UNIT 1 : Software development process (12m)
Introduction to software engineering and process models.

* Computer software → collection of data
↓
tells computer how to work.

Q1 Software is:

① Instructions (computer programs) that when executed provide desired features, functions and performance
② Data structures that enables the program to adequately manipulate information
③ Descriptive information (documents) in both hard copy & virtual forms that describes the operation & use of the programs.

Q3 Characteristics of software:

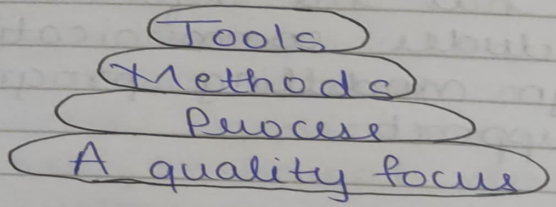
① Software is developed or engineered; it is not manufactured in the classical sense.

② Software doesn't "wear out." But it does deteriorate

③ Although the industry is moving is moving towards component-based construction, most software continues to be custom built.

Q6 Software engineering as layered technology approach:

Software engineering is a layered technology. The layers of software engineering as shown in the diagram below.



① A quality focus:
It defines the continous process improvement principles of software. It provides integrity that means providing security to the software so that data can be accessed by only an authorized person, no outside can access the data. It also focuses on maintainability and usability

② Process:
It is the foundation or base layer of software engineering. It is key that binds all the layers together which enables the development of software

before the deadline or on time. Process defines a framework that must be established for the effective delivery of software engineering technology. The software process covers all the activities, actions, and tasks required to be carried out for software development.

Process activities are:

communication planning modelling construction deployment

③ Method:
During the process of software development the answers to all "how-to-do" questions are given by method. It has the information of all the tasks which includes communication, requirement analysis, design modelling, program construction, testing, and support

Tools:
Software engineering tools provide a self-operating system for processes and methods. Tools are integrated which means information created by one tool can be used by another.

Types of Software

① System software

System software is a collection of programs written to service other programs. Some system software (eg. compilers, editor) processes complex, but determinate information structures. Other system applications (eg. operating system components, drivers) process largely ~~inter~~ indeterminate data.

② Application software

Application software consists of standalone programs that solve a particular / specific business need. Applications in this area process business or technical data in a way that facilities business operations or management / technical decision making.

③ Engineering / Scientific software

Formerly characterized by number crunching algorithms, engineering & scientific software applications range from automotive stress analysis to space shuttle orbital dynamics, & from molecular biology to automated manufacturing. eg. CAD / CAM software

④ Embedded software

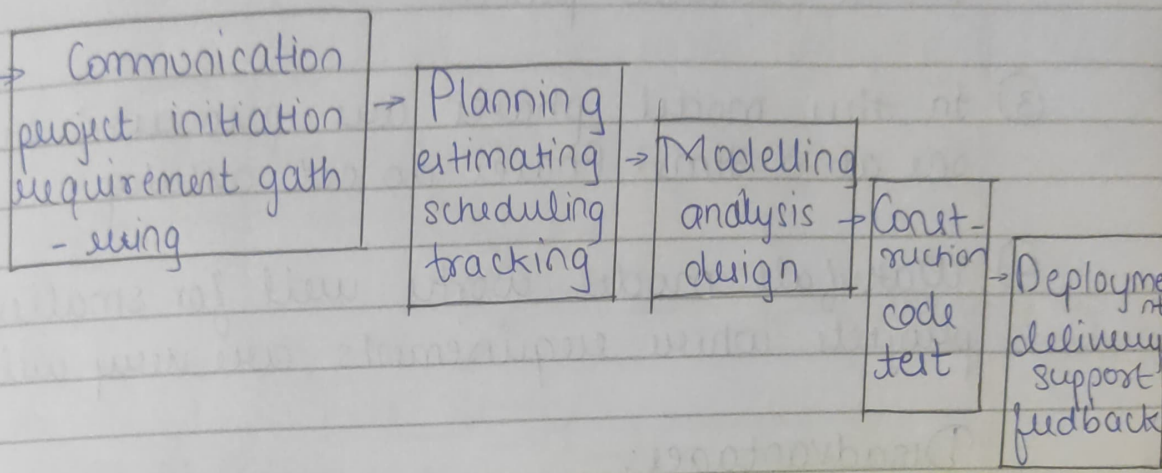
Embedded software resides within a product or system and is used to implement & control features & functions for the end-user & for the system itself. eg. Microwave, washing machine software.

Process Models

Prespective process model | specialized process models

Prescriptive process model

1. Waterfall process model



The waterfall model is a traditional method, sometimes called the classic life cycle. This is one of the initial models. Earlier this models was very popular but nowadays it is not used But it is very important because all the other software development life cycle models are based on the classical waterfall model.

The classical waterfall model divides the life cycle into a set of phases. This model considers that one phase can be started after the completion of the previous phase. That is the output of one phase will be the input of the next phase. Thus the development process can be considered as a sequential flow in the waterfall. Here the phases do not overlap with each other. The different sequential phases of the classical waterfall model as shown in above

Q11

Agile Process Model

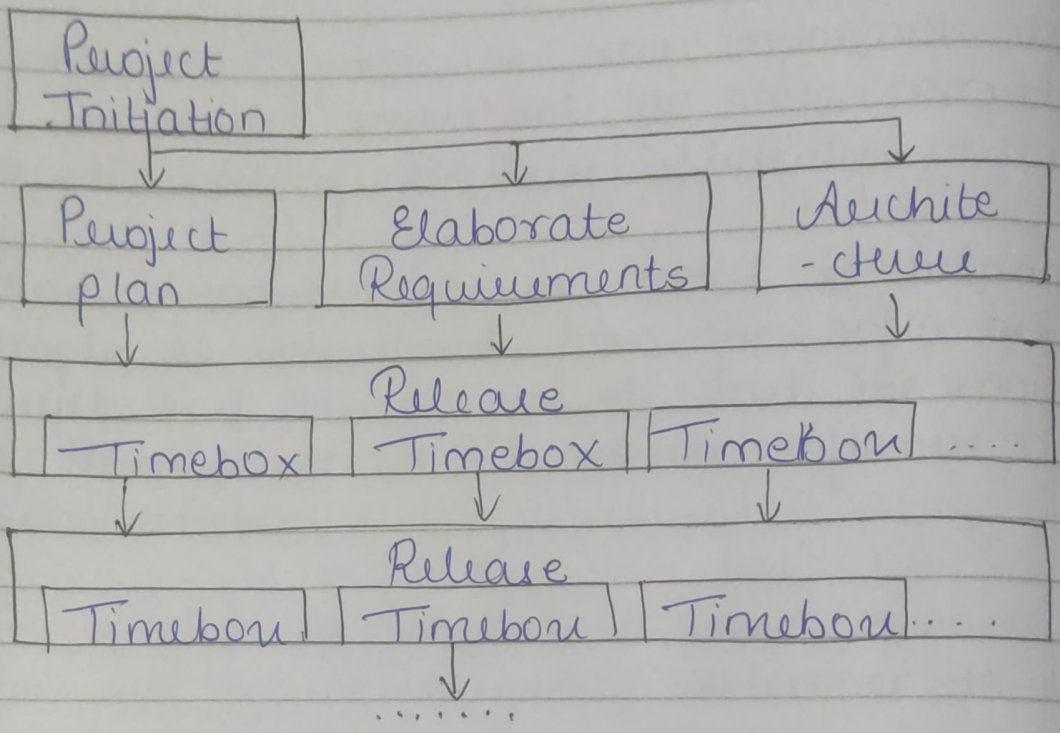
Agile software development is an approach to software development under which requirements and solutions evolve through the collaborative effort of self-organizing and and cross-functional teams and their customer(s)/end user(s).

It advocates adaptive planning, evolutionary development, early delivery, and continual improvement, and it encourages rapid and flexible response to change.

There is significant evidence that adopting agile practices and values improves the agility of software professionals, teams & organizations

Agile programming is an approach to project management, typically used in software development. It helps teams react to instability of building software through incremental, iterative work cycles known as sprints.

Generic agile life cycle.



1) Project Initiation: Set up and justify the proje
Determine initial high level requirement.

2) Project Plan: Plan out the rest of the project,
everything after project initiation

3) Elaborate requirements (optional): Gather the hig
level requirements during project initiation

4) Architecture: The agile approach is generally
design for today and refactor tomorrow, but th
agile community now generally accepts that some
upfront infrastructure work is necessary. This ma
setting up environments and testing frameworks,
plus doing the initial architecture.

5) Release: A release in this context is a piece of development ending in a public launch. Releases can be from 2 weeks to 6 months, but are usually about 3 months long. Releases have one or more time boxes.

6) Timebox: A timebox is 1-6 weeks long, but usually 3-4 weeks. The most important thing about a time box is that the delivery date is fixed.

Features of agile software development approach.

1) Iterative: Development process repeated many ti till the desired and result is achieved.

2) Flexible: Changes from clients are accommodat

3) Adaptable: Learn more to right than left f vice-versa.

4) Simple: Easily adopted by development teams

5) Transparent: Encourages client/user involvement

Attempt any THREE of the following:

Prescriptive process model and agile process model.

Prescriptive process model	**agile process mode**
Prescriptive process models stress detailed definition, identification, and application of process activates and tasks.	Agile process models emphasize project "agility" and follow a set of principles that lead to a more informal approach to software process.
A prescriptive model also describes how each of these elements are related to one another.	Agile methods note that not only do the software requirements change, but so do the team members, the technology being used.
It is Process oriented.	It is people oriented.
It follows Life cycle model (waterfall, spiral) development model.	It follows Iterative and Incremental development model.
Documentation required is to be comprehensive and constant.	Documentation required is to be minimal and evolving.
Predictive planning is required	Adaptive planning is required.
Customers role is important.	Customers role is critical.
Formal communication is required.	Informal communication is required.