**Introduction to system Development(Software Engineering)**

**1.Why to study Software Engineering or System Development:**

* The main objective of the course is to acquaint with the processes and methodologies that is followed while developing large computer systems and software.
* It discusses systematic and cost effective techniques for software development and maintenance of developed systems.

**1.1 Real life analogy:**

* The need to study software engineering can be analysed with the help of a real life analogy. Suppose you are a owner of small grocery shop.Since your shop is small in size you don’t have much customer base also the complexity in the business will be very less.You also don’t require much planning and inventory to run the shop.Now, when you plan to expand your grocery shop to a multistore showroom then you have to think in various aspects.Such as how much more capital is required,who will be the new buyers,how to grow sales,how much inventory you have to manage,you much more persons you required etc.Similar situations happens with Software when your program size is small you can easily manage but as soon as more requirements come into the picture you need to use software engineering principals to develop and manage the system in cost effective manner.
* In my opinion software engineering is the collections of good and trusted methods, principals, processes that has been evolved over a period time.

**1.3 Outcome of Studying Software Engineering**

* Before studying software engineering principals most students have the notion that system development is only about writing codes but after studying software engineering fundamentals you get to know that coding is only a part of whole development process before coding there will be requirement analysis,Design and after coding you have to test and validate the system before it is fully handed to the users.

**2. Different type of Software systems:**

* **General application software:** These software are developed by the companies for general use .The company is the sole proprietor and company can only modify the software .Examples are spreadsheet softwares like excel,word etc.Here new versions are launched by the companies based on the general requirements. These software are designed for mass use.
* **User specific systems:** These systems are developed based on the specific user requirement. The user may be a group of people or a company or a person. Examples of such softwares are University enrolment system, price forecasting systems.One important point here is that the user decide the system specification and future changes.
* **Independent Systems:** These are the systems which is designed for desktop use.Examples of such as systems developed using MS Access database with VB script.
* **Batch systems:** Batch systems are designed to perform high volume repetitive tasks such report generation from thousand of transactions etc.Reformating or resizing of a number of files etc.Batch systems are most commonly used in Banking and finance industry.
* **Embedded systems:** It is a combination of software and hardware which is designed for specific usage.Daily example will be mobile phone,Smart TV,ATM etc.
* **Modelling and Simulation systems:** These systems are designed for specific scientific application or to model a particular situation.price forecasting systems.A common example of it is MATLAB which is widely used for many scientific modelling.
* **Data collection systems:** Systems which are deployed in capturing real time data such as weather data collection systems etc.Some big organisation also uses it for capturing live data and monitoring process.

Different type of systems are developed by using different programming languages and approaches.

**A software engineer can be posted at any of the project . The size of the project may vary from few lines of code to thousands of line. As the size of the project grows,cost involved will also increase which will increase the complexity to develop the software. Sound knowledge of system development fundamentals is the key to success.**

**3. How Software Engineering principles helps**

**3.1 Technical Aspects:**

* **Helpful in Capturing user requirements:** Software Engineering principles helps in capturing the user requirements more precisely. Many techniques such as axiomatic specification, algebraic specification have been developed to capture the user requirements precisely**.UML is also used for specifying requirements.**
* **Helps is creating good design documents:** With the use of software engineering principle we are able to generate good design which is modular and reusable.
* **Provide good Testing Techniques:**Software engineering principle provides various testing techniques such as White box and Black box testing to test the functional and non functional requirements of the software.
* **Provide framework for project planning and size estimation:** Software engineering principles provides techniques such as Halstead size estimation etc for estimation of project size and creation of SPMP (software project management plan) which is helpful in manpower allocation and cost estimation.

**Other important terms widely used;**

* **Modularisation-** Modularisation is a technique in which a big system is divided into smaller independent sub systems**.**Modularisation promotes reusability across different systems and also divison of work can be done easily.It also enables high cohesion.
* **Abstraction-** It is the process of hiding the unnecessary details from the users.Such as the user should not worry about the internal details such as data structure and algorithm used for developing the system. Abstraction make the system more usable.
* **Verification –** Verification is the process of making the system error free and the system should be able to execute the task without any error.
* **Validation-** Validation is the process in which it is tested the final system developed is as per the requirements of uses and all the functional and no functional requirements of the customer is met as per mentioned in the SRS(Software requirement specification document).