# SE - Assignment

1. What is software? What is software engineering?

Software is a set of instructions, data or programs used to operate0020computers and execute specific tasks. It is the opposite of hardware, which describes the physical aspects of a computer. Software is a generic term used to refer to applications, scripts and programs that run on a device.

The two **main categories of software are application software and system software**. An application is software that fulfills a specific need or performs tasks. System software is designed to run a computer's hardware and provides a platform for applications to run on top of.

Other types of software include **programming software**, which provides the programming tools software developers need; **middleware software**, which sits between system software and applications; and **driver software**, which operates computer devices and peripherals.
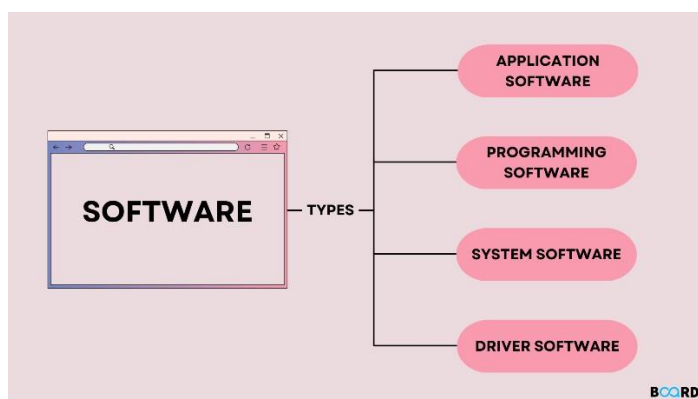

Software Engineering is the process of designing, developing, testing, and maintaining software. It is a systematic and disciplined approach to software development that aims to create high-quality, reliable, and maintainable software. Software engineering includes a variety of techniques, tools, and methodologies, including requirements analysis, design, testing, and maintenance.

-Main Attributes of Software Engineering

Software Engineering is a systematic, disciplined, quantifiable study and approach to the design, development, operation, and maintenance of a software system. There are Four main Attributes of Software Engineering.

• Efficiency
• Reliability
• Robustness
• Maintainability
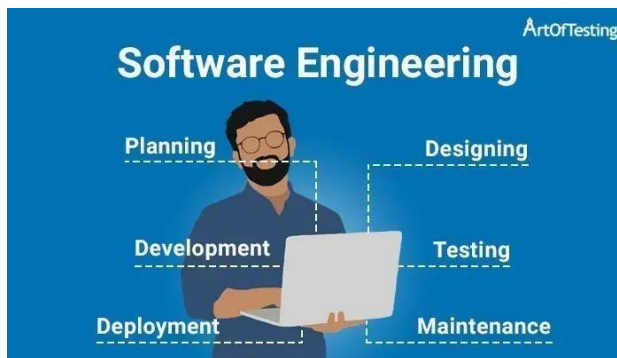
2. Explain types of software・Application software
The most common type of software, application software is a computer software package that performs a specific function for a user, or in some cases, for another application. An application can be self-contained, or it can be a group of programs that run the application for the user. Examples of modern applications include office suites, graphics software, databases and database management programs, web browsers, word processors, software development tools, image editors and communication platforms.

・System software
These software programs are designed to run a computer's application programs and hardware. System software coordinates the activities and functions of the hardware and software. In addition, it controls the operations of the computer hardware and provides an environment or platform for all the other types of software to work in. The OS is the best example of

system software; it manages all the other computer programs. Other examples of system software include the firmware, computer language translators and system utilities.

- **Driver software**
  Also known as device drivers, this software is often considered a type of system software. Device drivers control the devices and peripherals connected to a computer, enabling them to perform their specific tasks. Every device that is connected to a computer needs at least one device driver to function. Examples include software that comes with any nonstandard hardware, including special game controllers, as well as the software that enables standard hardware, such as USB storage devices, keyboards, headphones and printers.

- **Middleware**
  The term middleware describes software that mediates between application and system software or between two different kinds of application software. For example, middleware enables Microsoft Windows to talk to Excel and Word. It is also used to send a remote work request from an application in a computer that has one kind of OS, to an application in a computer with a different OS. It also enables newer applications to work with legacy ones.

- **Programming software**
  Computer programmers use programming software to write code. Programming software and programming tools enable developers to develop, write, test and debug other software programs. Examples of programming software include assemblers, compilers, debuggers and interpreters.

3. What is SDLC? Explain each phase of SDLC

SDLC is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.

The following figure is a graphical representation of the various stages of a typical SDLC.
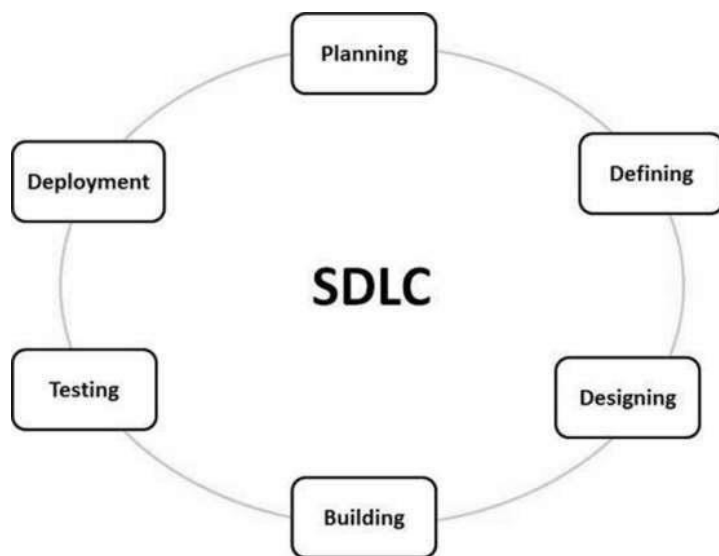
Stage1: Planning and requirement analysis

Requirement Analysis is the most important and necessary stage in SDLC.

The senior members of the team perform it with inputs from all the stakeholders and domain experts or SMEs in the industry.

Planning for the quality assurance requirements and identifications of the risks associated with the projects is also done at this stage.

Business analyst and Project organizer set up a meeting with the client to gather all the data like what the customer wants to build, who will be the end user, what is the objective of the product. Before creating a product, a core understanding or knowledge of the product is very necessary.

For Example, A client wants to have an application which concerns money transactions. In this method, the requirement has to be

precise like what kind of operations will be done, how it will be done, in which currency it will be done, etc.

Stage2: Defining Requirements

Once the requirement analysis is done, the next stage is to certainly represent and document the software requirements and get them accepted from the project stakeholders.

This is accomplished through "SRS"- Software Requirement Specification document which contains all the product requirements to be constructed and developed during the project life cycle.

Stage3: Designing the Software

The next phase is about to bring down all the knowledge of requirements, analysis, and design of the software project. This phase is the product of the last two, like inputs from the customer and requirement gathering.

Stage4: Developing the project

In this phase of SDLC, the actual development begins, and the programming is built. The implementation of design begins concerning writing code. Developers have to follow the coding guidelines described by their management and programming tools like compilers, interpreters, debuggers, etc. are used to develop and implement the code.

Stage5: Testing

After the code is generated, it is tested against the requirements to make sure that the products are solving the needs addressed and gathered during the requirements stage.

During this stage, unit testing, integration testing, system testing, acceptance testing are done.

Stage6: Deployment

Once the software is certified, and no bugs or errors are stated, then it is deployed.

Then based on the assessment, the software may be released as it is or with suggested enhancement in the object segment.

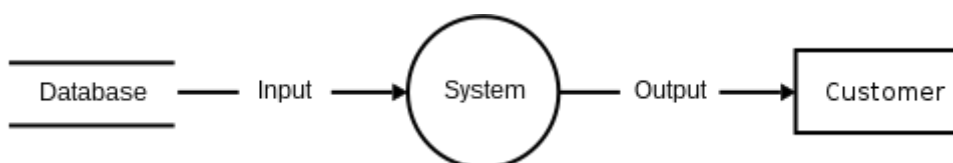After the software is deployed, then its maintenance begins.

Stage7: Maintenance

Once when the client starts using the developed systems, then the real issues come up and requirements to be solved from time to time.

This procedure where the care is taken for the developed product is known as maintenance.

4. What is DFD? Create a DFD diagram on Flipkart

A data-flow diagram is a way of representing a flow of data through a process or a system (usually an information system). The DFD also provides information about the outputs and inputs of each entity and the process itself.

DFD OF FLIPKART

-0 level DFD

Customer Management

Shopping Management

Payment Management

Online Shopping

Order Management

System user Management

Login Management

-1 level DFD

Customer info

Order status

Bill and delivery info

1.1

Managing/purchasing/order information

List of orders

Order details

Product category

Order info

Customer
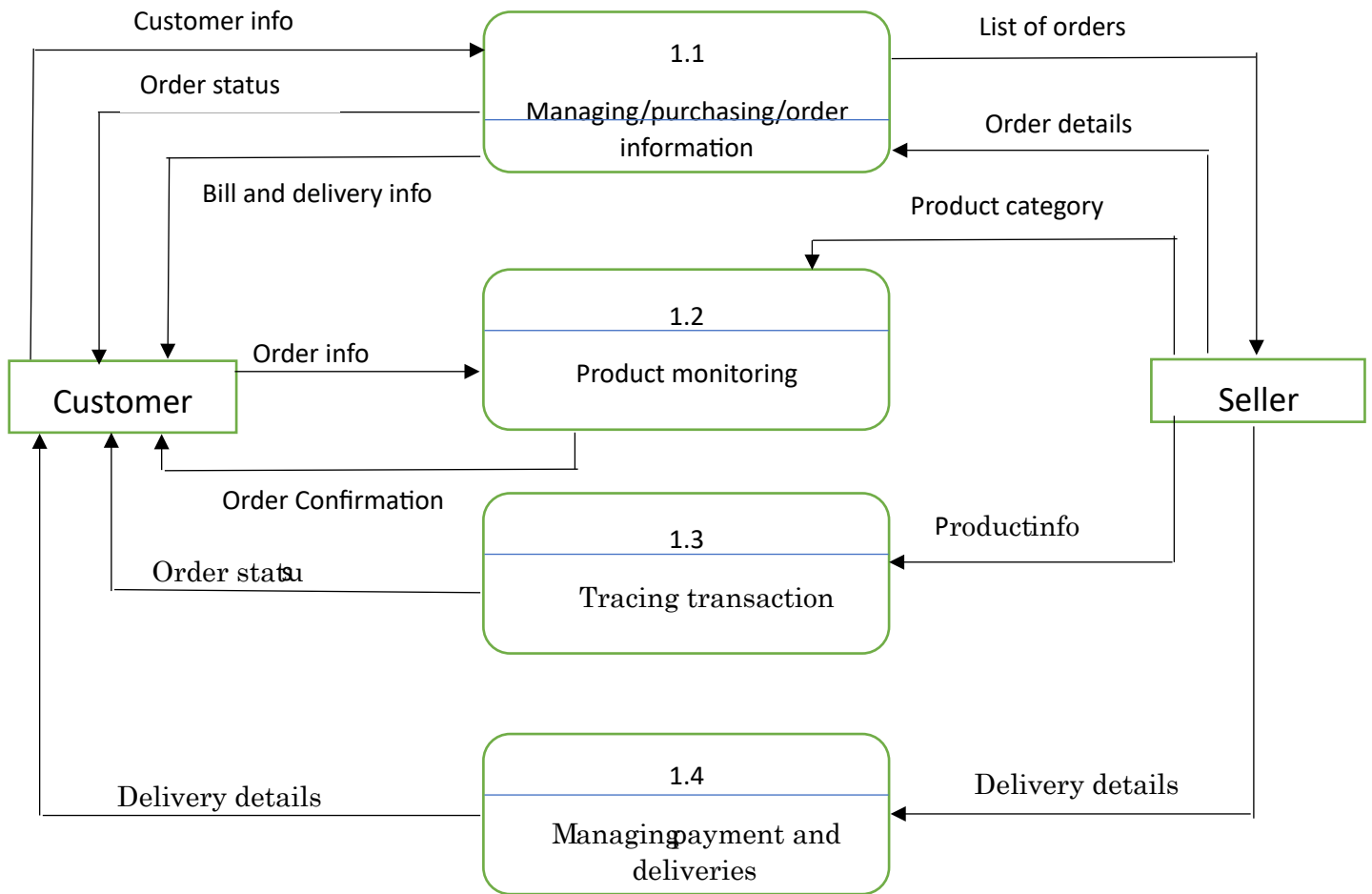
1.2

Product monitoring

Seller

Order Confirmation

Order status

1.3

Tracing transaction

Productinfo

Delivery details

1.4

Managing payment and deliveries
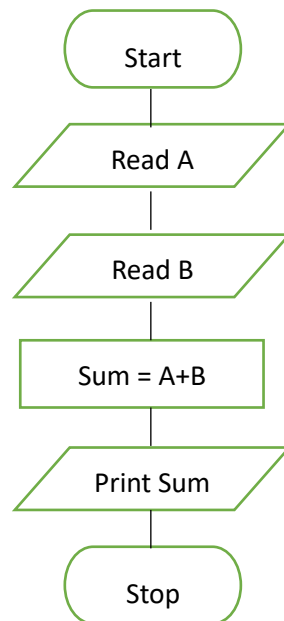
Delivery details

5. What is Flow chart? Create a flowchart to make addition of two numbers

A flowchart is a type of diagram that represents a workflow or process. A flowchart can also be defined as a diagrammatic representation of an algorithm, a step-by-step approach to solving a task.

Flowchart of Addition Of Two Numbers

```
        ┌─────────────┐
       (    Start      )
        └──────┬──────┘
         ╱──────────────╲
        ╱     Read A      ╲
        ╲────────┬────────╱
         ╱──────────────╲
        ╱     Read B      ╲
        ╲────────┬────────╱
        ┌─────────────────┐
        │    Sum = A+B     │
        └────────┬────────┘
         ╱──────────────╲
        ╱    Print Sum    ╲
        ╲────────┬────────╱
        ┌─────────────────┐
       (      Stop         )
        └─────────────────┘
```

6. What is Use case Diagram? Create a use-case on bill payment on paytm.

A use case is a list of actions or event steps typically defining the interactions between a role (known in the Unified Modelling Language (UML) as an actor) and a system to achieve a goal. The actor can be a human or another external system.

Visitor

Visit

Register

Update

Verify

Add user

Admin

Registered
User

Login

Transact
ions

View
Report

Logout

Transaction
Report

Service
Provider