Software System Engineering A Tutorial

Software systems have become larger and more complex than ever. We can attribute some of this growth to advances in hardware performance. But there are other reasons for increased size and complexity. It provides the cohesiveness and data control that enable a complex system to solve problems. Simply tracking a development project's managerial and technical status does not provide sufficient feedback about its health. we must manage the technical processes as well as its products. System engineering provides the tools this technical management task requires.

SYSTEMS AND SYSTEM ENGINEERING

A system is a collection of elements related in a way that allows a common objective to be accomplished. Systems engineering is an interdisciplinary field of engineering and engineering management that focuses on how to design and manage complex systems over their life cycles.

System engineering involves five functions:

• Problem definition determines the needs and constraints

• Solution analysis determines the set of possible ways to satisfy the requirements and constraints

• Process planning determines the tasks to be done, the size and effort to develop the product

• Process control determines the methods for controlling the project and the process

• Product evaluation determines the quality and quantity of the delivered product

WHAT IS SOFTWARE SYSTEM ENGINEERING?

Software Systems Engineering is the branch of systems engineering concerned with the development of large and complex software intensive systems. It focusses on: the real-world goals for, services provided by, and constraints on such systems; the precise specification of system structure and behaviour, and the implementation of these specifications; the activities required in order to develop an assurance that the specifications and real-world goals have been met; the evolution of such systems over time and across system families.

SwSE and Software Engineering

Software engineering is the application of principles used in the field of engineering, which usually deals with physical systems, to the design, development, testing, deployment and management of software systems.

The field of software engineering applies the disciplined, structured approach to programming that is used in engineering to software development with the stated goal of improving the quality, time and budget efficiency, along with the assurance of structured testing and engineer certification.

SwSE and project management

Project management as “the discipline of initiating, planning, executing, controlling, and closing the work of a team to achieve specific goals and meet specific success criteria. SwSE determines the technical approach, technical decisions, interfaces and approves and accepts the final software product.

THE FUNCTIONS OF SOFTWARE SYSTEM ENGINEERING

Requirements analysis

Requirements analysis encompasses those tasks that go into determining the needs or conditions to meet for a new or altered product or project, taking account of the possibly conflicting requirements of the various stakeholders, analyzing, documenting, validating and managing software or system requirements.

Software design

Systems design is the process of defining the architecture, modules, interfaces, and data for a system to satisfy specified requirements. Systems design could be seen as the application of systems theory to product development.

Process planning

Process planning specifies the project goals and objectives and the strategies, policies, plans, and procedures for achieving them. It defines in advance what to do, how to do it, when to do it, and who will do it.

Process control

Control is the collection of management activities used to ensure that the project goes according to plan. Process control measures performance and results against plans, notes deviations, and takes corrective actions to ensure conformance between plans and actual results. Process control is a feedback system for how well the project is going. Control must lead to corrective action —either bringing the status back into conformance with the plan, changing the plan, or terminating the project.

Verification, Validation, Testing

Verification determines whether the products of a given phase of the software development cycle fulfill the requirements established during the previous phase. Verification answers the question, “Am I building the product right?”

Validation determines the correctness of the final program or software with respect to the user’s needs and requirements. Validation answers the question, “Am I building the right product?”

Testing is the execution of a program or partial program, with known inputs and outputs that are both predicted and observed, for the purpose of finding errors.