IT project Report

Contribution to the team project

Before this project, I have worked in a start-up company that develop a social communication platform as a software engineer for almost 2 years. Therefore, I have some prior knowledges and experience for building an app from scratch. And I’m the only one that with this experience in the team. So, I lead my team members to work on this app development. Our team has 4 members, me, gilbert, chenghong and erya. This is a very code-oriented team, for this project, in total, we wrote about 10 ~ 20 thousands line of code to build this app. Everyone has the coding assignment. At the beginning, I set up the repository and all that require for the project, including database, storage and authentication as well as the app which is in iOS. After I finished setting up stuffs, I assigned tasks to my teammates, and sometimes help them with some difficulties if they encounter, this is the first time for all of us using swift, the language that used to build an iOS app, so there is a lot to learn, and I have used Object-C before so it will be easier for me to master the language, and then try to teach my teammates. Besides that, I also be responsible to build the app storage for the first few weeks, in order to store the media that the client upload to the app, both on the cloud and local, so the client will retrieve their data fast and safely. After I finished the building of app storage, I drew a graph that shows the structure of our app, in order to give a direction to the team how should we proceed in term of coding structure. Afterward, we don’t really have any job specification, I just keep coding the part that we need to be done and helping my teammates with their problem.

Peer Feedback:

Chenghong Zhu:

At the beginning, he’s the one responsible for doing authentication of the app, for both backend and frontend. From the overall view, he’s a good coder, working hard, able to finish the task assigned and always on time. As I mentioned above this is the first time that for all of us using swift, however this doesn’t slow him down from his coding. I’m quite impressed for that. After the authentication get done, he’s a co-operate with other team members, mostly me and erya, to help with some of the hard part. The thing that I am most impressed by him is that he never drag on anything, as long as there is something need to be done, he will go do it immediately, which is one of the reasons that helps us to push our project much faster than I expect. Therefore, there is not much to complain about him, he is a good team member that I would like to work with again. One recommendation that I could give him is that, sometimes I feel that he is satisfy with bare minimum. If there is a requirement, he will try anything to fulfill the requirement, which is fantastic, but then sometimes he will only aim for the requirement without thinking about what this requirement might affects other stuffs, or how should this requirement be extended. I guess in some cases this is not a big issue, especially in schoolwork, because you will only receive the requirement once from the assignment, you don’t need to think beyond that. However, it’s almost impossible that you will receive all the requirement at once from a project in the workplace. I think if he overcome this, he would be an excellent programmer.

Quality of work: 4/5

Cooperation with team: 5/5

Erya:

She is the UI/UX designer plus coder in our team. This should show how amazing she is. Not only she designer how the app’s looks, and using PS and AI to help us to make the image we need, she also code in the app. She is also a quick learner, not only she never used swift, she is the only one in the team that didn’t take the software modelling class, which is quite important for doing an app using an OOP like language to build a good structure. She is a nature for this. In addition, when she wants to do something, she will always download a sample from other people’s git and study it to make it better. She is the only one in the team that I am confident to not need check the code after she finished her part. One recommendation for improve, I think for the role she has in the team as the UI/UX designer plus coder, she did a pretty good job, especially this is the first time she works in a project in this scale. Maybe in next time, she could try more roles as well to master all the coding skill in general, to become a more diverse engineer.

Quality of work: 5/5

Cooperation with team: 5/5

Gilbert:

This is the one that I always struggle with. He is responsible for the client requirement, graph drawing, and database in the team. In term of skill of coding, I believe that he is just as good as the other two. However, in this project, he doesn’t show a lot of interest in working. Sometimes, we need the functions to put stuffs to database, we need to wait for a week, until we finally can’t wait anymore and ask him to write the code on the spot, and it’s actually not a lot of code to make those functions. Fortunately, the code he wrote, usually doesn’t have bug, we don’t need to ask him to write again once he’s done. I understand that he has more works in other subject, like he’s tutoring the database subject in this semester. That’s why I ask him to do the database for us, it relates to what he’s tutoring now, and this doesn’t take a lot of work. I know this might because all the rest of us work pretty hard on this, and he has a little bit hard time to catch up. However, we only expect him to put like 50% works compare to the rest of us, then this project will move without delay, and yet he fails that. At the first few weeks, he won’t even write any code without me supervising him in the meeting, I could have used those time to build more functions for the app. Besides that, he even tried to convince the other two teammates that to do less in the project, so that he can look better, that’s just pure selfish. And I know that he has lied to us a couple times to get out of the middle of a meeting, working in the middle of a task then suddenly decide to drop it and Erya has to finish his part. One recommendation from me is that, if you have to work on something, maybe you don’t like the project for some reasons, please think about the team, at least do the bare minimum so the team’s progress won’t be slow down. I guess a sense of responsibility is important for a software engineer.

Quality of work: 3/5

Cooperation with team: 2/5

Lehan(myself):

I am the leader of the team, and plan and code the project structure through the whole the project. I think in the coding part I nailed it as usual, we solved every technical issue that we encounter and build out the product eventually. And I worked hard to push the project forward. One thing that I think I can be improved is that, I should do more documentation while I am planning the project, instead of just think It inside my head. Because there was some mis-planning happened during the project, we only realized that’s not what we want after we already finished building the component. Then we need to tear down the whole component and re-do it again, which wasted a lot of time.

Quality of work: 5/5

Cooperation with team: 5/5

Reflection on the Subject:

Although I have worked in a team of engineers on projects before, this is my first time to plan the structure for the project and assigning task to the team members. It wasn’t easy, I have to know and understand each of the team members’ strength to assign them to the task that they fit.

Reflection/What have I learnt from the second guest lecture

In the second guest lecture, they talked about what are the sign of a good engineer/coder looks like, and what they are looking for a good teammate. First, they talked about what are the architecture requirement for a regular software. It will require the ability to deliver regulatory changes, as the clients’ demand will keep changing, and we need to be able to deliver the new change in a relatively short amount of time. Thus, tear down the project and create a new one is not feasible. The code needs to have a flexible structure which will be easy to add changes in the future. Also, the software engineer has to have the ability to foresee the future about which part of the code will likely be requested to add new thing.

In the lecture they said: the real-life software development looks like an iterative. Whenever there is a poor decision, put a pin there and they will be revisited when they have more information, and try to fix it. It will be difficult to do that if the software has a poor structure subject to change. The most important thing for a software engineer is teamwork. It’s inevitable that software engineer needs to work in team. Every awesome result is polluted by the thought of many mind. The software engineers that work in a same project have to have a consistent code style, so that they will be easier to understand each other’s code. Beside being a team player, as a software engineer, the software skill is also very important, and the skill is not about how many language/ what languages you know. From the guest lecturer, they said that they will be more focus on the core skill/ critical thinking skill and the ability to solve problem when they are recruiting. As the technology is evolving rapidly, being good and comfortable on a certain area/language is not going to help you to solve problems. A good engineer should be able to adapt the constantly changing world quickly and never stop thinking. In order to stand out during a job interview, we need to demonstrate that to the interviewer, rather than just pure raw coding skill. It’s more about the agnostic skills than specific technologies. In the interview, they will also try to see if you are good with teamwork, to see if you can fit in their team. When they talked about the sign of a good team member, the member should be a curious, question asking person, constantly asking why. Because when you are asking, you have a higher chance to get new information and this might help you become easier to solve a problem or making a solution better. And the person should be humble, play well with other. The ability to connect the dot for the team is also an important sign for a good team member. And the last but not least, this is an obvious one -- the person should do his/her best on his task.

Reflection/What have I learnt from the first guest lecture:

In the first guest lecture, we have a person that founded his own company. He mainly talked about communication as a software engineer, and how to have an idea, and how to build it. He mentioned that even though, software engineers are working with computers all the time, communication is still an important skill to them. As a software engineer, you need to work in team a lot, therefore, you need to know how to communicate with your team members in order to collaborate and talk with each other about their idea. Also, you have to know how to communicate with your clients, in order to figure out what they want and how to make a product that suitable for their requirement. For example, in this IT project, we have a client that wants to be able to saves their memory within their family and possibly pass down to generations. We need to communicate with the client all the time in the process to see if she is happy about our current design and explain to them what functions that our product will have, so she has a clear idea about what are we going, and we also can get feedback that can help us adjust our direction while we are working on it. Also, we have totally four members in a team in this project. And working together is not easy, especially this is the first time for most of us working as a team in a project, and there are a lot of difference between us. We have to have a constant communication to let each other stay on the same page and push the project forward. Also, in the guest lecture, he talked about commenting, the most boring but important thing while coding. He had an example that his brother as the CTO of his companies that doesn’t want to know anyone that don’t do commenting on their code. It’s true that it’s very upsetting when you can’t understand the code that your teammates is working on, you need to find the person that wrote that part and explain to you otherwise you can’t just simply add things on the code that you don’t fully understand, and this is time consuming. Commenting on code is actually very useful for the team and sometimes even for the person who wrote it. At last, he talked about how to generate an idea. He mentioned that to find a problem that you cannot imagine an answer for but you know people want to find it and it inspires you, and while you thinking about an idea, remember to have rest, constantly working without sleeping or eating is not health for both your body and mind, most of the time, when you are exhausting, the ideas you generated, are useless. In addition, for students in the university, the best practice to learn and experience the real-world project, you should treat the university as your client and deliver your best work all the time.

Reflection/What have I learnt from the third guest lecture:

In the third guest lecture, it mainly talked about job hiring, and some ethical consideration in the field of computer. For example, the ethics of electronic privacy issues, and the moral responsible software engineering. And also, in the network, media, the Facebook, fake news and ethics of censorship. Especially, we are living in an era of Ai, it’s very important that we need to make people feel safe and protect their privacy. About job seeking, the lecturer also gave a lot of great advices. She told us that, there are 3000 new job seekers added every year, 15000 jobs are advertised every year. Also, the statistic shows that nowadays, there are 30 applications for a job in average, and it’s a double of number compare to six years ago. Therefore, the competition between job seekers is getting bigger and bigger. You have to have a good strategy in your job seeking process in order to stand out and get the job that you want. After that, she shared some tips about interviewing for a job. We should treat every interview we go to seriously, before you go, you should have a mock interview, think about what questions that the interviewer is going to ask, and make sure you have the answer for all of them. The guest lecturers said that the person that doesn’t know all the questions and the answers to those questions before he/she walked into the interview, it’s not going to get hired. Practice makes perfect. Also, during the interview section, you should pay attention on the questions on where they are focusing on, are they focusing on the past? Or are they focusing on the future. And base on what they are focusing on, to give them the answer they want. And if you have questions for them after the interview, like if you want to know when you can hear the reply from them, or how do they think about you during the interview, or even the pay, the bonus etc. and you should show confidence when you asking about this question. And the last part, when you have been offered, remember to check your contract clearly before you sign anything. You should take your time and bring the contract home to check if all it said fulfill your requirements or the law requirements. You should know the market average salary and the national employment standard, to ensure that you been treat fairly. Other than going to the job-finding website and go to interview, you should have an LinkedIn account or a personal website that record your working status and what have you been accomplished so people will be much easy to find you when they are looking for a person to do the job that is suitable for you, sometimes this will save you a lot of time to searching for job, and it shows your value to the market. From this lecture, I have learned a lot of useful skill that I could use when I am looking for job. I would say this is a very valuable lecture.