Final Learning Journal

Student Name: Nisarg Hitesh Adalja

Student ID: 40276285

Course: SOEN 6841(Software Project Management)

Journal URL: nisarg291/SOEN6841_Journal (github.com)

Week 6: Chapter-7 Project Monitoring and Control System

In Week 6, In chapter -7, I learned how the managing software projects can be challenging due to unclear specifications, leading to assumptions and vagueness in project work. Despite this, project teams utilize various tools and techniques for monitoring and control. These include buffers in project plans to adjust for risks, as well as techniques like resource leveling and schedule optimization to overcome setbacks. Additionally, Earned Value Management (EVM) offers a comprehensive approach with performance indicators on a project dashboard, enabling prompt corrective actions when necessary.

Week 7: Chapter-8 Project Closure

During Week 7, In project closure chapter summarise that when project comes to an end, there are still a few important things to do, especially if the project didn't go smoothly. These tasks include letting go of resources, writing down what we learned, keeping track of the code we wrote, and organizing all the project information. It's important to store all this stuff correctly so we can use it later. Keeping track of the code is especially important because we make a lot of changes while testing. We need to decide which version of the software to give to the customer. When we store everything, we need to make sure we only keep what's important and put it in the right place for later.

Week 8: Chapter-9,10 Introduction to Software Life-Cycle and Requirement Management

In software development life cycle introduction, I covered its components, including rapid development techniques like concurrent engineering. I also learned the importance of software metrics for quality control and explored the various work products produced across different phases like requirements, design, construction, testing, release, and maintenance.

In chapter 10, we covered software requirement gathering techniques, management, change control, and versioning. Change and unclear requirements are common challenges in software projects, addressed by either focusing on a few requirements at a time or managing the entire development process for flexibility. To ensure defect-free requirement specifications, a thorough review process is essential, checking for clarity, testability, and maintainability.

Week 9: Chapter 11,12,13,14: Construction, Testing, Release, and Maintenance Life Cycle

In Week 10, testing, release, and maintenance. These phases involve coding the software, conducting different types of testing (such as unit, integration, system, and acceptance testing), deploying the software into production, and providing continuous maintenance and support. We emphasized the importance of adopting best practices like code development, test-driven development, continuous integration, and deployment automation to maintain high software quality and reliability across its lifecycle.

Final Reflections:

Overall Course Impact:

This course helped me understand a lot about managing software projects. It showed me how to handle projects from start to finish, covering important stuff like keeping track of progress, figuring out what's needed, planning out how the software will work, building it, testing it, releasing it, and keeping it working well. Learning all this has given me the skills to move through the different stages of making software and make sure it turns out great.

Application in Professional Life:

What I learned in this course is important for my job as a software project manager.
● I can use these skills in lots of different situations, like leading teams that build software, setting up ways to manage projects, and making things work better in companies.
● Being able to keep an eye on how projects are going, understand what's needed, come up with good plans, and guide the whole process will help me do well in managing hard software projects in the industry.

Peer Collaboration Insights:

Peer collaboration has been instrumental in enhancing my learning experience throughout the course. Engaging with classmates allowed me to gain diverse perspectives, exchange ideas, and learn from shared experiences. The collective wisdom of the group has not only

deepened my understanding of course concepts but also broadened my outlook on software project management practices, fostering a supportive and collaborative learning environment.

Personal Growth:

During the course, I've really improved my thinking skills, problem-solving abilities, and how I talk to people.

- Now, I'm better at looking at different ways of doing project management, finding the best solutions for tricky problems, and talking to both the team and other important people about it.
- Plus, I feel more confident about handling tough challenges in real-life software project management.

Overall, this course has been a big deal for me. It gave me the skills and confidence I need to manage software projects well. I'm really glad I took it, and I can't wait to use what I've learned at work.