Student ID: 40269370

Student Name: Harsh Tank
Course: Software Project Management (SOEN 6841)
Journal URL: https://github.com/harsh-tank/SOEN-6481-SPM
Dates Rage of activities: 21/10/2024 to 08/11/2024
Date of the journal: 09/10/2024

Key Concepts Learned:

I discussed the crucial tasks required to wrap up a project in Chapter 8 on Project Closure. This include completing deliverables, keeping track of project records, and carrying out in-depth evaluations to record lessons gained and lay the groundwork for future enhancements. To learn what works and where changes are needed, it is especially helpful to update code repositories and analyze critical performance data. This method emphasizes how crucial ongoing assessment is to improving project results over time.

Chapter 9, discussed the fundamental stages of software engineering and focused on different lifecycle models, such as waterfall, and iterative techniques, such as SCRUM and Extreme Programming. Because it follows a linear pattern, the waterfall model works best when project needs stay constant. Iterative models, on the other hand, provide the adaptability required for projects involving frequent modifications, enabling a more responsive and flexible procedure. Quality gates, which provide structure without requiring a great deal of rework in iterative models, were also emphasized as checkpoints to guarantee that each step satisfies predetermined requirements.

In Chapter 10,The topic of requirement management was covered, with an emphasis on methodically obtaining, confirming, and recording client requirements. Projects can adjust to shifting requirements while preserving traceability when requirements are managed effectively. To ensure clarity and conformity with project goals, requirements are separated into functional and non-functional categories. In order to bridge the gap between client expectations and the final product, configuration management is crucial in maintaining consistency as requirements change.

Application in Real Projects:

In real-world projects, applying lessons learned from project closure can provide valuable insights that guide future efforts, helping teams capture successful strategies and avoid repeating inefficiencies. These sessions ensure that effective practices are documented, creating a repository of knowledge to enhance subsequent projects. Additionally, using source code version control and preserving metric data support

future maintenance by allowing teams to access historical data for optimization and troubleshooting.

Selecting the appropriate lifecycle model in Software Lifecycle Management depends on the project's specific needs. For instance, SCRUM would be advantageous in dynamic environments with evolving client requirements, enabling iterative development and quick adaptation. In contrast, the waterfall model is well-suited for projects with high-stability demands, where a structured approach ensures each phase is fully completed before progressing to the next. This model minimizes risks and rework, especially in projects with stable, clearly defined requirements.

Peer Interactions:

Our team discussed on meeting about how we are going to approach our upcoming phase 2 project work. We split the topic between all 5 members and then will review each other's work. The experience of one peer made it clear how important it is to record lessons learned, particularly those that didn't work out, as this helps avoid similar problems in subsequent projects. A strong configuration management system is crucial in dynamic situations with changing requirements, according to another colleague.

Challenges Faced:

Recognizing the differences between different lifetime models and figuring out the best circumstances for each was one of the biggest obstacles. It was challenging to decide between an iterative and waterfall methodology, especially in mixed-project circumstances. It was also difficult to manage requirement changes without interfering with existing development because frequent changes can affect team focus and resource allocation.

Personal Development Activities:

I read articles about the use of lifecycle models in other industries, which gave me useful advice on how to choose the best models for a given project. This study improved my capacity to adjust and support project success in dynamic circumstances by providing insightful guidance on handling requirement changes, especially in fast-paced development environments.

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 system.