Mohammad Ehtesham Arif
40269948
SOEN 6841

Weekly Journal – Week 1

Chapter 1

1.1 Project Management –
Any set of activities can be considered as a Project if they have a definite start time and end time and these set of activities achieve some pre-defined goals.
Project management involves initiating an activity with the aim of achieving predefined goals within constraints such as limited resources, budget, and time. Resources and budget are utilized during the project's finite duration, with any remaining resources and budget released upon project completion. A well-structured project closure is necessary at the end of project execution to bring proper closure to the project.

1.2 Software Project Management –
Software Project Management can be defined as applying project management and software engineering methods to develop/maintain a software product so that the goal of developing/maintaining a software product can be achieved using minimum possible resources and money and within the minimum time possible.

Project management processes may include project initiation, project planning, project monitoring and control, and finally project closure. The software engineering processes may include requirement development, software design, software construction, software testing, software deployment and software maintenance.

1.3 Problems in Project Management –

Some of the problems associated with project management are: -

- Immaturity of Engineering Practices - Software engineering is relatively new, with only about 60 years of existence.
- Evolutionary Nature of Requirements and Specifications - Constant evolution adds complexity to project management processes.
- Dynamic Evolution of Tools and Technologies - Tools, technologies, and project management models for software projects are still evolving.
- Education and Training Challenges - Personnel may lack the desired skills, contributing to challenges in project execution.

The impact of these challenges are Communication difficulties, team management, effort estimation, work distribution, project reporting, and work tracking.

One way to avoid this kind of drifting away from the focus is to have a defined project process and use this process map to chalk out a project plan as to which tasks will be done at what time, in what sequence, who will be responsible for these tasks, etc. This kind of planning based on a process structure is extremely useful for large and complex projects.

To be a **successful software project manager**, one should understand project management, software engineering, tools, and technology, manage team, customer, suppliers and work under organization frameworks.

1.4 Processes in Software Projects –

A project can be broken down into project phases. These project phases include project initiation, project planning, project monitoring & control and project closure.
We can classify the processes under the following categories: -

1. **Evolving processes beyond a project**:
2. **Project management processes (project initiation, planning, control, monitoring, and closure)**:
3. **Software development life cycle (SDLC) processes (requirements, design, build, testing, maintenance, etc.)**:

1.5 Software Project Initiation –

The first among project management processes is the project initiation process. We can further divide this into processes for application initiation, product initiation, and product implementation initiation.

1.6 Software Project Planning –

The project plan varies based on the chosen software life cycle. In the linear waterfall model, processes occur only once, while in iterative models, processes can repeat, especially between construction and testing, with the extreme iteration involving all software engineering processes.

1.7 Software Project Metrics –

Over the years, several metrics have been defined and used in projects. Many of these approaches use statistical process control (SPC) methods. One SPC approach is popularly known as the "Seven Tools of Quality". These are Check sheets, Histograms, Pareto charts, Cause and effect diagrams, Scatter diagrams, Control charts, Graphs.

Chapter 2

2.1 Project Initiation –

Software projects commence in a manner like other project types.
Project **charter** is made by the top management of the organization for starting a software project. Project charter basically defines the purpose for starting the project.

A detailed project **scope** is developed to define the boundaries of the project. The scope will include what functionalities are needed in the software product to be developed. It will also define the level of quality needed in the software product.

The project should have a set of well-defined **objectives** that must be met. If any of these objectives are not met upon completion of the project, then the project will be a failure. The stakeholders state and set the project objectives. The objectives should be stated in clear language and the set of objectives should be kept as small as possible.

2.2 Activities performed during project initiation –

- An initial project schedule is crafted by breaking down the entire project into smaller tasks, determining start and end times for each task.
- To enhance project effort and cost estimates, a method called project division can be employed. After creating the project charter and scope, an expert is hired to provide effort and cost estimates. Software development companies are then invited to bid for project planning and execution based on the expert's figures.

Estimating Initial Project Effort, Costs and Size

Project stakeholders prioritize cost considerations, and exceeding the budget may lead to project cancellation. An initial budget is estimated and sanctioned, covering salaries, hardware, services, travel, and management costs. Project costs are linked to the project's size, with effort estimation crucial in determining labor costs.

In the **iterative model**, project planning occurs at three levels. The top level involves creating a project plan for the entire product development, spanning several years. At the middle level, planning focuses on major releases, with a time span of around a year or half a year. The lowest level is the iteration plan, covering a few weeks to a few months.

Project objectives should be SMART – specific, measurable, achievable, relevant, time constrained.

Conclusion –

This week's study required a proactive approach to self-learning. The diverse range of topics necessitated independent research and exploration to deepen understanding. The resources provided served as a launching point for further investigation into specific areas of interest.

The self-critique process has been integral in recognizing areas of strength and weakness in my understanding of SPM principles. Regular reflection on my engagement in class discussions and the completion of assigned tasks has allowed for continuous improvement. Recognizing the need for improvement in certain areas, such as estimating project size and effort, I have proactively sought additional resources and engaged in extra practice to address these gaps. This self-awareness has been instrumental in refining my understanding of the material.

Maintaining a high level of readiness and willingness to study has been a priority throughout this week's course. A proactive approach to attending lectures, participating in discussions, and completing assignments has been essential in staying abreast of the extensive material covered.