

Lecture: 1
Software Process: Introduction

Software Engineering is an engineering branch associated with the development of software product using well-defined scientific principles, methods and procedures. The outcome of software engineering is an efficient and reliable software product.

We can view it as a systematic collection of past experience. The experience is arranged in the form of methodologies and guidelines. A small program can be written without using software engineering principles. But if one wants to develop a large software product, then software engineering principles are indispensable to achieve a good quality software cost effectively.

IEEE defines software engineering as:

“The application of a systematic, disciplined, quantifiable approach to the development, operation and maintenance of software.”

Program vs. Software Product:

- Programs are developed by individuals for their personal use. They are therefore, small in size and have limited functionality but software products are extremely large.
- In case of a program, the programmer himself is the sole user but on the other hand, in case of a software product, most users are not involved with the development.
- In case of a program, a single developer is involved but in case of a software product, a large number of developers are involved.
- For a program, the user interface may not be very important, because the programmer is the sole user. On the other hand, for a software product, user interface must be carefully designed and implemented because developers of that product and users of that product are totally different.
- In case of a program, very little documentation is expected, but a software product must be well documented.
- A program can be developed according to the programmer’s individual style of development, but a software product must be developed using the accepted software engineering principles.

Causes and Solutions for software crisis:

- Organizations are spending larger and larger portions of their budget on software. Not only are the software products turning out to be more expensive than hardware, but they also present a host of other problems to the customers:
 - software products are difficult to alter, debug and enhance
 - use resources non-optimally
 - often fail to meet the user requirements
 - are far from being reliable
 - frequently crash
 - and are often delivered late.
- The cost incurred on software have increased due to ineffective development of the product characterized by inefficient resource usage and time and cost over-runs thereby contributing to the making of the present software crisis.
- Other factors contributing **are larger problem sizes, lack of adequate training in software engineering**, increasing skill shortage and low productivity improvements.

Need of Software Engineering:

The need of software engineering arises because of higher rate of change in user requirements and environment on which the software is working.

- **Large Software:** It is easier to build a wall than to a house or building likewise, as the size of software become large engineering has to step to give it a scientific process.
- **Scalability:** If the software process were not based on scientific and engineering concepts, it would be hard to scale an existing one.
- **Cost:** As hardware industry has shown its skills and huge manufacturing has lower down the price of computer and electronic hardware. But the cost of software remains high if proper process is not adapted.
- **Dynamic Nature:** The always growing and adapting nature of software hugely depends upon the environment in which the user works. If the nature of software is always changing, new enhancements need to be done in the existing one. This is where software engineering plays a good role.
- **Quality Management:** Better process of software development provides better and quality software product.