Development Process Report: Temperature Sensor Simulation Project

Through this project, I developed a temperature sensor simulation system using .NET C#, XUnit for testing and Git for version control- as well as using FigJam as my Kanban board. I worked to apply the skills I learned through the ‘Learning Git and Github’ and ‘C# Unit Testing with xUnit’ LinkedIn Learning courses to build a functional console application. I was able to work effectively and though there is room for improvement, I was happy to be able to present a functional application.

My workflow was the same for each feature. I began by reading the feature requirements and drawing simple diagrams/notes to visualise what I was to implement. I would then research similar work, which helped me decide on what to write. For each feature, I created a dedicated branch where I would iterate development until I had it run successfully. After committing what I had so far, I then created separate test branches to apply tests. These were then written and committed in the same fashion. In review I found that I was not pushing to origin which impacted some issues discovered later.

The testing phase was approached with the aim of achieving good coverage. Using XUnit, I developed tests that verified basic functionality and also edge cases. I found this part of the project challenging, as I had never attempted implementing complete coverage before. The LinkedIn Learning course gave me a good foundation of knowledge and researching as I went for particular use cases was helpful. I am confident now on how to structure testing- and where to get information when I am not quite sure where to start.

I adopted feature-branch strategy in Git, which proved to be a big learning of this project. Unlike previous assignments throughout my ICT study, where I worked directly in program files without employing Git at all, 'separating concerns' into feature branches significantly improved my ability to tackle the project as it provided me more mental clarity and a way to roll-back when I had seemingly messed the whole program up. While I was able to build the project successfully, there are areas where I could improve my Git usage discipline. When reflecting the GitHub Network Graph, I see that I was not always accurate with my strategy- regularly pushing to origin main and making regular commits were steps missed for most of the project. This was particularly evident when working on user-input for the thresholds, where I had included everything in an overall threshold feature branch. While I did meet the core requirements easily, I found myself regularly breaking the whole feature while working on the user-inputs. If I had committed more regularly, I would have saved significant time as I was often getting lost in the code and trying to simply get back to the starting state. The way I was working showed a lot of value in Git; it's just a matter of refining my approach and perhaps revisiting the LinkedIn Learning course now that I have more experience to work with.

To conclude, this project provided me with clarity in complete software development. While I was able to meet the requirements, this project highlighted big areas of improvement, and I plan to apply these learnings going forward especially the Git workflow strategies that proved valuable.