**Learning Journal 3**

**Student Name:** Weichen Wan

**Course:** SOEN 6841 Software Project Management

**Journal URL:** https://github.com/HWJFish/Software-Project-Management-Learning-Journal

**Dates Rage of activities:** Nov.2nd -Nov.9th

**Date of the journal:** Nov.8th

Key Concepts Learned:

This week, we learnt chapters 8 and 9 which focus on project closure and software development life cycle. For Chapter 8, we have covered project deliverables, source code version management, project measurement metrics filtering and archived, and we reviewed some content covered in this semester. For Chapter 9, we went through and compared different software lifecycle models mainly for waterfall model compared to iterative models including SCRUM and Extreme programming. Although iterative models are way more popular because of their flexibility to change based on the market and project execution, the waterfall is still better for projects with stable requirements.

Application in Real Projects:

Project closure is a necessary stage that many people ignore. As a student who mainly has school project experience, it is even worse since we only focus on submission most of the time. It is important to organize documentation, code versioning and metrics for future maintenance and to get experience and data for future projects. For example in real-world projects like our course project, closure activities can include organizing deliverables, filtering data for metrics analysis and learning from the development experience. For planning life cycle models, iterative models are very popular and widely used. For example, for a wedding app, vendor selection can be done in early sprints to make the app work. With later sprints, we can add features sprint-by-sprint for RSVP management, budget tools and more.

Peer Interactions:

For this week, we have discussed project lifecycles and project closure with the team. We have shared our experience of how to do the project closure properly. We have also compared different documents and metrics we need to work on, and we think that the risk management document is a part we don’t really look back on during the closure, but we should pay more attention to what kind of risks come true and how do they affect our project to have a better estimation and risk management strategy for the future. Also, we believe that we can learn from metrics like scope completion and earned value management to know how we can adjust our plan during project execution.

Challenges Faced:

One of the challenges I faced was that I found it quite hard to find how to do the project closure properly. For most of my previous projects, closures are not that important, and we hardly analyze metrics for the future. Therefore, I have checked online for how companies do the project closure as case studies, especially for what kinds of documents and what kinds of metrics they use and keep for the future. For learning software lifecycles, I realized I have many experiences with iterative-based approaches but have no experience with waterfall. I have difficulty understanding why we need a waterfall for the same cases. Therefore, I did some research on why we still need waterfall nowadays and how are they different compared to agile. I have learnt that projects like embedded systems where it is hard to provide updates after launch and critical systems like government websites still rely on waterfall. I also realized we need to pay more attention to planning stages such as requirements gathering since it can be very costly to change them since almost everything needs to be planned in advance.

Personal Development Activities:

This week, I have done many case studies and research as described in the previous section for project closure and software lifecycle models. I have also learnt the internal differences between different methodologies like SCRUM and Extreme Programming. Besides, I got hands-on practice in management with Jira for serval ongoing projects to manage WBS, progress and issue tracking. In addition, I have read some papers and did some summaries for software refactoring. Refactoring is another part that tends to be ignored in project management. From these papers, I got some insight into how to manage refactoring for projects and some trending topics for refactoring with AI.

Goals for the Next Week:

Next week, I plan to try to apply the methodologies we learned in class for project closure since many school projects this semester are close to the end. I plan to summarize and learn from these projects' experiences which I usually ignore for previous projects. Also, I plan to read more case studies about how the software industry moves from waterfall to agile to get a better understanding of both waterfall and agile, so that I can gain some insight into how the industry changes over time so I can prepare myself for the upcoming changes like AIs.