**Learning Journal**

**Student Name:** Amal Saji

**Course:** Software Project Management (SOEN – 6841)

**Journal URL:** https://github.com/amalsaji25/SOEN-6841-Software-Project-Management

**Week6:** 18/02/2024 – 09/03/2024

**Date:** 09/03/2024

**Key Concepts Learned:**

During the sixth week focus was given to understand Project Control Techniques in a software project. In a project there are many risks and uncertainties. The Project manager has to keep the balance with many trade-offs to keep the project on track.

* In project management, ***resource leveling*** is a way to avoid problems caused by limited resources. Imagine a team member working on several tasks at once. If one task gets delayed, it can push back the start of another. Resource leveling helps prevent this. If the schedule allows, the project manager can adjust the order of tasks. Another option is to simply assign more people to the delayed task. Project management software can help identify these issues and even fix them automatically. This ensures the project runs more smoothly and avoids last-minute mix-ups.
* To create the most efficient project plan, project managers need to remove any extra time buffers, especially along the most important sequence of tasks (critical path). They should find and eliminate these buffers in all possible critical paths, with the longest path being the truly critical one. If tasks on this critical path finish early during the project, the project manager can adjust the schedule by shortening this path or starting tasks that depend on it a little sooner. Another way to ***optimize the schedule*** is to do some tasks at the same time, break down large tasks into smaller ones, or use techniques that allow for design and engineering to happen simultaneously. This can significantly speed up the project.
* Project monitoring reports help identify deviations from the plan, like increased costs or schedule delays. To fix these issues, project managers need to find the root cause. Common reasons for cost increases include higher overhead expenses or unplanned work. Schedule delays can be caused by inaccurate estimates, resource problems, or scope creep. For all these issues, good data collection is key to making informed decisions and getting the project back on track.
* Effective project management relies on categorizing problems by importance. With many issues, it's easy to feel overwhelmed and see them all as equally critical. To overcome this, project managers assign weights to each problem based on its severity and how soon it needs to be addressed. This helps sort and prioritize effectively. If time allows, tackling multiple issues at once based on their weight and urgency can prevent problems from derailing the project. The most important issues, with the highest weight, are addressed first to minimize risk.
* In outsourced projects, the project manager needs to balance the interests of both sides. While the company seeks benefits, the service provider aims to make money. During the project, unforeseen events like rising wages can threaten the provider's profit. To address this, the project manager plays a key role. By ensuring that worker output keeps pace with salary increases, they can help the provider maintain profitability. One way to achieve this is through project portfolio management. This approach optimizes how people are assigned across projects, minimizing downtime and maximizing the value of highly-skilled (and likely higher-paid) staff.
* Overseeing a project's progress is crucial to keeping it on target. Careful observation of the work underway helps identifies problems early on. These problems could be extra costs, delays, or quality issues. To stay in control, managers track the project's schedule, budget, and how well it meets goals. By doing this, they can make sure resources are used wisely and the project delivers what it's supposed to, while meeting quality expectations.
* Closely monitoring the ongoing work allows managers to spot issues like cost overruns, delays, or quality problems before they snowball. To maintain control, they track the project's timeline, budget, and how well it aligns with the set objectives. This monitoring helps them guarantee that resources are used efficiently and the project delivers the intended results while meeting quality standards.
* In iterative software development, planning concentrates on short bursts of work, typically lasting a few weeks to a couple of months. This spreads out risks and makes them easier to handle. Sometimes, big changes in what the customer wants might require major adjustments to the current iteration's plan, throwing everything off track. This can make it difficult to monitor and control the project. If this happens, a whole new plan might be needed to fit the revised requirements. However, prioritizing features within each iteration can be a helpful way to keep the project on schedule. By focusing on the most important features first, the team can be sure to get them done on time, while less important ones can wait or be added if there's extra time.
* Agile projects track success differently than traditional ones. Instead of focusing on big milestones, they measure progress in smaller chunks called iterations. During each iteration, teams track how many features they deliver, how many bugs they find, and how productive everyone is. This helps them understand how well the project is going and make adjustments as needed
* In agile development, projects are broken down into short, focused phases where teams tackle a set amount of work. While initial phases may face hurdles due to estimating effort or unexpected problems, agile teams typically find their groove after a short period. These teams consist of members with clear roles who independently manage their workloads within each phase, prioritizing tasks based on user stories. This might involve writing unit tests or occasionally improving the code's design to ensure new features integrate smoothly.

**Reflections on Case Study:**

The case study explores into the project control techniques employed by a SaaS vendor, focusing on managing issues and risks during project execution. The vendor conducts major and minor software releases aligned with yearly project plans and iterations within those plans, respectively. To mitigate risks and issues, the team implements contingency plans and holds weekly iteration review meetings led by the project manager. These meetings address current issues and potential risks, allowing for proactive planning and action. Mitigation strategies include causal analysis, root cause identification, solution implementation, and impact assessment on schedules. Tracking tools such as Microsoft Project and Test Track Pro are utilized for project monitoring, resource allocation, and defect tracking. Additionally, the case study highlights a specific challenge faced during the development of a complex feature called the "*Appointment Scheduling Engine*." Through well-defined planning, testing, and resource allocation, the project team successfully addressed the challenges, resulting in the engine functioning as intended, showcasing effective project control techniques in action.

**Collaborative Learning:**

This week we were able to discuss and delve into project controlling techniques and explored in the dynamics of agile development methodologies. We explored the notion of time-boxed iterations, emphasizing the importance of completing the feature points within the designated time frames to ensure project progress. We brought up the challenges faced in early iterations, such as inaccurate effort estimation and unexpected issues, which can impede project momentum. We also explored the significance of roles in agile teams, noting that well -defined responsibilities empower team members to autonomously tackle tasks and adapt to changing project needs. Additionally, we discussed on the critical role of continuous refactoring in mitigating design issues and maintaining code integrity across iterations, recognizing it as a pivotal aspect of project control in agile environments.

**Further Research/Readings:**

On further reading on project control, I was able to gather the following key practices for agile teams. First, clear communication channels are crucial to keep everyone informed and aligned. Second, tools like burn down charts and Kanban boards provide real-time status updates and help identify roadblocks. Third, planning for potential issues with risk assessments and contingency plans helps minimize surprises. Finally, regular feedback from stakeholders ensures the project stays on track and meets their needs. By incorporating these elements, agile teams can effectively control projects and deliver value.

**Adjustments to Goals:**

I will need to figure out more advanced communication for agile teams, like running meetings and a collaborative environment. I also need to explore on project tracking tools like charts and boards to monitor progress. Additionally, I will need to put bit more focus on risk management techniques like planning for issues that can help me handle unexpected challenges. Through this I aim to be more proficient in project control and learn in depth on agile project management.