

### **Unit 3 – Reflection:**

During Unit 3, I learned the importance of the Unified Modeling Language (UML) in software system representation. The main objective of this week was to understand UML as a standardised set of tools for the design and documentation of software systems before their implementation. As I progressed through the unit, I learned about the standard syntax utilised in UML and the different UML models relevant to the object-oriented analysis, design, and development process. Familiarity with the syntax was critical in creating precise and comprehensible UML models representing specific software system aspects.

By the end of the unit, I had developed an ability to recognise the syntax used in UML models and differentiate between different UML models used in various stages of the Software Development Life Cycle (SDLC). This knowledge is crucial for effectively communicating and documenting the design of software systems. The unit allowed me to apply my understanding of UML by designing UML models using open-source tools. This hands-on experience enabled me to gain practical skills in creating UML diagrams, such as class, sequence, and state machine diagrams, that are essential for representing different aspects of software systems.

As part of the activities for this unit, I completed ePortfolio tasks that comprised discussing the most suitable UML models for different stages of the Software Development Life Cycle. I also designed a state machine diagram for a washing machine using 'The Unified Modeling Language Reference Manual Second Edition' as a reference. These exercises strengthened my understanding of the course concepts and allowed me to share my findings in my ePortfolio. Through this practical exercise, I developed my modelling skills and applied my knowledge of UML.

In conclusion, Unit 3 gave me a robust foundation in UML and its application in software system design. I am excited to continue exploring UML in the upcoming units of the course, as these skills are invaluable in preparing the design document for the forthcoming weeks and effectively communicating the structure and behaviour of software systems.