

## Learning Journal

**Student Name:** Rachit Rajesh Pednekar

**Course:** Software Project Management SOEN 6841 Winter 2024

**Journal URL:** <https://github.com/racCC/SOEN-6841-WINTER-2024/tree/learning-journal>

**Week5:** 18/02/2024-9/03/2024

**Key Concepts Learned:** The key concepts learned during this week revolved around Chapter 7 although it was not explained in the previous class had a glance at it for better understanding. The key concepts learned are as follows:

### 1. Project Monitoring:

- **1.1 Strategic Baseline Monitoring:**

- Project monitoring involves meticulous tracking of progress, schedules, and budgets.
- Use various tools such as status reports, Goldratt's critical chain method, Gantt charts, and earned value management for effective monitoring.
- Treat the project plan as a strategic baseline for progress reports, forming the cornerstone of monitoring against milestones.
- Proactive communication and strategic planning are essential if milestones are missed, utilizing techniques like resource leveling and optimization to realign the project trajectory.

- **1.2 Nuanced Task Progress Measurement:**

- Accurate measurement of task progress requires a nuanced understanding of planned and actual start dates, work volume, and task duration.
- Relying solely on dates can lead to misleading progress calculations, emphasizing the importance of accounting for work volume, such as code size, for meaningful assessments.

- **1.3 Deviation Identification:**

- Focus on identifying deviations from the planned schedule and costs during project monitoring.
- Calculations involving schedule performance and cost deviations offer insights into the project's health, acknowledging the simplifications addressed later through Earned Value Management (EVM).

- **1.4 Effective Performance Indicators:**

- Performance indicators crucial for assessing cost, schedule, and quality are generated through tools like Earned Value Management (EVM).
- The accuracy of these indicators relies heavily on the availability and reliability of baseline data for comparison with actual execution data.

- **1.5 Granular Project Schedule Monitoring:**

- Monitoring against the project schedule involves detailed tracking of task-level aspects like resource allocation, progress, and performance.
- Project schedules offer more granularity than project plans, necessitating daily tracking and monitoring, including assessing resource utilization percent, loading, and task progress.

### **1.6 Dynamic Periodic Measurement:**

- Given the dynamic nature of projects, frequent tracking and measurement of task-level progress become pivotal for early issue identification.
- Daily activity logging in a centralized system aids in tracking task progress, while cost monitoring involves effective sheets detailing resource hours worked against budgeted costs for each task.

### **1.7 Utilization of Earned Value Management (EVM):**

- Earned Value Management (EVM) is employed for tracking and measuring project progress, involving values like Planned Value (PV), Earned Value (EV), Actual Value (AV), Schedule Variance (SV), Cost Variance (CV), Cost Performance Indicator (CPI), and Schedule Performance Indicator (SPI).

### **1.8 Holistic Resource Utilization Measurement:**

- Resource utilization is measured at the program or business level, gauging the efficiency of engaged staff.
- Calculate the percentage of staff assigned to projects against the total available for a comprehensive understanding.

### **1.9 Effective Resource Loading Measurement:**

- Resource loading, tracking allocated and actual work hours for each resource, is crucial for gauging project efficiency.
- Adjustments may be needed concerning task completion times, requiring resource supplementation or workload increases.

### **1.10 Skills and Knowledge Monitoring:**

- Aligning resources with tasks during project planning necessitates addressing skill gaps through training plans.
- Execution involves tracking planned training completion to ensure competency, with adjustments made for unexpected skill gaps or resource changes.

### **1.11 Continuous Risk Monitoring:**

- Every project task carries inherent risks, demanding continuous risk identification and assessment.
- Factor in impact and probability throughout project execution to develop and implement effective contingency plans.

### **1.12 Timely Issue Resolution:**

- Issues are inevitable during project execution, requiring prompt resolution to avoid hindering progress.
- Prioritize and address issues based on their impact, ensuring a balanced and efficient resolution process like mitigating risks.

### **1.13 Crucial Role of Status Reports:**

- Status reports serve as essential tools for customer updates on project progress.
- After completing milestones, these reports include information on cost, schedule, and quality, addressing delays, and presenting remedy plans.
- Building a strong customer rapport remains crucial for appreciating the team's efforts and achievements during the reporting period.

## **2. Project Control Techniques:**

### **2.1 Conflict Resolution through Resource Leveling:**

- Resource leveling addresses conflicts arising from resource assignments to multiple tasks.
- Adjustments can be made to avoid impacts when one task's delay affects another.

- Software tools like Microsoft Project automate conflict resolution by modifying impractical start or finish dates, managing task dependencies, and addressing resource over-allocation.

## **2.2 Critical Path Identification and Schedule Optimization:**

- PERT/CPM methods help determine the critical path of a project, allowing for the identification and removal of unnecessary slack.
- Optimizing the schedule involves parallelizing tasks, sequencing them, or splitting tasks to compress the schedule.
- Concurrent engineering methods, especially in software projects, enable planning downstream activities, facilitating parallel work and significant schedule compression.

## **2.3 Adaptable Corrective Actions:**

- Prompt corrective actions are crucial in identifying deviations in project monitoring.
- Increased costs, schedule overruns, or quality deviations demand specific management approaches.
- Robust planning addresses risks related to resource availability, while software engineering techniques ensure high-quality work products and finished products.

## **2.4 Prioritized Issue Management:**

- Issues must be categorized, and top-priority issues addressed first, considering their time sensitivity and potential impact.
- Assigning weights to each issue aids in prioritization, allowing for effective issue management.
- Addressing higher-priority issues promptly ensures minimal impact on the project, while lower-priority issues can be handled as time permits.

## **2.5 Comprehensive Resource Optimization:**

- In outsourced projects, resource optimization involves monitoring expenses and ensuring productivity aligns with rising employee wages.
- Project portfolio management facilitates efficient resource allocation, with prioritizing tasks for higher-paid staff enhancing overall resource optimization.

## **3. Project Monitoring and Control Artifacts:**

- Artifacts such as PERT/CPM charts, network diagrams, EVM, resource charts, and reviews of requirement documents are crucial for project monitoring.
- These artifacts, alongside actual project cost, product quality, and schedule data, play a pivotal role in determining project productivity of size and quality.

## **4. Project Monitoring and Control in Iterative Model:**

### **4.1 Iterative Development Risk Management:**

- For iterative development projects, where planning and action occur at the iteration level, risks are managed by breaking the project into smaller iterations.
- A priority system for requirements or features aids in controlling iterations, ensuring high-priority features are completed even under unforeseen circumstances.

### **4.2 Agile Performance Measurements and Risks:**

- Performance measurements in agile projects assess features delivered per iteration, defects found per iteration, and team productivity in delivering features per person per iteration.
- These measures diverge from traditional waterfall models.
- Risks in iterative projects are often confined to initial iterations, with challenges like inaccurate effort estimation or unforeseen issues.

- Once stabilized, agile environments operate smoothly, with team members autonomously adhering to predefined roles.
- Refactoring poses a unique risk, potentially impeding subsequent iterations if not executed adeptly.

**Reflections on Case Study/course work:** Chapter 7 unveils a compelling case study, spotlighting a vendor's strategic handling of software development challenges, featuring aspects like strategic alignment, proactive contingency planning, and successful resolution of the "Appointment Scheduling Engine" in release 6.0. The reflections on the case study are as follows:

1. **Organized Review Meetings:** The project team conducts weekly review meetings, led by the project manager, ensuring proactive issue resolution.
2. **Effective Mitigation Strategies:** Contingency plans are in place, involving systematic analysis, solution finding, and risk elimination.
3. **Forward-Looking Risk Assessment:** The team addresses both past issues and potential future risks, taking a proactive stance.
4. **Schedule Adjustment:** Impact analysis prompts schedule adjustments, and, if necessary, overtime is considered for task completion.
5. **Tool Utilization:** Microsoft Project and TestTrack Pro are used for effective project tracking and defect management.
6. **Adaptive Problem-Solving:** Challenges with the "Appointment Scheduling Engine" was met by replacing testers, utilizing business analysts, and improving documentation.
7. **Collaborative Expertise:** Business analysts with product management experience collaborated, contributing to a successful testing strategy.
8. **Document Quality Improvement:** Identified deficiencies in documents led to proactive enhancements, ensuring successful testing and implementation.

**Some real-world applications related to the case study:**

- **Construction Projects:** In the construction industry, weekly review meetings, contingency plans for unexpected delays, and the utilization of tracking tools can streamline project management and help overcome challenges in implementing complex designs.
- **Healthcare System Implementations:** Implementing complex healthcare systems involves dealing with various stakeholders and potential risks. The case study's approach to collaborative expertise utilization and iterative problem-solving is applicable in the healthcare IT domain.

### **Collaborative Learning:**

In preparation for mid-term exams, our team strategically organized a study session to optimize academic performance and reinforce key concepts. Recognizing the proximity of midterms to the reading week, we efficiently divided the workload for our second project deliverable among team members. I took on the responsibility of developing the project plan, including the work breakdown structure and overseeing budget considerations based on my prior experience. This allocation will enhance my project management skills and ensure a comprehensive approach for both exams and future deliverables. Over the past two weeks, our team focused on completing Deliverable 2 for our AI-based academic advisor project, "Guidance Guru," emphasizing collective knowledge application. My active participation in pre-reading week study group sessions contributed to better exam preparation and delving into Chapter 7's case study deepened our understanding of project tracking software, like The Gant Project which I had used during my bachelor's.

**Further research/readings:**

Recently, I came across Asana's insightful guide, "Work Breakdown Structure (WBS): What Is It? " which explores the visual organization of project deliverables based on dependencies. Intrigued, I aim to explore more resources offering in-depth insights into project structuring. Additionally, Forbes' article, "How AI Is Revolutionizing Project Management: Three Use Cases," highlighted AI's transformative impact on project scheduling, prompting me to seek further articles and case studies on AI applications in diverse project scenarios. Gartner's "3 Budget Planning Hacks for AI Projects" was a valuable resource for understanding budget planning in AI projects, focusing on software, implementation, and governance expenses. Motivated by this, I'm keen to delve into additional readings that illuminate the financial aspects of AI projects and provide practical tips for avoiding unexpected overruns.

**Adjustment to goals:**

1. During the reading week, my team and I collaborated on the project deliverable with my aim of completing at least the project plan, and by the next week I'll be ready with the budget for reviewing it with the teaching assistant.
2. Additionally, I plan to conduct a comprehensive topic analysis for the posterathon, an upcoming event in the following weeks. I intend to discuss my findings with my group partner for further collaboration.
3. In the latest lecture, we delved into the solutions to the midterm questions, prompting me to allocate a substantial amount of time to thoroughly review the chapters where I had marked incorrect answers. My goal is to comprehensively reassess and enhance my understanding of the content to ensure a more accurate grasp of the material.
4. Furthermore, as a side project as there is no implementation required for the project I think I will also be starting Guidance Guru: An AI-based academic advisor so that I can enhance my development skills also be an exciting project to work which will also boost my resume