1. What is software? What is software engineering?

-Software is set of instruction, data, programs used to operate computer and execute specific task.It is opposite of hardware which describe the physical aspects of computer.

-The two main categories of software are application software and system software

Software engineering is the branch of computer science that deals with the design, development, testing, and maintenance of software applications. Software engineers apply engineering principles and knowledge of programming languages to build software solutions for end users.

2. Explain types of software.

*-* Software is a set of programs (sequence of instructions) that allows the users to perform a well-defined function or some specified task.

Software is responsible for directing all computer-related devices and instructing them regarding what and how the task is to be performed. However, the software is made up of binary language (composed of ones and zeros). There are various types of software, each serving different purposes and functions. Here are some common categories of software:

**TYPES OF SOFTWARE :**

* Application Software
* System Software
* Driver Software
* Middleware
* Programming Software

**Application Software:**

Application software refers to programs and tools designed to perform specific tasks for end-users. Examples include word processors, web browsers, and graphic design software.

**System Software:**

System software is a type of computer program that provides a platform for running application software and manages hardware resources. Operating systems and utilities are examples of system software.

**Driver Software:**

Driver software, or drivers, are specialized programs that facilitate communication between an operating system and hardware devices. They ensure proper functioning and compatibility between software and hardware components.

**Middleware:**

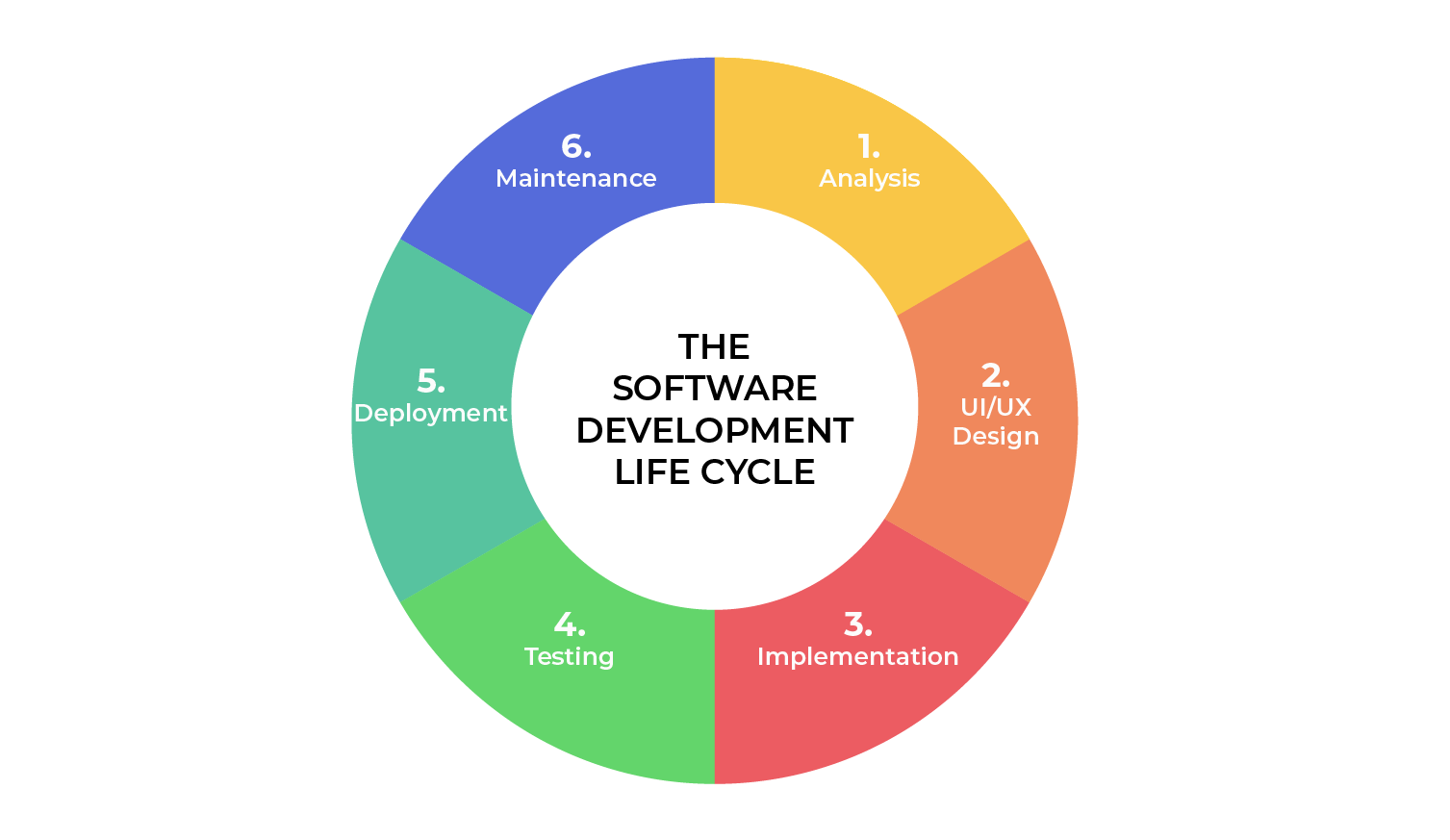
Middleware is software that acts as an intermediary layer between different software applications, facilitating communication and integration. It often enables interoperability between diverse systems and applications.

**Programming Software:**

Programming software includes tools and applications used by developers to create, debug, and maintain software code. Integrated Development Environments (IDEs), compilers, and debuggers are examples of programming software.

3. What is SDLC? Explain each phase of SDLC.

SDLC stands for Software Development Life Cycle. It is a systematic process used by software developers to design, develop, test, and deploy software applications. The SDLC encompasses a series of well-defined phases, each with its own set of activities and goals. The typical phases of SDLC include:



* Analysis
* Designing
* Implementation
* Testing
* Deployment
* Maintenance

**Analysis:**

Purpose: Understand and document user requirements to define the software's scope.

Activities: Gather information through interviews, surveys, and studying existing systems to create a comprehensive understanding of the project's needs.

**Designing:**

Purpose: Develop a structured plan and blueprint for the software based on the analysis.

Activities: Create high-level architecture and detailed design specifications, including user interfaces and system components.

**Implementation:**

Purpose: Transform the design into actual code to build the software.

Activities: Write, code, and integrate components, translating the design specifications into a functioning application.

**Testing:**

Purpose: Ensure the software functions correctly and meets specified requirements.

Activities: Conduct thorough testing, including unit, integration, and system testing, to identify and address any defects.

**Deployment:**

Purpose: Release the software for use in a live environment.

Activities: Install and configure the software, making it accessible to end-users while ensuring compatibility and stability.

**Maintenance:**

Purpose: Address post-deployment issues, enhance features, and ensure ongoing system performance.

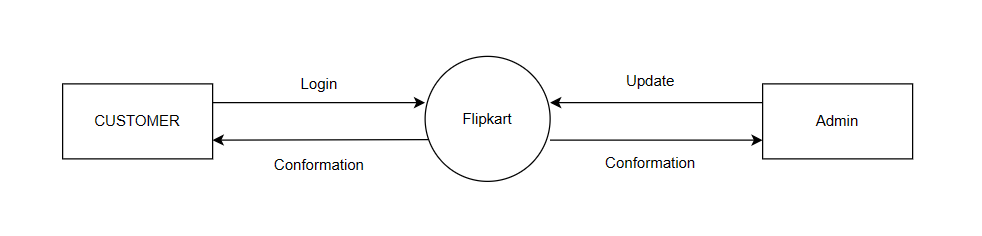
Activities: Fix bugs, implement updates, and make improvements based on user feedback, maintaining the software's effectiveness over time.

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

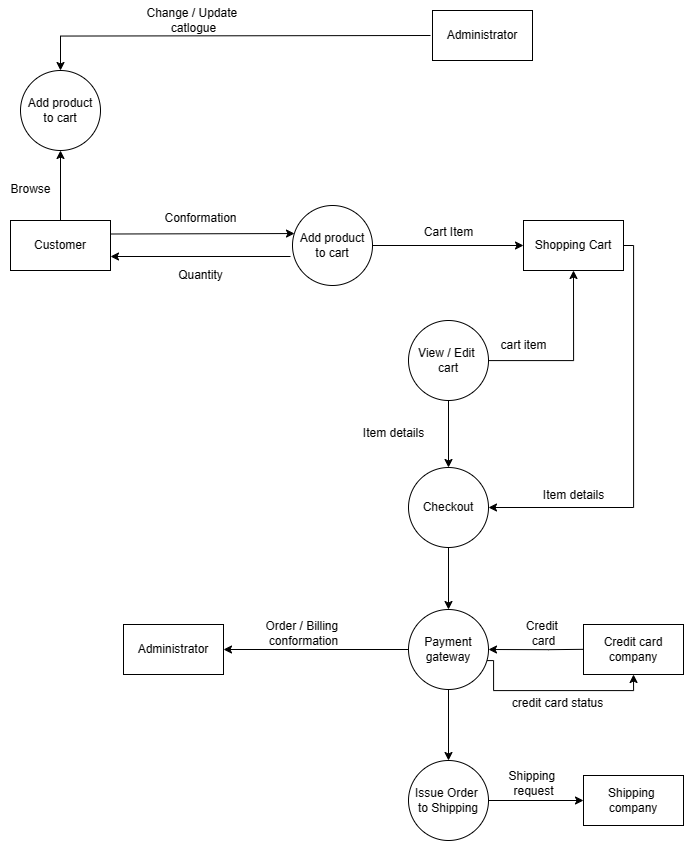
A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It can be manual, automated, or a combination of both.

It shows how data enters and leaves the system, what changes the information, and where data is stored.

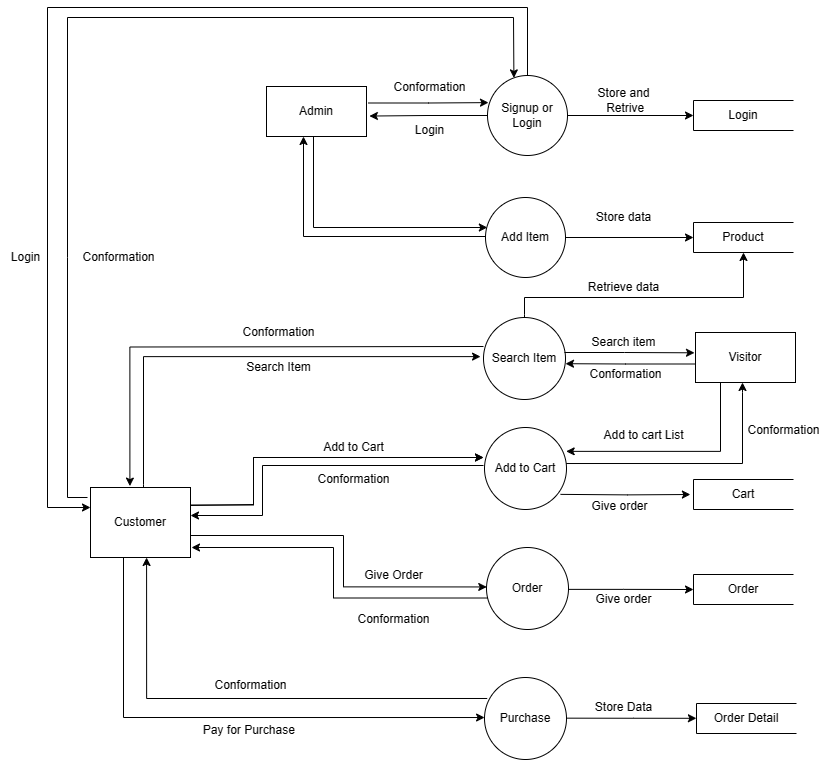
The objective of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communication tool between a system analyst and any person who plays a part in the order that acts as a starting point for redesigning a system.



**0** **LEVEL DFD OF FLIPKART**

****

**1** **LEVEL DFD OF FLIPKART**

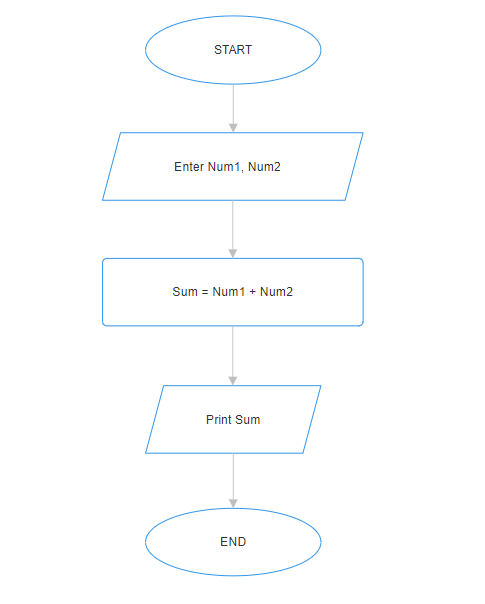
****

**2** **LEVEL DFD OF FLIPKART**

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

A flowchart is a visual representation of a process or system, using shapes and arrows to illustrate the sequence of steps and decision points. It provides a simple and clear way to understand, document, and communicate the flow of activities in a workflow or algorithm. Flowcharts are widely used in various fields for process analysis, problem-solving, and procedural documentation.

**Flowchart to make addition of two numbers**

****

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

A Use Case Diagram is a visual representation that shows how users or external systems interact with a software system. It uses actors (representing users or entities) and use cases (depicting system functionalities) to illustrate the system's behavior and its interactions with external elements. This diagram helps in understanding the functionality from a user's perspective and is commonly used in software development for requirements analysis and design.

