**Transcript Summary – Interview of an IT Professional**

**YouTube recording video of the interview:** [**https://www.youtube.com/watch?v=6F4LdLAyZW4**](https://www.youtube.com/watch?v=6F4LdLAyZW4)

**Team: Omega Ruby:**

Darren Stephen Miral - s3858795

Malcolm Fai Hao Tsang - s3333925

Minh Hau Truong - s3859157

Van Thanh Nguyen - s3807407

Sisi Meng Shih Liu – s3855898

-------------------------------------------------------------------

0. Can you briefly tell us what prior learning/qualifications you had before you started this job?

---------------------------------------------------------------------------------------------------------------------------------------------

At university I studied BSc Computer Science, my university is famous for its strong mathematical background and can build specialized knowledge from the courses I have attended. It gave a solid academical background, not specialized in any current technologies, but taught the "big picture". Was still studying at university while working.

Before university I went to a special high school which was specialized in mathematics and informatics. I was always interested in this field, maybe because my father was a computer mechanic and I've learned typing on a keyboard before I learned to read. I could use the computer (and by use, I mean starting and playing with computer games) before I attended grammar school.

---------------------------------------------------------------------------------------------------------------------------------------------

1. Please tell us about your IT work. What exactly do you do?

---------------------------------------------------------------------------------------------------------------------------------------------

Ok. So, my situation is a bit special.

Currently I'm between projects so mainly studying the technology stack for it. It includes:

+ Taking online courses in technologies (for example AWS) what is heavily used at my next assignment. Did the course on Udemy.

+ Practicing some skills, I haven't used recently but will need,

+ Preparing the development environment on my machine per the requirements of the new job,

+ Reading documentation about my new employer's product and current architecture and studying topics I'm not confident around and will be needed, reviewing requirements

+ Thought I’d only do coding, but not like that, still must review codes and check it.

+ Clarifying functional/non-functional requirements of tasks (Speak with business side or architecture team) Huge part of the job, you need to know what needs to be built, otherwise problems will happen if you don’t understand anything.

As you can see my current situation is special, so I will speak about my previous projects and talk about what were my tasks.

So, I was in a software engineer role all the time. The number of tasks I had to do were rising with my seniority.

In the end my normal day-to-day tasks were:

+ Support: check incoming questions or bug reports

+ Mentoring fellow developers on functional or technical questions

+ Coding (of course)

+ Reviewing another people's code

+ Keeping in touch with product side when developing new components

You may need to learn a new tech stack based on next project.

Between projects need to keep learning new skills, new technical stack. Lifelong learning while working.

---------------------------------------------------------------------------------------------------------------------------------------------

2. Please tell us about the industry you work in.

---------------------------------------------------------------------------------------------------------------------------------------------

First job in the healthcare industry, I was working on an ECG software, Healthcare in cardiology, working ECG curves.

Later I have transitioned to the investment banking sector.

Involved in Investment banking industry for many years and then worked for a bank. Currently in between 2 assignments and will be in Education. You will learn a lot of industries.

You will learn to work with doctors in healthcare, bankers in banking for example. Software developers will learn a lot of industries in your career, which is really interesting.

---------------------------------------------------------------------------------------------------------------------------------------------

3. What other kinds of work do you have to do?

---------------------------------------------------------------------------------------------------------------------------------------------

My title is Software engineer.

Apart from coding? I think the most important tasks are clarifying requirements with the product side and support. Usually there is a team who does support but when they don’t know they turn to us. We can clarify if the concern raised is a bug or feature.

Software engineering is not just about coding and doing the technical part, we have to get involved in the product development, we have to learn the domain, understand how the user works and then make our suggestions and sell it to the client. It is the most difficult part.

But back to the technical part as I have already mentioned my job includes mentoring other team mates when they have questions, I must review their code changes and support incoming requests.

Mentoring help each other technical or functional. Part of clarifying requirements keep in touch with product side. We need to understand product code. Sometimes we know it better, incoming requirement is contradicting we need to make sure and work with product to better suit the client.

---------------------------------------------------------------------------------------------------------------------------------------------

4. Who are all the different people you interact with in your work? Please tell us about them

---------------------------------------------------------------------------------------------------------------------------------------------

Other developers - we work together, review each other’s code, decide on the technical solutions.

Architects - Tech leaders. I go to them with my technical questions, also they are the ones who guard the integrity and connectivity of the components. Most experienced, programming professional, guard quality of code and teams work together and synchronize work.

Product owners - We get the tasks from them and we discuss the functional questions and design problems we may find. Shielding us and first firewall in front of dev team.

Infrastructure/IT guys - Whenever something is not working, or we need access to a database or anything else then we call them and try to find a solution. Sometimes they find us too when they have trouble with - for example - deploying.

Client – Rare to have interactions. Some companies don't want to expose developers at some other places it is natural. Get shielded from them all the time. I personally had to jump on calls with clients only several times and had to personally meet with them a few times.

---------------------------------------------------------------------------------------------------------------------------------------------

5. Please tell us about your interactions with other IT professionals.

---------------------------------------------------------------------------------------------------------------------------------------------

Mentoring part of the job, otherwise new members can’t get on board. Part of everyone jobs. Can call it mentoring, but it’s simply sharing information. Specialized in something then you can share as part of team work.

If I specialize in something, turns out the project needs to move to another database SQL for example, then I will take lead and show the team if I am the most experienced one. That’s how it works.

---------------------------------------------------------------------------------------------------------------------------------------------

6. What about your interactions with clients or investors?

---------------------------------------------------------------------------------------------------------------------------------------------

Can be dangerous, if you don’t have social skills. Important to have a shield from the client so we don’t get called directly and get bombarded.

It's a difficult question because in my experience we developers are always shielded from the clients. We originally weren't allowed to participate in any meetings or calls with them but sometimes we ended up on calls when the issue they were facing was too technical the the business side couldn’t help them. But it has happened only several times.

It is good to sometimes meet with clients though. I’ve had contact with clients before and thus rewrote software from the feedback, so it can be good to speak to the client, solution better fit the client’s requirements in the end.

----------------------------------------------------------------------

7. What aspects of your work do you spend most time on? Please tell us about these.

----------------------------------------------------------------------

Understanding the task, and clarifying requirements, especially new in the job/industry.

I remember many times when I've spent a day with understanding requirements and the coding part took around 1 hour. After many years in 1 industry understanding requirement was easy, but I needed to work there for that long. Most challenging to build the right thing and to know what the right things is.

Coding can take long, not typically box product, more technical. Research fields like automobile for example designing new brake system for Porsche that is a research task, more academical and coding heavy and will take a longer time. If we are working for supplier for client, creating according to client’s requirements, then understanding the tasks is most important.

----------------------------------------------------------------------

8. Which aspects of your work do you find most challenging?

----------------------------------------------------------------------

Understanding the tasks, need to be clear and need to set right at the beginning.

If the developer doesn't understand the product and don’t get requirements correct, mistakes can happen, and you need to rewrite it again.

There are challenging technical problems too especially when a new architecture must be designed or a totally new component, and it can be super difficult too, but I think understanding the tasks is a bit more complex.

----------------------------------------------------------------------

9. Can you share an example of the work you do that best captures the essence of the IT industry?

----------------------------------------------------------------------

Nice story I’d like to share, my favorite example is a rewrite project which was partially led by me. There was a component in our software which was heavily used in conferences but was really slow and unstable. It was generating schedules which were printed and given to the conference attendees.

I was chosen to continue the work with that component and I had to learn it from scratch. It took a lot of time to understand it and I had prior experience before. We knew about this component before and it was very complex component and took a long time to understand it. We knew it should have been rewritten years ago but we were always told we don't have time for that. The technical debt in it was tremendous.

We were preparing for a huge conference which was also the big test of this component. Everything seemed ok. It was a nice Thursday evening when at the end of the day I packed my bag, took my coat, and said bye to my colleagues, when my architect has grabbed my arm. Tibi, don't go yet, I'm in a meeting, we may have to look at this component.

It has turned out the component stopped working. All of a sudden. Yesterday it was working, today: no and the client wanted to start using it next Monday. Thursday, Friday, Saturday and Sunday I went home around midnight. The component was built on webforms which I didn't have too much experience with, so I had to learn that architecture and build a solution which quick fixes the problem.

We came up with a temp solution for now, our only option. It was dangerous solution, but only option. Went to conference for support if anything goes wrong. It worked as it could be, not that well though and survived the week.

Most important part, is that we could provide proof to management it must be re-written.

 I could finally sit next to people who were using the component and see what they are doing and how. So, when we have arrived at home we could start redesigning the component and start a complete rewrite.

In this project, I took lead as I knew the most, started to re-design on how it should work. With product owners, we worked together in designing etc. I used my domain knowledge and experience and contributed to the design part too.

As lead, I gave out tasks, checking another people's work, mentoring etc. I was reporting to senior management about the progress and also did my part in coding. Built QA testing site and jump on call with clients.

In the end it was a huge success, we were praised by the client too.

Covered everything here on what a developer does, not just coding and may end up in other parts.

----------------------------------------------------------------------

10. To keep abreast with the changing technologies, do you need to enroll in additional courses to upskill even when working full time? How do you keep up to date and adapt with new technologies and trends?

----------------------------------------------------------------------

Whenever I join a new company, I always ask if they support new learning of courses and give time for that. Subscriptions such as Plural, LinkedIn, Udemy.

I personally have Udemy subscription, but companies may give to others as well. I usually learn from home and have a lot of free time. If you have family, important the company gives time, like 1 hour a day so you can spend for learning. Its important, also reading articles, new databases then surf the internet and read articles and learn by using.

In this perspective I'm in fortunate situation since I don't have a family, so I can learn and try new things in my free time. But if someone has kids then the only opportunity to learn is when working and I think it is crucial for every employer to secure study-time for the employees.

----------------------------------------------------------------------

11. What hobbies do you have that are IT related? Are there any clubs, memberships or subscriptions which you recommend to join?

----------------------------------------------------------------------

I like tampering with my Raspberry PI, build from an idea. I like to practice coding and algorithms, I like to read articles and books about topics I'm interested in, like machine learning, genetic algorithms, AI, distributed systems.

Just to give some advice, do a Pet-projects. Home grow it and it will give you experience. Interested in AI, build an AI at home then, useful for next potential work opportunities.

----------------------------------------------------------------------

12. IT is quite vast, there are various roles such as systems engineers, .Net/Java developers etc. If you could choose again, which role would you pursue and why?

----------------------------------------------------------------------

Absolutely be a developer again, like to create from nothing. With a computer you can do almost everything. Still very interesting field for me. Would stay as a software engineer for sure.

----------------------------------------------------------------------

13. In your opinion, what is the most useful skill set to know in IT? For example, SQL, JAVA Script etc.

----------------------------------------------------------------------

Knowing a tech stack, Java or .Net is a must. I wouldn't emphasize a technology though. I think the most useful skill of a developer is the ability to adopt. And for that you have to know the general principles behind these technologies.

But most important is to have the Academic knowledge, the algorithms, mathematics, data structures. This is what you can build on. You need to know it really really well. You need to know the design patterns, the patterns we use, they are programming language agnostics. Doesn’t depend on programming language, structure and algorithms are the same. If you can learn .Net stack for example, and need to work in JAVA, you just need to learn a new language. Rules and algorithms are the same.

For example, a job looking for JAVA developer, but would consider a .NET developer. Not a problem, transition to another should be rather easy.

Of course, you should know a techstack, but having a general professional knowledge is much more important. A developer should be able to learn a new programming language in several weeks. (Ok, not to the professional level, but to a good level.)

But if I would need to name some technologies then I would say know a technology stack deeply and know some other languages at a lower level.

For example: for .NET stack you should speak C#, SQL (MSSQL dialect) and know how the .NET framework works. But you should have some knowledge of HTML, css, Javascript, Typescript, Python, maybe ADA and some functional language. This way if you have some minor knowledge in these different style of languages at a new assignment where you have to move to for example JAVA it will be much easier because you will be in the possession of the 'general idea' or 'general knowledge' (what I have mentioned before) and will be easier to adapt to the new environment.

----------------------------------------------------------------------

14. Lastly, is there any advise you would like to give beginners wanting to become successful in the IT field?

----------------------------------------------------------------------

Yes. The most important things are:

[Learn, learn, learn!]

Learn a tech stack and keep eyes open.

I’m a .Net developer, and half a year ago I started to play with Python as I am interested in it. So, now I have general knowledge.

Learn the basics! And learn continuously. Subscribe to a site: Pluralsight, LinkedIn Learning, or Udemy. Find a good platform to study.

You don't use your academical knowledge about algorithms and or distributed systems? Keep your knowledge by watching courses about these topics of Pluralsight, LinkedIn Learning, Udemy or even on youtube with the MIT algorithm courses!

[Be a geek!]

This profession is - in my opinion - not a normal 9 to 5 work. It is a way of life. Be a geek! Like what you are doing!

[Learn to say no!]

We are always pressurized by the product side or the management. We are always told we are late. We are always pressurized to finish our current tasks before it can be done.

Learn to say no. For example, say, we can do this but this or that won’t be ready, then the product won’t be stable. Then product can choose what to decide to go with. Really difficult to learn to say NO, when it’s coming from senior management.