**Learning Journal 4**

**Student Name:** Shehzar Aurangzeb Abbasi

**Course:** Software Project Management (SOEN-6841)

**Journal URL:** [Learning Journal 4](https://github.com/Shehzer-Aurangzeb/SOEN-6481-SPM/blob/main/40291795_Learning_Journal_3.pdf)

**Dates Rage of activities:** 27/02/2025 to 10/03/2025

**Date of the journal:** 16/03/2025

1. **Key Concepts Learned:**

In Chapter 8 on Project Closure, I covered the essential steps needed to complete a project. This includes finishing deliverables, monitoring project documentation, and conducting thorough assessments to document lessons learned and establish the framework for next improvements. It is particularly useful to update code repositories and analyze critical performance data to determine what works and where changes are needed. This approach highlights the need of continuous evaluation in enhancing project outcomes throughout time.

The basic phases of software engineering were covered in Chapter 9, along with several lifecycle models (like waterfall) and iterative approaches (like SCRUM and Extreme Programming). The waterfall approach is most effective when the project must remain consistent since it follows a linear pattern. Conversely, iterative models offer the flexibility needed for projects that require regular adjustments, allowing for a more responsive and adaptable process. In iterative models, quality gates—which offer structure without needing a lot of rework—were also highlighted as checkpoints to ensure that every phase meets predefined requirements.

Requirement management was discussed in Chapter 10, with a focus on systematically gathering, verifying, and documenting customer needs. When requirements are properly maintained, projects may adapt to changing needs while maintaining traceability. Requirements are divided into functional and non-functional categories to guarantee clarity and alignment with project objectives. Configuration management is essential for ensuring consistency when requirements change and bridging the gap between customer expectations and the finished product.

1. **Application in Real Projects:**

Applying project closure principles to real-world projects may yield insightful information that directs subsequent endeavors, assisting teams in identifying effective tactics and preventing the recurrence of inefficiencies. These meetings guarantee that successful procedures are recorded, building a knowledge base to improve future initiatives. Additionally, by enabling teams to access previous data for optimization and troubleshooting, the use of source code version control and the preservation of metric data enable future maintenance.

The particular requirements of the project determine which Software Lifecycle Management lifecycle model is best. For example, SCRUM would facilitate rapid adaption and iterative development in dynamic contexts with changing customer needs. The waterfall model, on the other hand, works well for projects requiring a high degree of stability since it follows a defined process that guarantees each phase is finished before moving on to the next. Particularly in projects with steady, well-defined needs, this paradigm reduces risks and rework.

1. **Peer Interactions:**

During our team discussion, we spoke about how we would handle the next phase 2 project work. After dividing the topic among the three members, we shall evaluate each other's work. One peer's experience demonstrated the value of documenting lessons gained, including those that didn't work out, since this helps prevent similar issues in future initiatives. Another colleague believes that in dynamic scenarios with shifting needs, a robust configuration management system is essential.

1. **Challenges Faced:**

One of the largest challenges was identifying the variations across several lifetime models and determining the ideal conditions for each. Particularly in mixed-project situations, choosing between an iterative and waterfall technique proved difficult. Because frequent changes might impact team concentration and resource allocation, it was also challenging to handle requirement changes without interfering with ongoing work.

1. **Personal development activities:**

I came across papers on the application of lifecycle models in various sectors, which provided me with helpful guidance on how to select the most appropriate models for a particular project. This research gave me useful advice on how to deal with requirement changes, particularly in fast-paced development settings, which enhanced my ability to adapt and support project success in dynamic conditions.

1. **Goals for the Next Week:**

Complete Phase 2 project work and learn about topics that are going to be covered in the upcoming week. Learn about git and github and version control systems.