Module: **Software Engineering(SE)**

Module Code: CTE303

Semester-V,   BE(IT) Programme

Tutor: Tsheten Dorji

Department of Information Technology
College of Science & Technology, RUB

Chapter 1

Introduction to Software Engineering

Overview

- **Definition of some terminology**
- **Software characteristics**.
- **Software Crisis**
- **Software Applications**.

What is Software Engineering?

- Software Engineering is an **engineering discipline** that is concerned with **all aspects of software production** from the early stages of system specification to maintaining the system after it has gone into use.

- Two key phrases in this definition:
  - Engineering discipline:- engineer applies theories, appropriate methods and tools to find the solution to problems within organizational & financial constraints
  - All aspect of software production:-   includes technical processes of software development, software project management, development of tools, Methods and theories to support software production.

Other terminologies

- **Software Product** is software systems delivered to the customer with documentation that describes how to install & use the system. In certain case, software product may be part of system products where hardware as well as software is delivered to customer

- **Software Crisis**: refers to set of problems characterized by an inability to develop software on time within budget and within requirements.

- **Software Process** is defined as the structured set of activities that are required to develop the software system.

Advantages of using Software Engineering for Developing software:

- Improve requirement specification
- Improved cost and scheduled estimates
- Improved Quality
- Better use of automated tools & techniques
- Less defects in final Product
- Better maintenance of software product
- Improved reliability & Productivity.

General Attributes of a good Software

There are various attributes associated with a software that decides whether software is good or bad.

- Maintainability: software must be written in a such a way that may evolve to meet the changing need of customers.

- Dependability: Software dependability has a range of sub-attributes like reliability through assured performance, security and safety. Dependable software should not cause physical or economic damage in the event of system failure.

- Efficiency: software should not make wasteful use of system resources such as RAM, CPU, etc

- Usability: Software must be usable. Software usability increases with good documentation & appropriate user interface.

Software characteristics

The nature & behavior of a software is quite different from other products like hardware.

- Software Does not Wear out: this is a very special characteristic of software. Unlike hardware , software product does not wear out. The intensity of software failure is very high in the initial phase of use and gradually becomes reliable over a time instead of wearing out. It may become obsolete/retired due to new requirements , environment changes etc.

- Software is flexible: Generally, software is considered to be flexible as program can be developed to do **almost anything** and allows accommodate any kind of change. However, it is not always true for all situations. Most of the times this characteristics has made software development difficult to plan, monitor & control.

Software characteristics

- Software is not manufactured : The software life begins from requirement analysis phase and ends up with the retirement of a software product. Software development is one time effort but requires a continuous maintenance effort in order to keep it operational. However, making 100 of copies does not involve any cost. In case of hardware product, every product costs us due to raw material & other processing expenses. Hence, it is not manufactured in classical sense.

- Reusability of components: Some components of the software product can be reuse for developing other new components.

Software Crisis

- The software crisis refers to a set of problems encountered in the development of software during 1960

- Due to absence of software engineering concept, software Industries failed their attempts to develop larger software systems and resulted the following problems:

  - Poor quality software was produced
  - Development teams exceeded the budget
  - Late delivery of software
  - User requirements are not completely met by the software
  - Unreliable software
  - Difficult to maintain software product.

- The problems increased because of increased in dependence of business on software and lack of a systematic approach to build the software

Some Reasons for Software Crisis

Software Crisis is characterized by an inability to develop software on time within budget and within requirements.

Following are the main reasons:-

- Lack of communication between software developers and users

- Increase in size of software

- Increase in cost of developing software

- Increase in complexity of the problem

- Project Management Problem

- Lack of understanding of the problem and its environment

- Efforts Duplication due to absence of automation in most of the development activities

- High optimistic estimates regarding the software development time & cost

Software Engineering came into existence as discipline to overcome these problems.

Software Applications

The Software application can be categorized as following:

1) System Software : is a collection of program written to service other programms. E.g : OS, compiler, Editors, Drivers, etc

2) Real –time Software: is a software that monitors/analyzes/controls the **real world events** as they occur. E.g GIS, Air line control System, etc

3) Business Software: is a software that facilitates the business operations or manages the decision making. E.g: Payroll, Inventory, Management Information System, etc

4) Engineering & Scientific software: e.g Automotive Manufacturing, Space Shutter Orbital, Computer Aided Design (like AutoCAD), System Simulation

Software Applications

5) Embedded Software: is a software that resides in Read-Only-Memory and is used to control the products & Systems for consumer & industrial markets.

6) Personal Computer Software: is a software that is used by one person at a time. E.g Word processing, Spread sheets, Multimedia, etc

7) Embedded Software: is a software that resides in Read-Only-Memory and is used to control the products & Systems for consumer & industrial markets.

8) Web based Software: e.g Browser

9) Artificial Intelligent Software: is software that sloves complex problems (not easy to analyze and compute) through learning mechanism

 e.g. Expert systems, Pattern recognition, Theorem Proving,

Game Playing, etc

Thank!!

See you in Next Class !!