Learning Journal 2

Student Name: Santhosh Dayakar

Course: SOEN 6481 Software Project Management

Journal URL:https://github.com/S-15-77/Software-Project-Management-Journal

Dates Range of activities: 22-09-2025 to 20-09-2025

Date of the journal: 26-09-2025

Key Concept Learned:

Over the past two weeks, I studied two core topics in software project management Configuration Management (CM) and Project Planning. CM focuses on controlling and documenting software changes to maintain stability, consistency, and accountability throughout a project's lifecycle. It helps prevent version conflicts and rework by defining structured processes for identifying, controlling, tracking, and auditing changes. The concept of a Change Control Board (CCB), which reviews and approves modifications before implementation, emphasized how CM goes beyond technical work to ensure disciplined coordination within teams.

Project Planning complements CM by providing a structured roadmap for executing and managing a project effectively. It involves defining tasks through a Work Breakdown Structure (WBS), estimating time and resources, and scheduling milestones using tools such as Gantt charts and the Critical Path Method (CPM). Connecting these lessons with earlier topics like risk management, I realized that planning minimizes uncertainties while CM ensures changes are systematically controlled. Together, they establish a foundation for delivering reliable, high-quality software projects.

Application to Real Project:

In another course, my team and I are developing a news analytics web application using the Play Framework, which integrates articles from the News API. The concepts of Configuration Management and Project Planning have been directly relevant to this project. We used Git and GitHub for version control, enabling configuration identification and change tracking across multiple modules. To maintain consistency, we introduced a simple change control process, requiring peer reviews before merging updates to the main branch. This mirrored the configuration control and auditing practices learned in class and helped reduce integration errors. One challenge we faced was resolving merge conflicts during concurrent development, which we addressed by establishing clear branching rules and scheduled code reviews an approach that improved coordination and reduced rework.

From a planning perspective, we applied techniques such as Work Breakdown Structure and Gantt chart scheduling to divide project tasks, estimate effort, and manage timelines more effectively. These tools allowed us to visualize dependencies, set milestones, and monitor progress, aligning closely with the activity organization and milestone tracking principles. Implementing daily check-ins also improved communication and minimized delays caused by misunderstandings. Overall, applying CM ensured version stability and accountability, while structured project planning provided clarity and control transforming a collaborative academic project into a well-organized, efficient, and goal-driven development process.

Peer Interaction and Collaboration:

Working with my team this week strengthened my understanding of Configuration Management and Project Planning concepts. Through collaborative discussions and code reviews, I saw how CM principles like configuration control and status accounting apply in practice. Managing versions in Git and reviewing peer changes helped me understand how consistent tracking and documentation maintain project integrity.

Team planning sessions also improved my grasp of project management concepts such as Work Breakdown Structure and milestone scheduling. Sharing feedback on task allocation and timelines made me realize the importance of clear communication and regular progress checks. These interactions transformed theoretical ideas into practical insights, showing how teamwork reinforces structured planning and disciplined change management in real projects.

Challenges Faced:

While studying, I struggled to bridge the gap between the theoretical aspects of Configuration Management and their real-world application. Terms such as configuration identification, status accounting, and auditing initially felt abstract until I compared them to Git's version-tracking mechanisms. Understanding how CCBs operate and how impact analysis reports are maintained took extra effort, as these processes differ across organizations. To overcome this, I explored case studies and experimented with CM tools to see how baselines and approvals are managed. This clarified CM's role in ensuring consistency, accountability, and project integrity.

In Project Planning, I found it difficult to visualize how techniques like top-down and bottom-up planning, or the critical path and critical chain methods, are applied in large-scale projects. Estimating resources and managing interdependent tasks were challenging, especially under potential delays. To improve, I created a sample WBS and Gantt chart to simulate dependencies and timelines. This helped me understand how milestones and deliverables act as progress checkpoints. These challenges pushed me toward a more hands-on learning approach and strengthened my appreciation for structured planning and control mechanisms.

Personal development activities:

I focused on deepening my understanding of Configuration Management and Project Planning through self-learning and practice. I explored how Git and GitHub support configuration identification and status accounting by tracking revisions and maintaining baselines. Tutorials on branching and merging improved my grasp of configuration control and traceability. I also used GanttProject to simulate scheduling and create a small Work Breakdown Structure for my coursework, which strengthened my estimation and planning skills. These activities enhanced my technical competence and confidence in applying structured management principles to real-world projects.

Goals for the Next Week:

Looking ahead, I aim to:

- Apply the principles of Configuration Management by practicing configuration control, version tracking, and change documentation using Git to strengthen my understanding of real-world change management.
- Enhance my project planning and scheduling skills by creating a detailed Work Breakdown Structure and Gantt chart to visualize dependencies.
- Explore how critical path and critical chain techniques influence project timelines and resource management to improve my ability to plan and forecast effectively.