

SOEN 6841 – SOFTWARE PROJECT MANAGEMENT

NAME: Devanshu Kotadiya [40268999]

JOURNAL URL: <https://github.com/devanshu-kotadiya/Software-Project-Management/tree/main/Learning%20Journals>

DATES RANGE OF ACTIVITIES: 1/10/2024 – 21/10/2024

JOURNAL DATE: 02/11/2024

KEY CONCEPTS:

Importance of SCM:

1. Minimize confusion and create structure.
2. Guarantees accurate product configurations.
3. Mitigates legal liability by maintaining a record of activities.
4. Lowers life-cycle expenses.
5. Creates a stable work environment.
6. Improves adherence to standards.
7. Strengthens status tracking.

Characteristics of a Good CMS:

1. **Centralized Configuration Management:** Ensures functionality across teams with secure, role-based access.
2. **Version Control:** Supports accurate version control and the creation of new software versions.
3. **Auditable System:** Provides document verification and version tracking.
4. **Secure Access:** Role-based controls safeguard sensitive information while enabling access for remote teams
5. **Continuous Integration:** Automated smoke tests integrated with continuous builds maintain integrity and quality.
6. **Artifact Management:** Centralized storage of artifacts simplifies retrieval and enhances team collaboration.

Purpose of Configuration Management System (CMS)

1. Configuration Identification:

This involves defining the baseline components of the system, answering the question, "What is my system configuration?"

2. Configuration Control:

This provides a framework for preparing, evaluating, approving, or rejecting all changes throughout the system's lifecycle, addressing the question, "How do I manage changes to my configuration?"

3. Configuration Status Accounting:

This mechanism maintains a record of the system's evolution and reports on the traceability of changes made to the baseline throughout the software lifecycle, allowing for inquiries such as, "What changes have I made to the system?" and "What changes are yet to be implemented?"

4. Configuration Audits:

This ensures that the system being developed meets the specified requirements by establishing a baseline and verifying that software configuration management (SCM) processes and procedures are being followed. This answers the question, "Does the system I am building fulfill the stated needs?"

Project Management Fundamentals

1. Top-Down Approach
2. Bottom-Up Approach

Top-Down Project Planning:

Involves setting predetermined release dates for the project that align with market needs.

Bottom-Up Project Planning:

Entails collecting information regarding the project's scope, requirements, and service level agreements (SLAs).

Work Breakdown Structure (WBS)

The Work Breakdown Structure (WBS) is a methodical approach to dividing the overall project work into smaller, manageable tasks. It also preserves the relationships between these tasks, allowing for an understanding of task dependencies, such as which tasks must be completed before others can begin.

Activity Organization

Activities within a project should be structured to generate concrete outputs that enable management to assess progress. Milestones represent the completion point of a process activity, while deliverables are the project outcomes provided to customers. The waterfall model facilitates the clear definition of progress milestones.

Activities: Milestones

Feasibility Study: Feasibility Report

Requirements Analysis: Requirements Definition

Prototype Development: Evaluation Report

Design Study: Architectural Design

Requirements Specification: Requirement Specification

Project Scheduling

Divide the project into individual tasks and assess the time and resources needed to complete each one.

Arrange tasks to be executed concurrently to maximize workforce efficiency.

Reduce task dependencies to prevent delays that arise when one task must wait for another to finish.

Two approaches: Calendar based, Activity networks based.

Graphical notations are often used to visually represent the project schedule. These notations break down the project into tasks, which should be appropriately sized—ideally taking about one to two weeks each. Activity charts illustrate task dependencies and highlight the critical path, while bar charts display the schedule in relation to calendar time.

Supplier Management Plan:

1. Service Level Agreement
2. Part Quality Check
3. Communication Plan
4. Central Configure Plan
5. Communication Integration

Project Planning Techniques

Critical path method: The critical path defines the total duration of the project. Tasks on this path are crucial, as any delays in these tasks will directly affect the overall project schedule.

Goldratt's Critical Chain Method: It suggests eliminating buffers for tasks that are well-defined and easily estimated. For tasks with uncertainty, buffers are separated, tracked independently, and then reinstated at the project's conclusion.

CASE STUDY REFLECTIONS:

The case study in Chapter 5 examines how a U.S.-based mid-market software vendor successfully implemented a centralized configuration management system. Key insights include the use of a central system for incremental iteration development, reinforcing the principles of configuration management and iterative processes. The vendor improved collaboration among internal, external, and offshore teams, which helped reduce costs and cycle times, highlighting effective teamwork strategies in global development. The system ensured security with 24/7 access and a two-tiered rights structure, addressing secure configuration management needs. Automated testing practices, including smoke testing for compatibility, enabled quick issue identification, aligning with discussions on automated testing. Developers also synced local builds with the central system to reduce failures, reinforcing the importance of version control and local development practices, while a robust workflow facilitated prompt issue escalation.

In Chapter 6, the case study focuses on a SaaS vendor's project planning using both iterative and top-down approaches. Major release planning involved fixed dates and prioritized features, aligning with project initiation discussions. The vendor addressed feature selection challenges through collaboration with the CTO. Iteration planning, influenced by release dates, introduced flexibility, resonating with agile management principles. The comprehensive planning included effort and cost estimates, risk assessment, and resource management, demonstrating the practical application of theoretical concepts in real-world scenarios.

PEER INTERACTIONS:

I connected with a former colleague, now a senior at Concordia University, to discuss configuration management techniques.

- During our conversation, I learned that configuration management is also essential for version control.

- He provided valuable insights into handling changes, merging, and configuration within various methodologies like spiral, agile, and rapid development.
- From a project standpoint, we meet twice a week to address general questions about the project and task allocation.
- Our initial tasks, including project setup and market analysis, are now complete, and we're looking forward to the next steps.

Additionally, I reached out to a few seniors for guidance on exam formats and to clarify some questions.

CHALLENGES FACED AND ADDRESSING THOSE CHALLENGES:

I encountered challenges in understanding merging after configuration and grasping the critical path concept. However, by reviewing the textbook case study and discussing with my former colleague, I was able to understand both concepts.

PERSONAL DEVELOPMENT ACTIVITIES:

I acquired an ebook on software management and reviewed key topics on software projects, various development models, management skills, team building, and project management.

- I already have a background in risk management and am now focusing on feasibility studies. Given some relevant topics in our project, I think this additional knowledge will be valuable.
- I'm also using the ebook to deepen my understanding of the CI/CD pipeline.
- I plan to review all chapters for exams, cover all the concepts, and resolve any questions using the ebook and other resources.

GOALS FOR NEXT WEEK:

As the professor advised, I need to review Chapters 7 and 8, so I'll aim to complete them this week. I'll continue working on project planning, specifically focusing on the feasibility study and risk management. The week will wrap up with a few group meetings about the project.

I'll also delve into risk mitigation and the feasibility study, as these are our next major topics for the digital skill platform project.