Learning Journal 2

Student Name: Gaurav Sharma(40284996)
Course: SOEN – 6841 Software Project Management
Journal URL: **https://github.com/gauravsharma2802/LearningJournalSPM**
Week 3(Chapter 4): 23rd September 2024 – 29th September 2024
Date of the journal: 29th September

Key Concepts Learned: In Week 3, the key concept I learned is about **risk management**, it provides important concepts about the definition of risk as the combination of probability and negative consequences. Many categories are identified, including **estimation errors or technical risks**. The process of assessing risks involves **identification, analysis, prioritization** according to **likelihood and impact**. Response strategies include **acceptance, avoidance, transference, and mitigation.** Continuous control over risks involves activities such as **monitoring and resolution**. **Iterative models** help reduce risks because early feedback means improvements in accuracy at the requirements level.

Application in Real Projects : In real-world software projects, applying risk management is essential for success. For example in **e-commerce platform** during peak traffic times like Black Friday. The team can foresee risks of **server crashes** and mitigate them by employing **cloud-based scalable infrastructure**. An innovative approach could be using **predictive analytics** to monitor performance, preventing issues before they arise. There are challenges, such as being able to persuade the stakeholders to adapt new tools, but proactive risk management is in itself a success with minimum disruption to the project.

Peer Interactions: This week, peer discussions were particularly enlightening as we explored the importance of **risk management** if the project goes live for customers. We discussed potential risks such as **system downtime** and **security vulnerabilities**, and one peer suggested the use of **automated testing** and **cloud scalability** to mitigate these issues. This feedback helped me understand the importance of **proactive planning** in preventing major problems and highlighted the need to address **scalability** and **security** early on. The discussion enhanced my understanding of **risk management strategies** and fostered collaborative learning.

Challenges Faced: The biggest problems I faced was with **quantitative risk analysis**, especially in assigning **accurate probabilities** and impacts to risks, as this process often lacks clear, reliable data to base decisions on. Additionally, **prioritizing risks** was difficult because it requires balancing risks with different levels of likelihood and impact, which can be subjective. Another challenge was in selecting the appropriate **risk response strategy** (**acceptance, avoidance, transference, or mitigation**), as each choice has potential long-term effects. Lastly, applying these concepts to real-world projects was challenging due to project-specific uncertainties.

Personal development activities: This week, I continued working on my personal project, specifically focusing on the risk management aspects. I applied what I learned by **identifying potential risks, analyzing their likelihood** and impact, and developing appropriate **mitigation strategies**. This activity helped me gain hands-on experience in managing risks, allowing me to strengthen my **decision-making and planning skills**. By addressing real-world risk scenarios, I'm improving my ability to foresee challenges and effectively mitigate them in future projects, contributing to my overall professional growth

Goals for the Next Week: Next week, I plan to thoroughly read Chapter 5, focusing on specific concepts related to risk management, and apply these techniques directly to my personal project to enhance my ability to handle project risks. Additionally, I will collaborate with my teammates to finalize **our first deliverable and practice for the upcoming pitch**, ensuring that we meet all project requirements. I will be reviewing some of the concepts I studied this week, such as the **identification of risk and mitigation strategies**. The findings from these objectives helps me to meet not only immediate tasks of the project at hand but also enhance my long-term skills in the management of projects, more so in the aspect of risk assessment that makes part of the development of my career goals in the management of complex project

Student Name: Gaurav Sharma (40284996)
Course: SOEN – 6841 Software Project Management
Journal URL: **https://github.com/gauravsharma2802/LearningJournalSPM**
Week 4(Chapter 5): 29th September 2024 – 4th October 2024
Date of the journal: 4th October

Key Concepts Learned: In week 4, I learned about **configuration management (CM)**, which is essential for controlling and documenting changes in software projects to ensure success and discipline. CM addresses changes that arise from evolving requirements, technological advancements, and shifting customer expectations. **Software Configuration Management (SCM)** helps prevent project chaos, delays, and quality issues by maintaining traceability and order. The benefits of CM include reducing confusion, ensuring compliance, maintaining product integrity, and lowering life-cycle costs. The **core functions** of CM involve **configuration identification** (defining baselines), **configuration control** (managing changes), **configuration status accounting** (tracking changes), and **configuration auditing** (ensuring standards are met). Additionally, a formal **change control policy** ensures all changes are properly documented, approved, and traceable, which is crucial for maintaining project stability and success.

Application in Real Projects: In real-world software projects, **configuration management (CM)** is vital for tracking **versions and ensuring changes are traceable**, helping teams maintain order and compliance. It prevents issues like version conflicts and reintroduced bugs, especially in large, collaborative environments.
Challenges arise in fast-paced agile projects, where frequent changes make CM more difficult, but tools like Git can streamline the process. By applying CM in real-world software projects enhances traceability, ensures compliance, and provides stability, but also requires creative solutions to manage challenges, particularly in fast-paced development environments.

Peer Interactions : This week, I had insightful discussions with my peers about the importance of **configuration management** (CM) in preventing issues **like version conflicts and reintroduced bugs**. We shared experiences from **past projects** and explored the challenges of managing frequent changes **in agile environments**. One key insight was the necessity of a structured change control policy, even in fast-paced development. A peer introduced the idea of using AI-driven tools to predict configuration conflicts, which expanded my understanding and inspired me to explore **innovative solutions**. These interactions enhanced my learning, providing valuable perspectives on applying CM more effectively in **real-world scenarios**.

Challenges Faced : This week, I faced challenges in understanding both **configuration auditing** and the **impact analysis report template**. In configuration auditing, I struggled with how the process verifies that a system aligns with its baseline across multiple versions and how concepts like **traceability**, **baselines**, and **configuration items** are applied in large-scale projects. Additionally, understanding the **Impact Analysis Report Template** was difficult, particularly in estimating factors like **relative benefit**, **penalty**, **cost**, and **risk** accurately. I also needed more clarity on estimating **total effort**, **lost effort**, and **schedule impact**, as I wasn't sure how to measure these aspects effectively. Both topics required extra effort to grasp how they contribute to maintaining project stability and making informed decisions

Personal development activities: This week, I have taken a closer look at **configuration management** for my own project: version history is maintained via **Git**. With Git, I also looked into **SVN, Bitbucket by Atlassian**, since both offer advanced features in version control and collaboration. Then I began investigating **Ansible**, which goes beyond version control and supports the automation of configuration management for large projects. While playing with these tools, I came across various different tools that handle software changes and configurations. All of these activities contributed towards my professional growth, since now I have a broader perspective about configuration management tools and how to apply them in field projects.

Goals for the Next Week: Next week, I plan to **read Chapter 6 and** understand its concepts to apply them to my personal project, focusing on **configuration management** and **process automation**. I will also **revise previous class topics** for an upcoming quiz. In terms of l**ong-term growth**, I aim to strengthen my skills in **software development and DevOps practices**, aligning with my career goal of mastering **automation** and **project management**. These activities will support both my **academic success** and **career development** in software engineering.