**Learning Journal**

**Student Name:** Saheb Singh Chandok

**Course:** Software Project Management

**Journal URL:** <https://github.com/SahebChandok/SOEN-6841_LearningJournal>

**Dates Rage of activities:** 16 January 2025 – 23 January 2025

**Date of the journal:** 28 January 2025

**Key Concepts Learned:**

From Chapter 1, I learned the fundamental principles of software project management, including the characteristics of a project, and the critical roles of a Project Manager, Leader, Scrum Master, and their responsibilities in managing resources, risks, and quality. In Chapter 2, I learned about project initiation, project charter, scope, and SMART objectives (Specific, Measurable, Achievable, Relevant, Time-bound). Additionally, I understood the importance of accurate cost and effort estimation.

**Application in Real Projects:**

The concepts, such as defining a clear project scope, creating a project charter, and setting SMART objectives, are essential for managing real-world projects effectively.For example, consider the development of an e-learning app. Defining a clear project scope ensures features like video lessons, quizzes, and progress tracking are well-documented. Breaking the project into phases, such as planning, development, and testing, ensures smooth execution.

**Peer Interactions:**

This week, we were assigned the task of selecting our project team and the team for topic analysis. I had a discussion with my friends, Abhijit Banerjee and Arjun Yadav, about potential topics for our project and how we could present them innovatively. Additionally, I had a separate conversation with Abhijit Banerjee regarding our poster presentation. We discussed the potential challenges we might face while preparing for it and explored various strategies to overcome these challenges while presenting our topic in a creative and engaging manner.

**Challenges Faced:**

I initially struggled to understand the differences between the project charter, project scope, and project objectives. However, after further studying and reviewing the PPTs uploaded on Moodle, I can now confidently say that I have a clear understanding of these concepts.

**Personal Development Activities:**

While working on exercises 1.1 and 2.2 from the textbook, I took the opportunity to read about the latest IT projects funded by the government. I learned about their start dates, target completion timelines, and assigned budgets. Additionally, I explored various open-source projects and their project charters. This information was new to me and helped me stay updated on government investments in the IT sector.

**Goals for the Next Week:**

I aim to gain a deeper understanding of the topics covered last week. Additionally, I plan to meet with my teammates to collaborate on the market analysis for our project and work on refining our project pitch.

**Dates Rage of activities:** 29 January 2025 – 9 February 2025

**Date of the journal:** 9 February 2025

**Key Concepts Learned:**

This week, I explored key concepts related to effort and cost estimation in software

project management, along with risk management strategies. I gained insights into

various estimation techniques such as function point analysis, wide-band Delphi, and

the COCOMO model. These techniques help in making accurate predictions regarding

the effort required to develop software projects. Additionally, I learned that experience based and algorithmic cost modeling approaches play a significant role in estimation

accuracy. In risk management, I learned about different types of project risks, including estimation

risks, resource unavailability, technology obsolescence, and schedule risks. Risk

assessment involves identifying, analyzing, and prioritizing risks to mitigate their impact

on project progress. I also understood the importance of risk response strategies like

avoidance, transference, mitigation, and acceptance, which are crucial for maintaining

project stability.

**Application in Real Projects:**

The concepts of effort estimation are highly applicable in real-world software

development projects. By using function point analysis and COCOMO models, project

managers can estimate required effort and costs, ensuring that resources are allocated

effectively. For example, in projects involving iterative development, effort estimation

techniques must adapt to changing requirements, making estimation more dynamic and

flexible.

Risk management is another vital component that applies to real-world projects.

Implementing risk identification and mitigation techniques allows project teams to

anticipate potential issues before they become major obstacles. For instance, by

applying risk prioritization, teams can focus on addressing high-impact risks early in the

project lifecycle, reducing project delays and budget overruns.

**Peer Interaction:**

Discussions with my peers Abhijit Banerjee and Arjun Yadav provided diverse

perspectives on the challenges of project estimation and risk management.

Collaborative exercises in effort estimation, such as using the Delphi method, helped

refine the accuracy of individual estimates by incorporating multiple viewpoints. Peer

discussions on risk assessment highlighted real-world examples of risk factors in

projects, reinforcing the importance of proactive risk management. Moreover I also had

a discussion with my team regarding the project progress report. We also had a

discussion with our assigned TA Piyush, to get a feedback on our previous submissions

and understood how we can ace the project pitch next week.

**Challenges Faced:**

One of the main challenges I faced was understanding the complexities of algorithmic

cost models, such as COCOMO. These models involve multiple factors, making it

difficult to determine accurate effort multipliers and scale factors. Another challenge was

applying risk assessment techniques in scenarios with limited data, where estimating

the probability and impact of risks requires significant judgment and experience.

Additionally, ensuring that effort estimation techniques remain accurate throughout a

project's lifecycle is difficult, as project scope and requirements often change. This

challenge emphasizes the need for continuous updates and refinements in estimation

techniques.

**Dates Rage of activities:** 10 February 2025 – 23 February 2025

**Date of the journal:** 23 February 2025

**Key Concepts Learned:**

This week, I explored two critical concepts in software project management: Configuration Management (CM) and Project Planning.

* **Configuration Management:** CM is essential for controlling and documenting changes in a system. I learned about its key functions: configuration identification, control, status accounting, and auditing. The importance of CM lies in its ability to prevent uncontrolled changes, which can lead to project delays and quality issues.
* **Project Planning:** Effective project planning involves work breakdown structures (WBS), scheduling techniques (top-down and bottom-up), and resource allocation. I understood the significance of setting milestones, deliverables, and utilizing scheduling methods like Gantt charts and critical path analysis.

**Application In Real Time Project:**

Understanding CM is crucial in agile and DevOps environments where frequent changes occur. In previous projects, I have faced challenges in maintaining code versions, which led to defects reappearing. Implementing proper version control and change request tracking would have prevented such issues. For project planning, I can apply WBS to break down complex tasks and assign appropriate resources. Utilizing tools like JIRA and MS Project would enhance efficiency by visually tracking progress and dependencies.

I believe configuration management is a fundamental aspect of any project, regardless of its size. During my academic term at Concordia University, I encountered several instances where effective configuration management enabled timely project delivery. For example, in the Fall 2024 term, I worked on a course project with a team of four members. Since multiple members were working simultaneously, we frequently faced merge conflicts and versioning issues. However, by leveraging Git and GitHub, we efficiently managed these conflicts and ensured the successful and timely submission of our project.

**Peer Interaction:**

Discussions with peers helped me understand different perspectives on CM implementation. One peer shared their experience of dealing with versioning issues in a software development project, which reinforced the need for automated tracking systems. Another discussion focused on the importance of contingency planning in project scheduling, highlighting real-world difficulties in estimating task durations accurately. This week I also had my mid term exam. Prior to the exam I had a discussion with my peers regarding the important topics that could be asked in the exam. After my exam, I also discussed with my peers, the questions which I was not able to answer. This discussion helped me understand my mistakes. I also learned some ways in which I could ace my exams in the future.

**Challenges Faced:**

One of the main challenges was understanding the integration of CM tools (such as Git and SVN) into larger project management frameworks. Additionally, comprehending the impact of scheduling dependencies on resource allocation was complex, especially when dealing with multiple teams and evolving project scopes. I also faced difficulty in understanding project planning.

I read the PPT which was uploaded on Moodle. Initially, I couldn't get the hang of it because a lot of information was given in the slides. Moreover, I specifically faced difficulty in understanding the Critical Path Method and Goldratt’s Critical Chain Method. Here, I was not able to understand the concept of buffer. To resolve my doubts, I read the textbook suggested to us. Additionally, I watched various videos on YouTube to clarify my concepts. But apart from these challenges, I can confidently say that I now understand these topics well.

**Personal Development Activities:**

To strengthen my knowledge, I explored online tutorials on CM tools like Git and Jenkins. I also practiced creating WBS using MS Project and experimented with scheduling methods in a sample project. Additionally, I reviewed case studies on software project failures due to poor configuration management to learn from past mistakes.

**Goals For the Next Week:**

* Deep dive into automated CM tools and their integration in CI/CD pipelines.
* Work on a small project implementing WBS and scheduling techniques.
* Engage in a discussion with industry professionals to gain insights into real-world project planning challenges.
* Improve my understanding of risk management and contingency planning in project execution.
* We have to prepare a pitch for our project, and present it in the classroom.
* Schedule a meeting with project teammates, to plan further actions, and get an update on previously assigned tasks.

By focusing on these goals, I aim to develop a strong foundation in project planning and configuration management, ensuring effective project execution in future professional settings.

**Dates Rage of activities:** 3 March 2025 – 16 March 2025

**Date of the journal:** 16 March 2025

**Key Concepts Learned:**

Over the past two weeks, I gained a thorough understanding of two crucial aspects of Software Project Management: Project Monitoring & Control and Project Closure. I learned that project monitoring involves collecting accurate data to measure progress against the project baseline. It emphasizes the importance of timely tracking of schedule, budget, and quality metrics. Project control ensures that corrective actions are taken whenever deviations occur, maintaining alignment with planned objectives. The chapter also introduced me to Earned Value Management (EVM), which integrates cost and time constraints to evaluate project progress. Techniques such as S-curves, slack management, buffer management, and resource levelling were discussed in detail. Additionally, performance indicators like defect density and resource utilization metrics were highlighted for effective monitoring. Moreover I understood the importance of finalizing all project deliverables, version control, and archiving project metrics. The chapter emphasized conducting lessons learned sessions and using archived data to enhance risk management and future project planning.

**Application in Real Projects:**

The concepts of EVM, monitoring, and closure activities are proved to be highly applicable in real-world software development projects during my time at my previous company. Under the guidance of my manager, our team actively implemented EVM practices by maintaining baseline costs and timelines, ensuring that budget reviews were conducted regularly, and recalibrating priorities whenever schedule variances occurred. I observed how, in one instance, when a critical module faced delays, earned value data was utilized to forecast project completion and resources were strategically reallocated to meet deadlines. Additionally, my manager introduced a structured 'Lessons Learned' template, which helped the team systematically document challenges and identify areas for process improvement. Inspired by these practices, I took the initiative to develop automated scripts for collecting project metrics and generating EVM reports, which significantly reduced manual effort and improved real-time visibility into project progress.

**Peer Interactions:**

Collaboration with my peers played a crucial role in enhancing my learning. During a study group session, we had an in-depth discussion about slack management and buffer techniques. One peer shared their experience where excessive slack led to resource underutilization, prompting me to focus on resource levelling. Additionally, I presented my approach to integrating EVM into an Agile workflow and received valuable feedback. My peers during the group session were Abhijit Banerjee and Arjun Yadav.

**Challenges Faced:**

One of the main challenges I faced was understanding the calculations involved in Earned Value Management, particularly the relationships between planned value, earned value, and actual cost. To overcome this, I revisited the course material and practiced using real project data. Another challenge was ensuring consistent and accurate data collection from distributed teams.

**Personal Development Activities:**

To foster my professional growth, I completed a LinkedIn Learning course on advanced project monitoring techniques, which provided hands-on experience with tools like MS Project and Jira. I also explored articles on Agile-EVM integration, expanding my knowledge on how to balance traditional control methods within Agile frameworks. Additionally, I conducted a retrospective review of my previous projects, applying project closure activities to retrospectively identify improvement areas.

**Goals for the Next Week:**

For the upcoming week, my immediate goals are to apply slack removal and resource leveling techniques in an ongoing team project to ensure optimal workload distribution. I also plan to conduct a 'mini-lessons learned' meeting at the end of our current sprint to reinforce closure practices. For long-term growth, I aim to deepen my expertise in project performance forecasting by learning advanced predictive analytics tools. Additionally, I will focus on enhancing my leadership skills by initiating peer review sessions focusing on project metrics and performance indicators. These goals align with my career aspiration of becoming a Project Manager, equipping me with both technical and strategic skills.

**Final Journal**

**Dates Rage of activities:** 13 March 2025 – 30 March 2025

**Date of the journal**: 29 March 2025

**Overall Course Impact:**

This course has profoundly transformed my understanding of Software Project Management (SPM) by bridging theoretical concepts with practical applications. Initially, my perception of project management was limited to scheduling and task delegation. However, through this course, I gained a holistic view of SPM, encompassing initiation, planning, execution, monitoring, and closure, along with risk and configuration management.

Key concepts that I learned include:

* The importance of a well-defined project charter, scope, and SMART objectives in setting a clear direction.
* The role of effort and cost estimation techniques (e.g., COCOMO, Function Point Analysis) in realistic project planning.
* The criticality of risk management—identifying, analyzing, and mitigating risks to avoid project derailment.
* The value of configuration management in maintaining software integrity across versions.
* The differences between waterfall and iterative models, and how Agile methodologies (e.g., Scrum) enhance adaptability.

A challenging yet transformative realization was how traditional project management principles (e.g., Waterfall) contrast with Agile approaches. Initially, I believed rigid planning was essential, but learning about iterative models demonstrated how flexibility and continuous feedback lead to better outcomes in dynamic environments.

**Application in Professional Life:**

The knowledge from this course is directly applicable to my professional career, particularly in software development and team leadership roles.

1. Project Planning & Estimation: In future projects, I will use WBS (Work Breakdown Structure) and Earned Value Management (EVM) to track progress and budget adherence. Techniques like Delphi estimation will help in collaborative effort forecasting.
2. Risk Management: I now recognize the need for proactive risk assessment (e.g., technical, scheduling, and resource risks) and mitigation strategies (e.g., contingency buffers).
3. Agile & Iterative Development: For fast-paced projects, I will advocate for Scrum or Kanban to improve adaptability and stakeholder feedback integration.
4. Configuration & Quality Assurance: Implementing version control (Git/Jira) and quality gates will ensure software reliability.

Long-term professional opportunities include:

* Transitioning into project management roles (e.g., Scrum Master, Product Owner).
* Leading cross-functional teams with structured methodologies.
* Enhancing client communication through clear requirement documentation and change management.

**Peer Collaboration Insights:**

Collaboration with peers was instrumental in deepening my understanding. Key takeaways:

* Group discussions on case studies (e.g., risk scenarios) provided diverse perspectives on problem-solving.
* Team projects (e.g., developing a software proposal) improved my ability to delegate tasks, resolve conflicts, and integrate feedback.
* Peer reviews of learning journals helped refine my critical thinking and self-assessment skills.

A notable experience was debating the merits of Waterfall vs. Agile—this exchange highlighted how different industries (e.g., healthcare vs. startups) require tailored approaches.

**Personal Growth:**

This course fostered significant **personal and academic growth**:

1. **Critical Thinking**: Analyzing real-world project failures (e.g., due to poor risk management) sharpened my problem-solving skills.
2. **Communication & Leadership**: Presenting topic analyses improved my ability to articulate complex ideas clearly.
3. **Adaptability**: Learning multiple estimation techniques (e.g., FPA vs. COCOMO) enhanced my flexibility in choosing the right tool for a given scenario.
4. **Self-Reflection**: Maintaining a **learning journal** helped me track progress and identify areas for improvement (e.g., time management in project scheduling).

Areas for further development:

* **Advanced Agile practices** (e.g., SAFe, Lean).
* **Negotiation skills** for stakeholder management.