Technical Competence March:

For me, my projects revolve around specific software skills and my ability to use them properly. There are concepts and similar software that I have learnt and used during my study in Computer Science, however because software terms tend to be different, I generally still have to spend time learning new softwares for specific projects. However, for the role that I am given, 2 related technical competencies can be highlighted as having a major relationship with this role. Those 2 being System application and System comprehension, as defined by several studies.

The 2 competencies, being system application and system comprehension, represent skill sets that are tied together by the fact that they are related to computer systems, however are different skills entirely. System application being a competency is knowing how to execute tasks on specific systems. This is the aspect which requires study of specific pieces of technology to get a better understanding on how they work and how to work with them. Meanwhile system comprehension is the skill required to understand why the specific machines work. It is to study and analyse logic and reasoning as well as standardised methods that are shared between similar softwares. This competency is gained passively as you are more familiar with the theory and flow behind software.

An example of how this is applied to my work is working with router software. I have already gained a basic system comprehension behind routers because of previous courses, however I have no system application competency yet because I am totally unfamiliar with the software. I quickly gained that system application competency by exploring and following tutorials, but also gained a better understanding on how the logic and best practices are, so I can utilise the software in a more efficient way, as well as transition to other similar softwares easier since I understand what needs to be done in general.

Both system application and comprehension is difficult to get, requiring a balance of intense background study as well as practising those skills with real equipment. With softwares constantly upgrading, documentation, lessons, and tutorials of those equipment might also be different or outdated, so the system application side is especially difficult.

I resolve this by using system comprehension, understanding the flow and logic behind the software to apply to different versions of the same software. It requires a lot of trial and error, so I have been experimenting with the softwares that my projects utilise.

References:

Kollee, C., Magenheim, J., Nelles, W., Rhode, T., Schaper, N., Schubert, S., & Stechert, P. (2009, July). Computer science education and key competencies. In *9th IFIP World Conference on Computers in Education–WCCE*.

Figl, K., & Motschnig, R. (2008, October). Researching the development of team competencies in computer science courses. In *2008 38th Annual Frontiers in Education Conference* (pp. S3F-1). IEEE.

Clear, A., Clear, T., Vichare, A., Charles, T., Frezza, S., Gutica, M., ... & Szynkiewicz, J. (2020). Designing computer science competency statements: A process and curriculum model for the 21st century. In *Proceedings of the Working Group Reports on Innovation and Technology in Computer Science Education* (pp. 211-246).

Hernandez-de-Menendez, M., Morales-Menendez, R., Escobar, C. A., & McGovern, M. (2020). Competencies for industry 4.0. *International Journal on Interactive Design and Manufacturing (IJIDeM)*, *14*, 1511-1524.

Prabhakar, B., Litecky, C. R., & Arnett, K. (2005). IT skills in a tough job market. *Communications of the ACM*, *48*(10), 91-94.