**Learning Journal Template**

**Student Name:** Ting Lok Hui

**Course:** SOEN 6841 Software Project Management

**Journal URL:**

**Dates Rage of activities:** 10thNov – 18thNov

**Date of the journal:** 6ndNovember 2024

**Key Concepts Learned**

**Chapter 10:**

I have learned that requirements are categorised into two distinct types: functional and non-functional. Each type plays a crucial role in defining the software's characteristics. Functional requirements describe the specific behaviours and functions that the system must perform. Non-functional requirements address quality attributes such as performance, security and usability. Understanding these distinctions will help me create more comprehensive and precise requirement documents in future projects.

**Chapter 11-14:**

learned the importance of creating robust architectures that can accommodate change requests while maintaining compatibility with the original design. The top-down and bottom-up design approaches provided me with a strategic perspective on the decomposition of complex software projects into manageable components. I understand that software testing is no longer an ancillary activity but a vital quality assurance mechanism.

**Summary of the course:**

This course provided a comprehensive understanding of software project management, covering the entire project lifecycle and its associated principles. I gained valuable insights into defining and initiating projects, including the development of project charters, objectives, and scope. The course emphasised the importance of SMART objectives and tentative planning. I gained proficiency in cost and effort estimation techniques, including Function Point Analysis (FPA), COCOMO, and Wide Band Delphi, enabling me to create realistic project estimates. The course provided me an insight of the instruction on risk management, including the systematic identification, assessment, and response to risks using strategies such as mitigation, avoidance, and transference. Configuration management emphasised the importance of controlling and documenting changes with tools such as the SCCB. The project planning module focused on Work Breakdown Structures (WBS) and scheduling methods, while the project monitoring module introduced Earned Value Management (EVM) for tracking progress effectively. The course also placed great emphasis on the importance of project closure, outlining key activities such as documenting lessons learned and archiving deliverables. I gained insight into a range of software development models, including Waterfall and SCRUM, and how they can be applied to different projects. I also learned to distinguish between functional and non-functional requirements, which improved my ability to analyze requirements. Finally, I explored software architecture and testing, recognizing their importance in maintaining quality and adaptability. This course enhanced my ability to manage software projects, ensuring quality, efficiency, and alignment with business goals.

**Application in Real Projects:**

The knowledge gained from these chapters has direct applications in our Virtual Wedding Planning Concierge project. The techniques for managing requirements will enable us to create more precise and adaptable specifications. We can implement a robust change management process that allows us to incorporate evolving client needs while maintaining project coherence. The design and construction insights will enable us to develop a more modular and flexible system architecture. By adopting coding standards and implementing regular code reviews, we can enhance the overall quality and maintainability of our project.

**Peer Interactions:**

Through discussions with my teammates, we explored real-world scenarios where miscommunication or incomplete requirements led to project complications. These conversations highlighted the importance of clear, continuous communication with stakeholders and the need for iterative requirement validation. And we particularly debated the balance between being responsive to change and maintaining project stability.

**Challenges Faced:**

One of the main challenges I faced was understanding how to effectively implement a comprehensive requirement change management process that would allow us to balance the need for adaptability with maintaining project scope and timeline. It was essential to understand how to document, evaluate and integrate requirement changes without derailing the project's progress. This required careful consideration and strategic thinking. Another challenge was understanding the complex interdependencies between different software lifecycle phases. It was necessary to understand how decisions made in the requirement gathering and design phases affect subsequent construction and testing stages.

**Personal development activities:**

To enhance my understanding of project closure and software lifecycle management, I have started using the Trello and Microsoft Azure DevOps for requirement tracking and management. This hands-on experience has helped me better understand the practical aspects of implementing theoretical concepts learned in class. I have also begun studying successful case studies of project closures to learn best practices that we can adapt for our project.

**Goal for the Next Week:**

My first goal is to review Chapter 1 to 14 of "Software Engineering: A Practitioner's Approach" by Roger S. Pressman to better understand quality management techniques and write a summary of what I learned throughout this course.