Software is a program or set of programs containing instructions that provide desired functionality. And Engineering is the process of designing and building something that serves a particular purpose and finds a cost-effective solution to problems.

Program vs Software Product:

A program is a set of instructions that are given to a computer in order to achieve a specific task whereas software is when a program is made available for commercial business and is properly documented along with its licensing. Software = Program + documentation + licensing.

A program is one of the stages involved in the development of the software, whereas a software development usually follows a life cycle, which involves the feasibility study of the project, requirement gathering, development of a prototype, system design, coding, and testing.

Software components are parts of system or application. Components are a means of breaking the complexity of software into manageable parts. Each component hides the complexity of its implementation behind an interface.

This reduce the complexity of software development, maintenance, operations and support and allows the same code to be reused in many places.

Examples of component: Views, Models, Controllers, Data Access Objents, Services, Plugins, APIs.

Characteristics of Software:

Maintainability – It should be feasible for the software to evolve to meet changing requirements.

Efficiency – The software should not make wasteful use of computing devices such as memory, processor cycles, etc.

Correctness – A software product is correct if the different requirements as specified in the SRS document have been correctly implemented.

Reusability – A software product has good reusability if the different modules of the product can easily be reused to develop new products.

Testability – Here software facilitates both the establishment of test criteria and the evaluation of the software with respect to those criteria.

Reliability – It is an attribute of software quality. The extent to which a program can be expected to perform its desired function, over an arbitrary time period.

Portability – In this case, the software can be transferred from one computer system or environment to another.

Adaptability – In this case, the software allows differing system constraints and the user needs to be satisfied by making changes to the software.

Interoperability – Capability of 2 or more functional units to process data cooperatively.

Classification of Software

Software can be applied in countless fields such as business, education, social sector, and other fields. It is designed to suit some specific goals such as data processing, information sharing, communication, and so on. It is classified according to the range of potential of applications. These classifications are listed below.

• **System software:** This class of software manages and controls the internal operations of a computer system. It is a group of programs, which is responsible for using computer resources efficiently and effectively.

• **Real-time software:** This class of software observes, analyzes, and controls real world events as they occur. Generally, a real-time system guarantees a response to an external event within a specified period of time. Example: weather forecasting

• **Business software:** This class of software is widely used in areas where management and control of financial activities is of utmost importance. The fundamental component of a business system comprises payroll, inventory, and accounting software that permit the user to access relevant data from the database.

• **Engineering and scientific software:** This class of software has emerged as a powerful tool in the research and development of next generation technology. Applications such as the study of celestial bodies, under-surface activities, and programming of an orbital path for space shuttles are heavily dependent on engineering and scientific software.

• **Artificial intelligence (AI) software:** This class of software is used where the problem-solving technique is non-algorithmic in nature. The solutions of such problems are generally non-agreeable to computation or straightforward analysis. Instead, these problems require specific problem-solving strategies that include expert system, pattern recognition, and game-playing techniques.

• **Web-based software:** This class of software acts as an interface between the user and the Internet. The software incorporates executable instructions written in special scripting languages such as CGI or ASP. Apart from providing navigation on the Web, this

software also supports additional features that are useful while surfing the Internet.

• **Personal computer (PC) software:** This class of software is used for both official and personal use.This software is used predominantly in almost every field, whether it is database management system, financial accounting package, or multimedia-based software. It has emerged as a versatile tool for routine applications.

Generic views of software Engineering.

The process of a software development has three Generic views which are:

1. **Definition Phase** - It is the base of Definition phase. The experts get the knowledge about "What".
    - Information needed for processing.
    - Which functions are required.
    - Expectations about the capacity.
    - Interface which is established.
    - Area of the validation.

    This phase defines all the expectations depending on the standard of the software Engineering. It contains three steps.
    - Analysis of system
    - Planning of project
    - Requirement Analysis
2. **Development phase** - Focus point of development phase is "How". After the explanation of "What" it turn to "How". Various type of question raised in developer mind that how to design the data structure and Architecture of software,

Procedural detail how to implemented and how design convert in a programming language and testing of software how to perform. Three special steps always taken in this phase which are

- Design of software
- Coding
- testing of software system

3. **Maintenance phase** - The main focus of maintenance phase is change which cause is correction of errors, adaption of new idea, According to the needs of software after change in customer mood.

What is a software process?

A set of activities whose goal is the development or evolution of software

Generic activities in all software processes are:

• Specification - what the system should do and its development constraints

• Development - production of the software system

• Validation - checking that the software is what the customer wants

• Evolution - changing the software in response to changing demands

What is a software process model?

A simplified representation of a software process, presented from a specific

perspective

Examples of process perspectives are

- Workflow perspective - sequence of activities
- Data-flow perspective - information flow
- Role/action perspective - who does what

Generic process models

- Waterfall
- Evolutionary development
- Formal transformation
- Integration from reusable components