**Learning Journal Template**

**Student Name:** Ting Lok Hui

**Course:** SOEN 6841 Software Project Management

**Journal URL:**

**Dates Rage of activities:** 27thOct – 9ndNov

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

**Key Concepts Learned**

**Chapter 8:**

I learned about project closure and its importance in software project management. Project closure involves several critical activities that ensure proper completion and documentation of the project. Through this chapter, I discovered the significance of managing source code versions effectively and how to properly archive project deliverables. The process of filtering and organizing project metrics data for future reference was particularly enlightening. The concept of documenting lessons learned stood out as a crucial practice that can significantly improve future project outcomes. I now understand that project closure is not just about delivering the final product but also about preserving knowledge and experiences gained throughout the project lifecycle.

**Chapter 9:**

I learned about software engineering principles and various software development lifecycle models. The chapter introduced two main approaches: the waterfall model and iterative models like SCRUM and eXtreme Programming. The waterfall model follows a sequential approach where each phase must be completed before moving to the next, while iterative models allow for more flexibility and continuous refinement. I found it particularly interesting how different types of projects benefit from different models. For instance, large-scale systems like ERP applications are better suited for the waterfall model, while modern technology projects like mobile applications benefit more from iterative approaches. The chapter also emphasized the importance of quality gates and how they contribute to better software products.

**Application in Real Projects:**

The knowledge gained from these chapters has direct applications in our Virtual Wedding Planning Concierge project. We are now implementing proper documentation practices and preparing for eventual project closure by maintaining detailed records of our development decisions. For the software lifecycle management, we are discussing whether to use an iterative approach since our project requires frequent updates based on user feedback and changing wedding planning trends. We are also discussing how to define quality gates at different stages of development to ensure our product meets the required standards. These practices are helping us maintain better control over our project and ensure higher quality deliverables.

**Peer Interactions:**

Through discussions with my teammates, we explored different approaches to implementing project closure activities and selecting appropriate development methodologies. We shared experiences about documentation practices and debated the benefits of different lifecycle models. One teammate shared their experience with implementing quality gates in an agile project, which helped me better understand how to balance quality control with rapid development. These interactions enhanced my understanding of how theoretical concepts can be effectively applied in real project scenarios.

**Challenges Faced:**

One of the main challenges I faced was understanding how to effectively implement quality gates in an iterative development model. The concept of maintaining rigid quality checkpoints while staying agile proved theoretically contradictory at first, especially when trying to determine the right metrics to measure at each gate. Another significant difficulty was grasping the proper scope of project closure documentation - determining what constitutes essential historical data versus unnecessary details became a complex decision-making process. The integration of concurrent engineering principles in the software lifecycle also presented a conceptual challenge, particularly in understanding how parallel development activities can be managed without creating dependencies that might lead to project delays. These complexities highlighted the need for a deeper understanding of how theoretical models adapt to real-world scenarios. The challenge of quantifying lessons learned in a way that would be meaningful for future projects also proved difficult, as it required not just collecting data but understanding how to structure it for effective future reference.

**Personal development activities:**

To enhance my understanding of project closure and software lifecycle management, I have started experimenting with different documentation tools and project management practices. I am currently working on creating templates for project closure documents and establishing guidelines for quality gates in our Virtual Wedding Planning Concierge project. 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 create a comprehensive project closure plan for our Virtual Wedding Planning Concierge project, including templates for lessons learned documentation and criteria for archiving project artifacts. Second, I plan to implement our first set of quality gates focusing on code quality and user experience metrics. I will also preview Chapter 10 and 11 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. Additionally, I aim to organize a team discussion about effective documentation practices and how we can integrate them into our daily development workflow without impacting productivity.