Student Name: Darshansinh Devda (40261713)

Course: SOEN 6841 Software Project Management

Journal URL:
https://github.com/Darshansinh28/SOEN-6841-Software-Project-Management

Week: October 15 - October 29

Date: 9 November 2024

Key Concepts Learned:

- Project Closure
  - **Definition**: The final phase of project management, where all activities to complete the project are finalized.

- Activities in Project Closure
  - **Final Deliverables:** Ensuring all deliverables are complete and meet quality standards before formal project closure.
  - **Source Code Version Management:** Finalizing and documenting the source code version history for traceability and maintenance.
  - **Data Filtration for Archiving:** Filtering and archiving project metrics and data for future reference and lessons learned.
  - **Lessons Learned:** Reviewing project insights, successes, and areas of improvement to support future projects.

- SDLC (Software Development Life Cycle)
  - **Definition:** Stages from requirements to deployment.
  - **Phases:** Requirements, design, development, testing, deployment.
  - Lifecycle Models

- Lifecycle Models
  - **Waterfall:** Sequential, stable projects.
  - **Iterative:** Flexible, suited for changing requirements.

- Quality Assurance
  - **Definition:** Ensures quality through checks at each SDLC phase.

- Customer Requirements
  - **Definition**: End-user needs that guide the development process.

- Requirement Gathering
  - **Definition**: Interacting with users to collect and analyze their needs, forming a structured set of requirements.

- Requirement Management
  - **Definition**: Ensuring changes to requirements are recorded and integrated with minimal rework.

- Types of Requirements
  - **Functional Requirements**: Core functionalities of the software.
  - **Non-Functional Requirements:** Quality attributes like performance, security, and usability.

- Requirement Levels
  - **Definition:** Hierarchical organization of requirements, from high-level goals to detailed specifications.

- Sources of Requirements
  - **Definition**: Origins of requirements, including users, stakeholders, and market demands.

- Process Cycles
  - **Change Management Cycle**: Manages updates to requirements throughout development.
  - **Validation Cycle:** Ensures requirements meet user expectations, particularly in iterative models.

Reflections on Coursework and Case Study:

1. **Foundational Learning:** This week focused on understanding the importance of assessing project feasibility, planning, and resource allocation. It emphasized addressing stakeholder concerns, proactively managing risks, and ensuring financial transparency by estimating and controlling project costs.
2. **Future Application**: In future projects, I will ensure clear planning, manage risks proactively, and keep track of budgets for financial transparency.

Peer Interactions:

1. **Work Distribution:** Allocated tasks for Phase II deliverables and set up the GitHub repository to ensure timely completion.
2. **Documentation Discussion**: Collaborated on creating key documents like feasibility studies and project plans, and used the GitHub repo for version control and project management.

Challenges Faced:

1. **Project Documentation and Planning:** Developed the feasibility study by addressing technical, economic, and scheduling factors. The solution proposal outlined methodologies and risk mitigation strategies. The project plan focused on resource allocation, contingency planning, and accurate cost estimation to ensure alignment with the budget.

Personal development activities:

1. **Reading**: Studied articles and case studies on project closure, requirements gathering, and risk management for real-world application.
2. **Networking**: Joined a project management discussion group to learn about managing customer requirements and quality assurance.
3. **Skill Development**: Learned to set up a GitHub repository for version control, project management, and task tracking.

Goals for the Next Week:

1. **Next Phase of Project**: Start the next phase of the Powered Resume Builder, ensuring clear deliverables and efficient task allocation.
2. **Baseline Check**: Check the baseline in Prophectr to confirm the project is progressing as expected, making adjustments where necessary to stay aligned with the initial plan.