Learning Journal Week 4
Student Name: Jasmanpreet Kaur Bedi
Course: SOEN-6841: Software Project Management
Journal URL: https://github.com/jasmanpreet0209/Software-Project-Management-Learning-Journals
Week 2: February 11 – February 17
Date: February 16

Key Concepts Learned:

This week started by reflecting on the previous week's learnings about configuration management, its parts and strategies that can be made to deploy a configuration management system successfully for a project. Moving further, I learned about concepts in chapter 6:

1. **Project Planning**

   When planning a software project, quality, time, cost, and benefits for the organization need to be balanced. For projects within a company, these benefits include things like gaining more customers, cutting costs, managing risks, and following rules. Users benefit from easier work and improved performance. Knowing about these concepts can help a project manager satisfy customers better. For example, if the project's main goal is to make staff more productive, the manager might design the software to require less input, making users more efficient. In external projects, the goal is to make a profit. So, when planning resources, the project manager must be careful not to impact the project's profits.

   Project planning can be done using top down or bottom-up approach, depending on the time frame and scope of the project.

   In a **top-down approach**, release dates are fixed, and the team works within a specified time frame to meet market demands. This method involves planning components such as project scope, SLAs, and chosen software engineering models. A **bottom-up approach** involves gathering information about project scope, requirements, and SLAs at the beginning. The project team then defines the development strategy and creates a detailed project plan based on available inputs.

2. **Components of Software Project Planning**
   a. *Work Breakdown Structure (WBS):* Tasks are included in the plan with start and end dates. WBS helps identify task dependencies, critical paths, milestones, and phases, making it easier to manage.
   b. *Resource Allocation:* Software projects have varying staff requirements over phases, with about 50% of resources needed during construction and 30% for testing. Concurrent engineering models are used to reduce project cycles.
   c. *Supplier Management Plan:* Needed for projects done by outsourced teams, including SLA creation and compliance, ensuring quality and integration of outsourced work into the main build.
   d. *Configuration Management Plan:* Ensures all teams have the same version of source code and documents, utilizing a centralized system with high-quality security and access control.
   e. *Communication Management:* Depends on project organization structure, customer management, and supplier needs. Involves defining what, how, and when to communicate among project teams.
   f. *Defect Prevention Strategy (Quality Assurance):* Integral for project success, involving validation and verification of work products after each phase.

g. *Project Duration:* Calculated using the critical path, representing the longest path in the project. The length of the critical path determines the project duration.
h. *Project Cost:* Begins with effort estimation, incorporating resource costs and overhead expenses. Detailed information is available in Chapter 3.
i. *Tool Management:* Involves planning for the selection of programming languages, software, hardware platforms, productivity tools, and other resources.
j. *Scope Management:* Crucial for defining the volume of work based on requirements and quality levels.

Effort Estimate and Risk Management have been discussed in previous chapters.

3. **Project Planning Techniques:**
   a. **Critical Path Method (CPM):** It was developed to handle project scheduling, especially when tasks run in parallel. Tasks are laid out in order based on start dates, dependencies are identified, and a critical path is established by adding the times for the activities in each sequence and determining the longest path in the project. The critical path defines the project duration, and its length determines the project timeline.
   b. **Goldratt's Critical Chain Method:** It Introduces the theory of constraints, focusing on protecting projects from threats and identifies fixed and variable tasks, removing buffers for well-understood tasks but providing buffers for uncertain ones. Monitors project success by tracking buffer consumption rather than task durations. It Helps reduce uncertainty in projects, ensuring more certain project deliveries. I also learned about bar charts, activity networks, task durations, dependencies, activity bar, staff allocation.

4. **Project Planning in Agile Models:** It is suitable when requirements are unclear or small deliveries are preferred. Agile models, like Scrum and extreme Programming, involve iterations from requirements to release. Planning in Agile differs from the waterfall model, with top-level product plans, major release plans spanning a year, and detailed iteration plans. Agile models emphasize customer feedback, adaptability, constant resource requirements, and easier resource management. Iteration planning is based on velocity, measured in feature points per iteration, with features analyzed for size and complexity. Risk factors like refactoring time and potential issues are considered in planning. Agile models allow for flexibility and responsiveness in project execution.

Reflections on Case Study/course work:

In this case study, a Project Management Office (PMO) at a SaaS vendor handles organization-level management for IT projects. The PMO is involved in resource planning, business planning, and infrastructure planning to fulfill the business needs of the parent organization. At the project and iteration levels, planning includes identifying and prioritizing features, allocating tasks to resources, and implementing iterations. The case study highlights the challenges of feature selection between the marketing and development teams and how the Chief Technology Officer plays a crucial role in finalizing features. It also emphasizes the balance between flexibility and responsiveness in agile models, particularly in the context of fixed release dates and the time-boxing concept. The vendor uses a time-boxing approach to handle the absence of fixed iteration plans, which allows for flexibility in implementing features based on priority while maintaining responsiveness to customer needs. Overall, the case study gives insights into the practical application of project planning concepts as mentioned above at the PMO, project, and iteration levels in an agile environment.

Application in Real Projects:
Project planning is very important in real projects, throughout the entire project lifecycle. It involves defining the goals, organizing tasks, allocating resources, setting timelines, and managing risks. The Critical Path Method (CPM) helps in identifying and prioritizing tasks, that allows teams to focus on critical activities to meet project deadlines. Goldratt's Critical Chain Method is also important for managing uncertainties and risks and provides a structured approach to buffer tasks and maintain project timelines.

Agile models are especially important when project requirements are constantly changing. Iteration planning in Agile allows for continuous adaptation to changing needs, ensuring that features are developed incrementally, and customer feedback is incorporated continuously. This approach helps manage uncertainty and enhances responsiveness throughout the project lifecycle.

Peer Interactions/Collaborations:
I collaborated with my peers throughout the week to review the final delivery of Project Phase 1, focusing on problem identification and market analysis. Additionally, we met to finalize our project name and to discuss and rehearse the presentation for the Project pitch. We also discussed about how we incorporated Concepts learnt in the class to our project and how we can do it better in the further project deliverables.

Challenges Faced:
Doing online research for the market analysis of our project deliverable was a challenging task as there were numerous businesses that were doing a similar thing to our problem statement. Identifying the ones more related to our specific use case was a very time-consuming task. Making a summary of each competitor and their advantages and disadvantages as well as coming up with specific features for our project was challenging.

Personal development activities:
I got a deeper insight into project planning, and I aim to try to apply these concepts in the real world as well starting from our group project. Moving forward, I also aim to excel in project pitching and public speaking. Recognizing the power of effective communication, I am committed to refining these skills to confidently present projects, engage audiences, and demonstrate the value of our proposals.

Further Research/Readings:
The paper J. W. Horch (1996) Practical Guide to Software Quality Management, Artech House, Boston, MA really caught my interest and I aim to read about the Software Quality Management in detail.

Adjustment to Goals:
I'm going to review all the chapters for the midterm exam. Also, I saw that they started phase II of the project, so I'll begin studying the project requirements soon. This way, I'll be ready for both the exam and the next steps in the project.