Learning Journal 1

Student Name: Masoumeh Farokhpourshekalgourabi

Course: Software Project Management

Journal URL: Learning Journal1.docx

Dates Rage of activities: 23.1.2025-28.1.2025

Date of the journal: 28.1.2025

Key Concepts Learned:	**Application in Real Projects:**	**Peer Interactions:**	**Challenges Faced:**	**Personal development activities:**	**Goals for the Next Week:**
Introduction to Software Project Management, including the definition of project management, processes involved in software projects, and characteristics of successful project managers. Importance of project charters, defining scope, objectives, and initial planning for software projects. Effort and cost estimation methods, such as Function Point Analysis, Wideband Delphi, and COCOMO models. Risk management techniques, identifying risk categories (budget, time, resources), and strategies for mitigation. Configuration management and version control, focusing on maintaining stability in iterative development environments.	Effort estimation techniques can be applied to Agile and Waterfall methodologies for accurate project planning. Risk management is critical in real-world projects to mitigate scope creep and resource constraints. Configuration management is essential for maintaining software stability and consistency across versions. Challenges: Implementing COCOMO models in large-scale projects requires deep familiarity with project data, which can be time-intensive.	Participated in group discussions about the pros and cons of using Agile vs. Waterfall models in software projects. Insights gained: Collaboration highlighted the importance of tailoring project management techniques to team size and project scope.	Difficulty in mastering the mathematical aspects of the COCOMO model for effort estimation. Understanding how to balance risk management with agile flexibility required additional research.	Watched online tutorials on COCOMO and Function Point Analysis to strengthen theoretical understanding. Experimented with version control systems (e.g., Git) to better grasp configuration management concepts.	1. Practice more examples of cost estimation using Function Point Analysis. 2. Explore real-world applications of risk management frameworks. 3. Enhance skills in configuration management tools and integrate them into small projects.