

## ❖ What is software? What is Software Engineering?

- a. Software is a set of instructions, data, or programs used to operate a computer and execute specific tasks. In simpler terms, software tells a computer how to function. It's a generic term used to refer to applications, scripts, and programs that run on devices such as PCs, mobile phones, tablets, and other smart devices. Software contrasts with hardware, which is the physical aspects of a computer that perform the work.
- b. Software engineering is defined as a process of analyzing user requirements and then designing, building, and testing software application which will satisfy those requirements.
- c. IEEE, in its standard 610.12-1990, defines software engineering as the application of a systematic, disciplined, which is a computable approach for the development, operation, and maintenance of software.
- d. Fritz Bauer defined it as 'the establishment and used standard engineering principles. It helps you to obtain, economically, software which is reliable and works efficiently on the real machines'.

## ❖ Explain types of Software?

### ○ **1. System Software:**

- A system software aids the user and the hardware to function and interact with each other. Basically, it is a software to manage computer hardware behavior so as to provide basic functionalities that are required by the user. In simple words, we can say that system software is an intermediary or a middle layer between the user and the hardware. These computer software sanction a platform or environment for the other software to work in. This is the reason why system software is very important in managing the entire computer system. When you first turn on the computer, it is the system software that gets initialized and gets loaded in the memory of the system. The system software runs in the background and is not used by the endusers. This is the reason why system

- Some examples of Operating systems given below:

- Android
- CentOS
- iOS
- Linux
- Mac OS
- MS Windows
- Ubuntu

**2.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

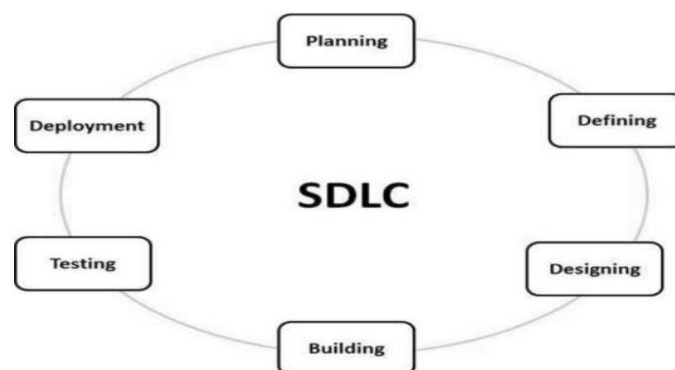
some cases, for another application. An application can be self-contained, or it can be a group of programs that run the application for the user.

- **3. 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.
- **4. 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.
- **5. 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.

➤ **What is SDLC? Explain each phase of SDLC?**

**Ans:**

- 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.



○

#### ❖ **Stage 1: Planning**

- Requirement analysis is the most important and fundamental stage in SDLC. It is performed by the senior members of the team with inputs from the customer, the sales department, market surveys and domain experts in the industry. This information is then used to plan the basic project approach and to conduct product feasibility study in the economical, operational and technical areas.

#### **Stage 2: desing**

In this phase, the requirement gathered in the SRS document is used as an input and software architecture that is used for implementing system development is derived.

#### **Stage3: Implementation**

Implementation/Coding starts once the developer gets the Design document. The Software design is translated into source code. All the components of the software are implemented in this phase.

#### **Stage4Testing**

Testing starts once the coding is complete and the modules are released for testing. In this phase, the developed software is tested thoroughly and any defects found are assigned to developers to get them fixed.

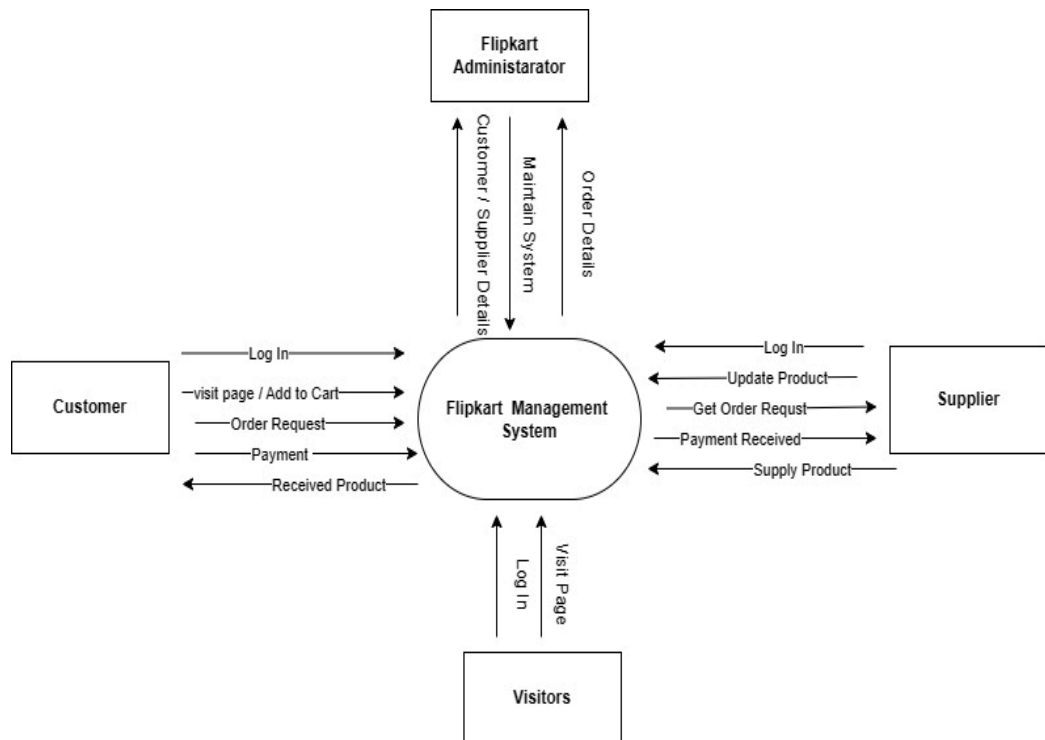
Retesting, regression testing is done until the point at which the software is as per the customer's expectation. Testers refer SRS document to make sure that the software is as per the customer's standard.

### **5) Deployment**

Once the product is tested, it is deployed in the production environment or first UAT (User Acceptance testing) is done depending on the customer expectation.

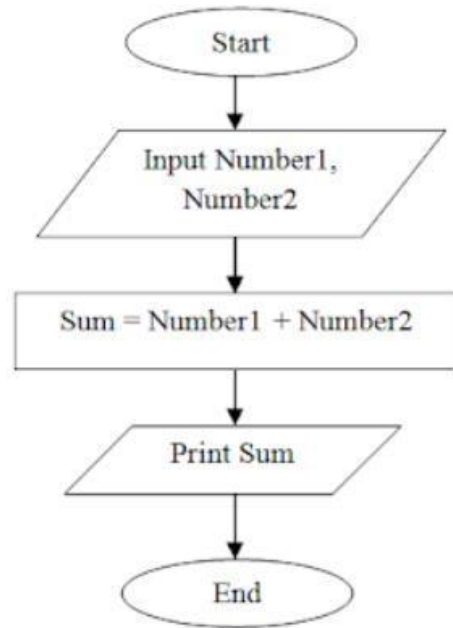
#### ❖ **What is DFD?Create a DFD diagram on Flipkart?**

- DFD is the abbreviation for Data Flow Diagram. The flow of data of a system or a process is represented by DFD. It also gives insight into the inputs and outputs of each entity and the process itself. DFD does not have control flow and no loops or decision rules are present. Specific operations depending on the type of data can be explained by a flowchart. Data Flow Diagram can be represented in several ways. The DFD belongs to structured-analysis modeling tools. Data Flow diagrams are very popular because they help us to visualize the major steps and data involved in software-system processes.



❖ **What is flow chart? Create a flowchart to make addition of two numbers?**

- A flowchart is a diagram that depicts a process, system or computer algorithm. They are widely used in multiple fields to document, study, plan, improve and communicate often complex processes in clear, easy-to-understand diagrams. Flowcharts, sometimes spelled as flow charts, use rectangles, ovals, diamonds and potentially numerous other shapes to define the type of step, along with connecting arrows to define flow and sequence. They can range from simple, hand-drawn charts to comprehensive computer-drawn diagrams depicting multiple steps and routes. If we consider all the various forms of flowcharts, they are one of the most common diagrams on the planet, used by both technical and non-technical people in numerous fields. Flowcharts are sometimes called by more specialized names such as Process Flowchart, Process Map, Functional Flowchart, Business Process Mapping, Business Process Modeling and Notation (BPMN), or Process Flow Diagram (PFD). They are related to other popular diagrams, such as Data Flow Diagrams (DFDs) and Unified Modeling Language (UML) Activity Diagrams.



❖ **What is use case Diagram? Create a use case on bill payment on paytm?**

- ❖ A use case diagram is usually simple. It does not show the detail of the use cases:
- ❖ It only summarizes some of the relationships between use cases, actors, and systems.
- ❖ It does not show the order in which steps are performed to achieve the goals of each use case.
- ❖ As said, a use case diagram should be simple and contains only a few shapes. If yours contain more than 20 use cases, you are probably misusing use case diagram.

