

Learning Journal 3

Student Name: Zaziba Ahmed

Course: Software Project Management

Journal URL: <https://github.com/ZazibaAhmed/SOEN6841-Learning-Journals>

Date Range of Activities: 6th Feb 2025 - 20th Feb 2025

Date of the Journal: 2/23/2025

Key Concepts Learned:

Over the past few weeks, we covered Chapters 1 through 6 in class, focusing on key aspects of software project management. While no new topics were introduced due to the midterm, I spent time revisiting all the chapters, especially in preparation for midterms. This revision helped reinforce key concepts in project planning, project initiation, effort estimation, risk management, configuration management and project planning.

- **Chapter 1: Understanding Software Projects**

- We discussed what makes a project unique, emphasizing that it has a defined start and end time and is aimed at achieving specific goals.
- Software project management involves planning, organizing, and managing resources within constraints like budget and time.
- Some key project roles include: **Leader, Manager, Scrum Master, Project Manager**
- The typical phases of a project include initiation, planning, execution, monitoring & control, and closure.

- **Chapter 2: Project Initiation**

- Every project starts with a **Project Charter**, which outlines its purpose, scope, objectives, and budget.
- We learned about **SMART objectives**, which ensure goals are specific, measurable, achievable, relevant, and time-constrained.
- Understanding **project scope** is critical to setting boundaries and expectations.
- Initial scheduling involves breaking down tasks, identifying dependencies, and estimating durations before finalizing a baseline schedule.

- **Chapter 3: Effort Estimation Techniques**

- Estimating effort is one of the biggest challenges in software project management.
- There are 2 types of estimation techniques:
 - Experience-based techniques
 - Algorithmic cost modeling
- We explored different estimation technique in details:
 - **Function Point Analysis (FPA)**
 - **Estimation by Analogy**
 - **Algorithmic Cost Models (COCOMO II)**

- Challenges include uncertainty due to evolving technologies, project uniqueness, and team skill variations.
- We also compared estimation in **waterfall vs. iterative models**—the latter being more adaptable but requiring continuous reassessment.
- **Chapter 4: Risk Management**
 - Risk management involves identifying, analyzing, and mitigating risks that could impact a project.
 - We categorized risks into estimation risks, resource risks, and technological risks.
 - The steps in risk assessment are:
 - **Risk Identification:** Listing potential risks affecting the project.
 - **Risk Analysis:** Evaluating likelihood and impact (qualitative and quantitative approaches).
 - **Risk Prioritization:** Addressing high-impact risks first.
 - We discussed risk response strategies such as:
 - **Acceptance:** Preparing contingency measures.
 - **Avoidance:** Eliminating risks by modifying the project plan.
 - **Transference:** Passing risk responsibility to third parties.
 - **Mitigation:** Reducing the probability or impact of risks.
 - Iterative models help mitigate risks better than waterfall models by allowing frequent feedback and adjustments.
- **Chapter 5: Configuration Management**
 - Configuration Management (CM) is crucial for tracking changes and maintaining project stability.
 - We explored challenges caused by poor CM, such as version control issues, lost documentation, and miscommunication.
 - CM ensures:
 - Product integrity and compliance with standards.
 - Clear traceability between requirements, documentation, and code.
 - The four main CM functions are:
 - **Configuration Identification:** Defining project components.
 - **Configuration Control:** Managing and approving changes.
 - **Configuration Status Accounting:** Keeping track of changes and their impact.
 - **Configuration Auditing:** Ensuring compliance with requirements.
- **Chapter 6: Project Planning**
 - Planning is an ongoing activity throughout the project lifecycle.
 - Key project planning components include:
 - **Work Breakdown Structure (WBS):** Breaking tasks into manageable units.
 - **Project Scheduling:** Using top-down or bottom-up planning approaches.
 - **Resource Allocation:** Assigning resources efficiently.
 - **Risk Contingency Planning:** Preparing for unexpected issues.

- **Supplier and Communication Planning:** Ensuring clear collaboration with external partners.
- We also discussed scheduling techniques:
 - **Critical Path Method (CPM):** Identifying the longest sequence of dependent tasks.
 - **Gantt Charts:** Visualizing project timelines.
 - **Goldratt's Critical Chain Method:** Removing unnecessary buffers to optimize scheduling.
 - **Activity Networks:** Mapping task dependencies.
- We compared **iterative vs. waterfall project planning**, highlighting the flexibility of iterative models in handling change.

Application in Real Projects:

- **Project Planning:** Applying SMART objectives and project charters ensures clear goal-setting and stakeholder alignment.
- **Effort Estimation:** Techniques like FPA and analogy-based estimation improve budget planning and resource allocation.
- **Risk Management:** Understanding risk categories and mitigation strategies helps proactively manage uncertainties.
- **Configuration Management:** Proper version control and documentation tracking prevent major project setbacks.

Peer Interactions:

- We had in-depth discussions on risk assessment and real-world estimation challenges.
- Collaborative exercises on project planning improved our understanding of scheduling techniques.
- Simulations of CM issues helped reinforce best practices in managing software configurations.

Challenges Faced:

- **Understanding Estimation Techniques:** Understanding FPA Analysis took more time. COCOMO II was complex
- **Configuration Control Policies:** Managing multiple versions and tracking changes across teams required a structured approach

Personal Development Activities:

- Reviewed case studies on risk management in software projects.
- Practiced using Git and JIRA for version control and issue tracking.
- Explored scheduling tools like Microsoft Project and Trello to gain hands-on experience.
- Focused on revising all concepts in preparation for midterms.

Goals for the Next Week:

1. Continue reinforcing key concepts.
2. Improve my understanding of project scheduling techniques by creating a detailed project timeline.
3. Engage in discussions with peers to clarify any remaining uncertainties before exams.