any company where they also set up SonarQube.

Some benefits of SonarQube

- Sustainability

- Increase productivity

- Quality code

- Detect Errors

- Increase consistency

- Business scaling (Lomror, 2020)

I will show some examples, some SonarQube lint rules for typescript because I used TypeScript in my previous project.
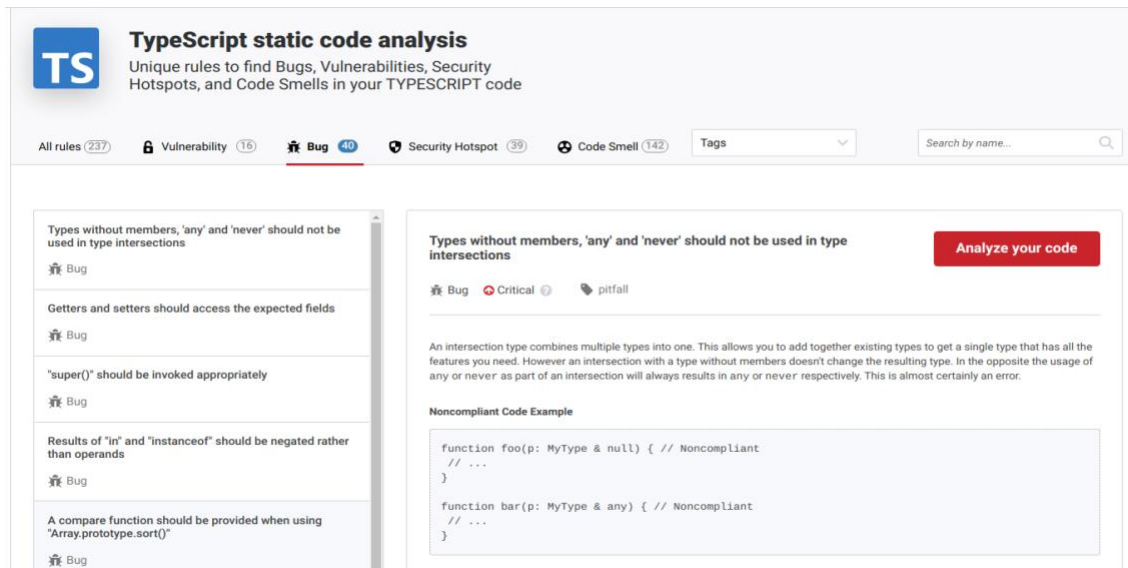
Figure 22. Some lint rules of TypeScript in SonarQube

Source: https://rules.sonarsource.com/typescript

They have some levels of how severe your mistakes in the code are. The highest sever level is vulnerable, in this case if you cannot fix the bug, the file cannot be built. For example, if the pull request has some vulnerabilities or bugs from SonarQube, the developer has to fix them all before his pull request can be merged.

3.9   Observation week 9 (December 21 – December 25, 2020)

Monday 21 December 2020

In this week, I moved to another project because the sprint of the previous project has ended. This new project is working with React TypeScript in the frontend and Node TypeScript in the backend.

I actually spent a day setting up the project. This project is quite new comparing to the previous I worked with. There is no Wiki page for these configuring steps. I have to communicate a lot with the scrum master also the senior developer in the team to do step by step the set up.

Tuesday 22 December 2020

We were supposed to have a daily meeting at 9.30AM every morning but other developers are already on their holidays therefore, I and the scrum master all agreed that we will have a meeting in the afternoon to talk about what I will do next week, because I will be the only developer who is at work.

In the meanwhile, I was working on 2 small tickets which have only 2 story points each and they are actually linked to each other. The ticket is about filter the survey result based on the close day of the survey and order them based on the open date. So programmatically, I have to sort the object based on the value of one specific property in that object. There are many ways to implement but I tried to make it short and keep the codes clean. I found some nice techniques from other developer who shared on Stack Overflow.

Wednesday 23 December 2020

I have a paid day off today as a Christmas gift from the company.

Thursday 24 – Friday 25 December 2020

Christmas holidays.

Week 9 evaluation

This week has passed quickly, I only worked 2 days, but I managed to set up the project and finish 2 tickets in the backlog and I feel quite satisfied with it. This week I have a chance to work with TypeScript again after 4 months since my previous job. I studied JavaScript for 2 years, doing my projects with React, whenever I want to learn TypeScript, my friend always said that I should learn more about JavaScript because TypeScript is so complicated and inconvenient. As a fan of clean code and writing maintainable code, after working with TypeScript, I think this language is so beautiful and not so hard to learn as my friend said and I recommend every JavaScript developer should learn and apply it in their project.

TypeScript is a language which builds on JavaScript, often called superset of JavaScript, one of the world's most used tools, by adding static type definitions.

In TypeScript, all the variables and objects need to pre-define, providing better documentation, and allowing TypeScript to validate that the code is working correctly.

Writing types can be optional in TypeScript, because type inference allows you to get a lot of power without writing additional code (typescriptlang, n.d.).
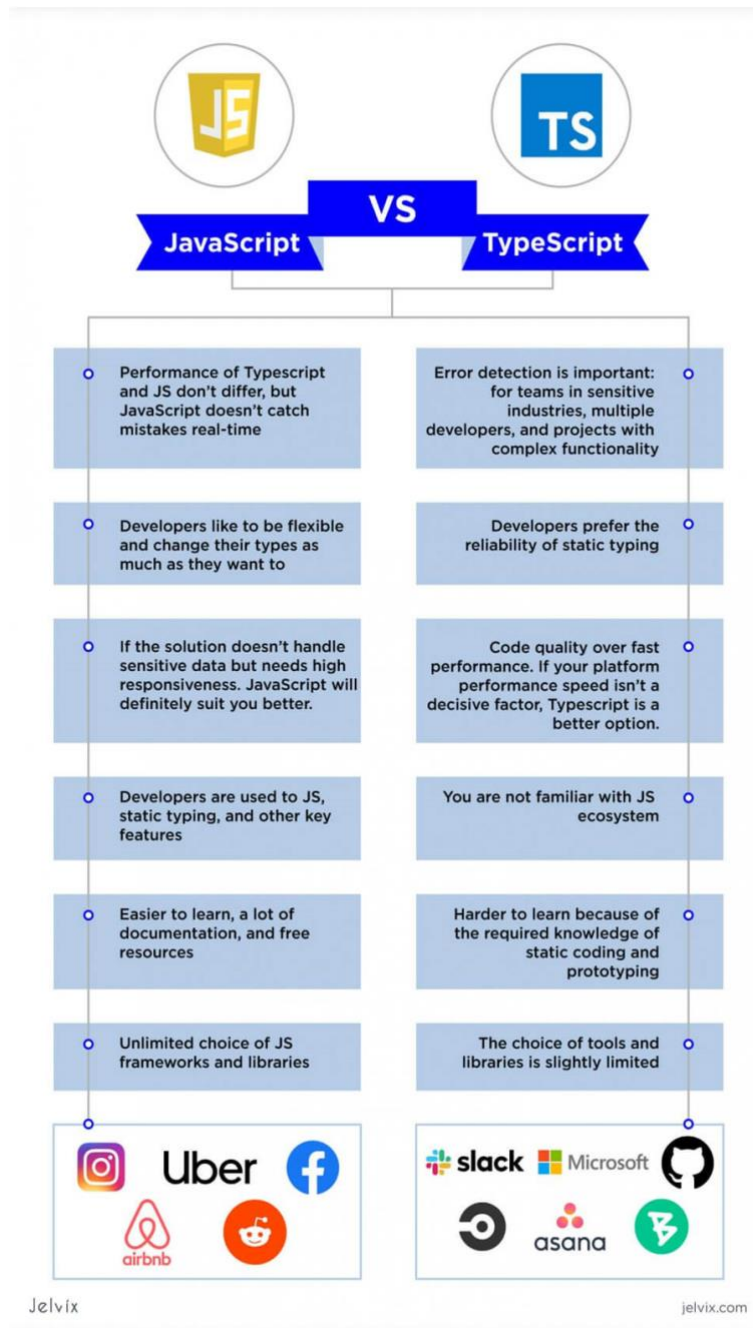

Figure 23. A table compare JavaScript and Typescript

Source: https://jelvix.com/blog/typescript-vs-javascript

The chart above is some comparisons between Typescript and JavaScript.

3.10 Observation week 10 (December 28 – January 1, 2021)

Monday 28 December 2020

Today, I work alone with an 8-story points ticket which is so difficult and complicated in the product for managing invoices named MyFinance. MyFinance is a system that fetches a CSV transfer file produced by SAP every day. This CSV file includes information about all KPMG engagements and is used to display engagement manager's information about the invoicing status of their engagements.

The requirement is we want to have a scheduled job in place that does the following every day:

- o Reads a CSV file (such as attached) from a predefined location
- o The CSV file only includes rows that have changed in some way, so we will need to replace the changed rows in the full data
- o Turns CSV data into JSON format using the csvtojson package
- o Passes the JSON form data to the invoice survey and displays the latest data to the user
- o The data is parsed correctly so that numbers are numbers and dates are dates etc.
- o Implementation details

We will have to think about at least the following as we plan the implementation:
How is the data fetched and stored the first time? (API create endpoint)
How is the data compared with the delta file fetched every day? (API update endpoint)
How to implement a scheduling crone job? (in a separate docker container)

Tuesday 29 December 2020

Today I have an Open Performance Development meeting with the Head of Application Technologies department. In general, in the meeting we discussed about what is my plan in next year 2021, what I plan to achieve.

After the meeting, I came back to analyse the ticket and try to find solution. However, there are so many unclear points in the code that need to refactor as

well as structure again. Because I worked alone and got stuck, I feel so much less motivated.

Wednesday 30 - Thursday 31 December 2020

After considering carefully, even I was assigned to finish the 8-story points ticket before new year holiday, I believe that there are barriers and blockers from lacking information prevent me completing the ticket. I decided not to spend any more time being stuck here. I moved on other ticket in the backlog.

I will create a custom reminder for the user and send them to the email when they register the invoicing management. The first thing I should do is adding a new email template for this service because currently, there is no email is sent when the user clicks to register the service. After that, in email service, we already have a function sendMail for other purpose with receive invoice ID and list of recipients as input. However, to reuse this function in the reminder service, I need to modify the input, that the function also receives reminder custom text as an input. In reminder service, I just need to call this function and pass the value I get from the form in front end.

The logic in backend has been built, and I continue to modify the front-end UI as well. It took me few hours to finish coding and few hours to test the feature I built.

Friday 1 January 2021

New Year holiday

Week 10 evaluation

It was quite intense at the beginning of the week but in the end, I can handle the struggles and this week is a relatively good week for me.

During this week, I learned a lot about how to manage my time and tasks to not get blocked by any difficult task. When I spent many days building the features myself, my boss recommended that I should not spend more than a day, struggling alone on one single task, always seek support from other colleagues. He suggested me to raise my issue whenever I want because it also reduces the

cost for the company. However, I just get into this project and I was the only developer because the other two developers were on holiday. There was so much information that I need to make clear, but I couldn't ask anyone. I was supposed to finish one ticket at the end of this week. After considering carefully it, I decided to work on other tickets, not to get stuck in that ticket which I know that it is impossible to complete by myself.

Another highlight of this week is a meaningful meeting with my boss. The meeting lasted in 1 hour and often happens in the last week of the year. Normally, in our team, we have this session 2 times a year, but since 2021, we will have it 4 times a year every quarter.

So according to the definition in the company website, KPMG Open Performance Development (OpenPD) is the global approach to supporting high performance and developing people. It defines the performance development activities and gives clarity on what high performance looks like at KPMG – someone who delivers impact seeks growth and inspires trust in all they do. The OpenPD activities include regular conversations between team members and PM setting and updating focused and stretching goals throughout the fiscal year seeking and giving feedback on a regular basis as a result of the former, a year-end review with no surprises. From the fall of '19 onwards, KPMG used a portal named OpenPD (SAP SuccessFactors) as the performance development system. The system supports employees in the performance development activities mentioned above, allowing them to set and update goals, record conversations, ask and give feedback and complete their performance review.

This performance review has 5 levels to grade. A normal employee who reaches all their goals will get 3 as an average. 5 is the lowest grade, if you are doing good, better than your set goal, you can get a grade of 2. Grade 2 is good enough to get a promotion or increasing the salary rate. The grading system is applying globally for all KPMG in the world.

One of my goals for next year is getting a Cloud certificate, currently, in KPMG training in-house, there is an Azure certificate. I will take this certificate as a goal for next year.

In general, the meeting was quite comfortable, it was just a conversation between me and the boss to help me grow better next year. After the meeting, I accessed the OpenPD system on the company's internal website to modify my goal and submit the final to the manager.

4 Discussions and conclusions

At the beginning of the job, I set 3 goals, they were increasing professional knowledge and training, quickly adapting to a new international working environment, shifting my focus from frontend to backend development. After 10 weeks of working in the new company, I have reached all the goals.

I started my career as a software engineer more than 1 year ago, in October 2019. Compared to that first job and this position, I can see significant growth in myself. At that time, I was so panicked and self-skeptical about my ability. I always feel that my knowledge in school was not enough to meet the job's requirements. Every day at work had passed extremely stressful for me. I was afraid to ask for help from colleagues because I thought I would bother them, and it was my responsibility to overcome the difficulties in my time. In the end, this doesn't lead to a good result.

However, after 1 year of working in 2 companies, I prepared myself with some skills that help me a lot in this job. I am confident to communicate with other colleagues for pushing the speed of the development process. I am not afraid to seek support because I was not supposed to know everything, and not feel ashamed to ask questions. After all, it can save time and cost for the business as well. Researching skills and fast learning by doing are also very essential skills that I picked up during working. I will never feel ready if I just learn from school and think that I can use them at work. In reality, what I have done at work is less than 50% of what I learned from the university. The useful skills at work that I have been practicing in the university are problem-solving skills and researching. I think every programmer has to agree that once you get a problem, you spend 90% of your time thinking about how to implement it and only 10% for actually coding. Coding is not writing an essay; we don't need a long answer. The best solution is a short and clean one. Debugging skill is also very important. Personally, I spent 30% working time on debugging. Thanks to the knowledge I have learned in university, I can debug quite quickly and save a lot of time.

Programming skill: I also recognized that once you have strong fundamental programming skills in any language, it will be really quick to work and learn other languages or frameworks. I have become more eager to get to work with a project that has technologies I have not learned or worked on before. This is a very good chance to broaden my knowledge and be flexible, be more valuable at work.

Writing clean code: One other thing that I learned a lot from working is writing clean code. This is extremely important, especially in a big company. Normally, in a big company, the code base is really large and is a result of many developers. That's why it is a high priority to keep the code maintainable, easy to read and understand, also a high level of security. Those experiences and skills are gained by working with experienced developers, who can review my codes and give critical feedback. With an open mindset and the will to learn, I was really focused to absorb these valuable pieces of knowledge.

Git skill – version control system: These skills I already learned from university, but I can only understand and use them properly when I work in a big development team. In my first job, I worked on a project with another developer, I can learn about creating a git branch, send a pull request, and merge. Because only 2 developers were working on a project, and we worked on different features, therefore there is no conflict in the code. However, in the second company, I worked on a project together with 6 other developers, and we often modified it in the same file. I have learned a lot about solving the conflict in git after many times I mistakenly deleted others' codes. In this current company, I learned about using git rebase instead of git merge and I am also more confident to use git reset when I need it.

Testing skill: The testing skill I mention here including writing unit tests or manual tests. I have not learned to write coverage tests when I studied in university. I picked up the knowledge while I worked on a project which has over 80% test coverage as a requirement. I started to learn the unit test framework based on the language I worked with such as JUnit and Mockito for Java, and Jest, Mocha, or Jasmine for JavaScript. I also observed how the tester worked in my previous job, learned some good tips from her, and had better skills when I have to be a tester as well in this new position.

Writing a thesis as a diary turns out to be a great experience and brings many values for me. As becoming a programmer, a developer is a long continuous learning journey, I have been noting down what I have learned in an online diary website since the first day at work. However, I normally just noted down the short syntax, very quick tips, in general very technical documentation. This is a good chance for me to not only note down but also analyze further the issue. I have a chance to look back on myself have been growing mentally from week 1 to week 10. Many good lessons have been learned. For example, I understand that the work is not always fruitful, but it does push me to move forward. I sometimes feel I wasted the working time and achieved nothing, but the experience was priceless. I learned some different things every time I failed. And those lessons grew me into an experienced developer.

I feel grateful that the team understands my expectation and gave me more opportunities to work on the backend more than the frontend. It had some difficulties and full of challenges at first, but it was how I have grown. In the future, I will have a chance to join a project that works a lot with Cloud technologies, because my team is moving all the in-house servers to store in the cloud. I will take advantage of this opportunity to get my Cloud certificate and sooner or later, I will focus to be a Cloud developer.

5 Bibliography

blog, C., 2020. *The Main Phases of The Scrum Project.* [Online]
Available at: https://intetics.com/blog/the-main-phases-of-the-scrum-project
[Accessed November 2020].

documentation, s., n.d. *sonarqube documentation.* [Online]
Available at: https://docs.sonarqube.org/latest/
[Accessed 2020].

https://www.atlassian.com, n.d. *Merging vs. Rebasing.* [Online]
Available at: https://www.atlassian.com/git/tutorials/merging-vs-rebasing
[Accessed October 2020].

IBM, d., 2015. *OWASP top 10 vulnerabilities.* [Online]
Available at: https://developer.ibm.com/solutions/security/
[Accessed October 2020].

Ilyukha, V., n.d. *The Comparison of Typescript vs JavaScript, Functionality and Use Cases.* [Online]
Available at: https://jelvix.com/blog/typescript-vs-javascript
[Accessed 23 December 2020].

Kouperman, I., 2020. *For a React app theme, is there an advantage of material-ui makeStyles vs. SCSS architecture?.* [Online]
Available at: https://stackoverflow.com/questions/59696613/for-a-react-app-theme-is-there-an-advantage-of-material-ui-makestyles-vs-scss
[Accessed 12 November 2020].

Lomror, K., 2020. *Sonarqube: What it is and why to use it?.* [Online]
Available at: https://www.loginradius.com/blog/async/sonarqube/

MaterialUI, n.d. *@material-ui/styles.* [Online]
Available at: https://material-ui.com/styles/basics/
[Accessed November 2020].

Memon, Z., 2020. *Apple Compatibility macOS vs. Ubuntu.* [Online]
Available at: https://linuxhint.com/macos_vs_ubuntu/
[Accessed October 2020].

OBrien, T., 2010. *"Why Nexus?" for the Non-Programmer.* [Online]
Available at: https://blog.sonatype.com/2010/04/why-nexus-for-the-non-programmer/
[Accessed December 2020].

Overflow, S., 2012. *What is the proper REST response code for a valid request but an empty data?.* [Online]
Available at: https://stackoverflow.com/questions/11746894/what-is-the-proper-rest-response-code-for-a-valid-request-but-an-empty-data
[Accessed November 2020].

ralfstuckert, 2017. *Github.* [Online]
Available at: https://github.com/ralfstuckert/pdfbox-layout/tree/master/examples
[Accessed 2020].

Stackoverflow, 2009. *Difference between "git add -A" and "git add .".* [Online]
Available at: https://stackoverflow.com/questions/572549/difference-between-git-add-a-and-git-add?rq=1

Taylor, K., n.d. *WireGuard vs OpenVPN: Detailed Difference Between the Two.* [Online]
Available at: https://www.hitechnectar.com/blogs/wireguard-vs-openvpn/
[Accessed November 2020].

typescriptlang, n.d. *typescriptlang.* [Online]
Available at: https://www.typescriptlang.org/
[Accessed 2020].

visual-paradigm.com, n.d. *What is Sprint Retrospective Meeting in Scrum?.* [Online]
Available at: https://www.visual-paradigm.com/scrum/what-is-sprint-retrospective-

meeting/

[Accessed December 2020].

visual-paradigm.com, n.d. *What is Story Point in Agile? How to Estimate a User Story?.* [Online]
Available at: https://www.visual-paradigm.com/scrum/what-is-story-point-in-agile/
[Accessed November 2020].

www.iana.org, 2011. *Hypertext Transfer Protocol (HTTP) Status Code Registry.* [Online]
Available at: https://www.iana.org/assignments/http-status-codes/http-status-codes.xhtml
[Accessed November 2020].