Learning Journal
Student Name: Pranavkumar Sakariya

Course: Software Project Management

Journal URL: https://github.com/pranav687/SOEN6841_SPM.git

Week 7 & 8: 6 October 2024 to 2 November 2024

Chapter 8:

Key Concepts Learned:
we focused on **Chapter 8**, which talked about wrapping up a project. We went over key activities, such as finishing up the deliverables, keeping track of source code versions, storing project metrics, and noting lessons learned to improve future projects. The chapter emphasized how important it is to keep data well-organized and to take time to review the project's results to make transitions smoother for the next steps.

Application in real world:
Healthcare Research: In clinical trials, researchers archive patient data, study results, and procedural documents after project completion. Documenting lessons learned, such as adjustments in trial protocols or data collection methods, can make future research safer and more effective.

Peer Interaction:
Collaborated with my teammates to better understand the project closure process. We exchanged ideas on how various organizations manage the final stages of a project and discussed the best ways to handle data and document everything for future reference.

Challenges faced:
Figuring out the specific metrics needed for data archiving was challenging. It took careful review to make sure all the important information was properly recorded, which is crucial for project audits and quality assessments.

Personal Development Activities:
Reviewed extra materials on effective project closure practices, including case studies from different industries, to understand how various organizations document their results and apply lessons learned to improve future projects.

<u>Chapter 9:</u>

Key Concepts Learned:
we studied Chapter 9, which covered Software Lifecycle Management. We learned about the basics of software engineering, different lifecycle models like Waterfall and SCRUM, the phases of software development, quality assurance processes, and how both iterative and sequential models adjust to evolving requirements.

Application in real world:
Understanding lifecycle models helps in making better decisions about how to approach projects. The Waterfall model works well for projects with fixed requirements, while iterative models like SCRUM are better suited for projects with changing needs. Using the right model can enhance a project's flexibility and improve the quality of its results.

Peer Interaction:
I had a discussion with my classmates about the advantages and disadvantages of different software lifecycle models. This exchange helped me gain valuable insights into how these models can be adapted to various industries and project types, deepening my understanding of how to choose the most suitable lifecycle for a project.
.

Challenges faced:
Learning the differences between lifecycle models and knowing when to use each one was challenging. It took extra effort to understand the subtleties between iterative and sequential models and how they affect the success of a project.

Personal Development Activities:
I explored advanced readings on lifecycle management and quality assurance, focusing on how different lifecycle models can improve productivity and help manage risks effectively.

Goals for next week:
Start preparing for Chapter 10 on Customer Requirements Gathering and Management, focusing on the importance of configuration management and quality assurance during the requirements phase.