**Final Learning Journal**

**Student Name:** Arik Kantesaria

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

**Journal URL:** <https://github.com/Arik39/SOEN6841_SPM>

**Dates Rage of activities:** 11th November 2024 to 21th November 2024

**Date of the journal:** 22nd November 2024

**Key Concepts Learned**

1. **Requirement Management :**
   * Understanding and managing customer requirements to minimize rework.
   * Systematic handling of changes with processes like requirement validation and change management cycles.
2. **Software Design Management :**
   * Top-down and Bottom-up design approaches for modular and holistic planning.
   * Techniques such as prototyping, refactoring, and object-oriented design for robust development.
3. **Software Construction and Testing :**
   * Importance of coding standards, code reviews, and defect tracking in iterative models.
   * Testing frameworks, including manual and automated approaches for ensuring software quality.
4. **Release and Maintenance :**
   * Managing product releases with end-user training and documentation.
   * Types of software maintenance, such as preventive, corrective, and adaptive, for post-release support.

**Application in Real Projects**

* Requirement gathering and validation were applied to simulate real-world scenarios, enabling smoother adjustments during project development.
* Using top-down approaches for initial project frameworks and bottom-up methods for independent module designs streamlined resource allocation and task management.
* Automated testing tools were used in iterative projects to optimize regression testing and maintain reliability across product versions.
* In a mock product release, lessons learned and proper documentation were integrated to enhance the project's closure activities and future usability.

**Peer Interactions**

* Discussions on estimation techniques clarified my understanding of methods like FPA and COCOMO, enhancing my effort estimation skills.
* Peer feedback on the pros and cons of agile versus waterfall models broadened my perspective on software lifecycle management.
* Collaborative exercises in SCRUM highlighted the challenges of iterative development and provided practical insights into managing dynamic requirements.

**Challenges Faced**

* Handling frequent requirement changes without disrupting the design's integrity required continuous refinement.
* Balancing manual versus automated testing posed challenges in managing iterative workflows effectively.
* Structuring concise yet actionable lessons learned documentation demanded thoughtful reflection on project lifecycles.

**Personal Development Activities**

* Practiced requirement gathering and change management with real-time scenarios.
* Explored project management tools like JIRA, Microsoft Project, and TestRail to enhance scheduling, testing, and defect tracking processes.
* Conducted mock product releases and SCRUM iterations to strengthen my familiarity with agile methodologies.

**Goals for the Future**

* Master advanced testing strategies and automation tools for enhanced quality assurance.
* Apply refactoring techniques to improve legacy systems while ensuring compatibility with modern requirements.
* Deepen my understanding of configuration and risk management for large-scale projects.

**Final Reflections**

**Overall Course Impact**

This course has transformed my understanding of software project management. By emphasizing structured approaches to planning, execution, and monitoring, it has provided me with essential tools for managing dynamic and complex projects effectively. Key insights include the importance of clear requirement management, accurate estimation, and robust lifecycle techniques to ensure project success.

**Application in Professional Life**

The knowledge gained is directly applicable to professional scenarios:

* Using estimation techniques like COCOMO for resource planning.
* Implementing SCRUM for projects with rapidly changing requirements.
* Ensuring quality through defect tracking and automated testing.  
  These skills will enable me to manage projects more effectively and deliver better outcomes in my professional role.

**Peer Collaboration Insights**

Collaborating with peers was invaluable in reinforcing theoretical knowledge with practical applications.

* Discussions on real-world challenges, such as iterative development and risk mitigation, offered diverse perspectives.
* Feedback and shared experiences enhanced my ability to adapt methodologies to different scenarios.

**Personal Growth**

This course has refined my analytical thinking, technical proficiency, and leadership skills. I am more confident in managing projects end-to-end, utilizing modern tools and frameworks, and navigating the complexities of software development.