MODULE: 1

SE – Overview of IT Industry

1) What is software? What is software engineering?

Answer: Software is a set of instructions, data or programs used to operate computers 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.

=>software engineering :

Software engineering is an engineering approach to software development. A practitioner, called a software engineer, applies the engineering design process to develop software. The terms programmer and coder overlap software engineer, but they imply only the construction aspect of typical software engineer workload.

2) Explain types of software

Answer: **1. System software**

2. Utility software

3. Application software

1. System software

If you think of software as being in layers, the system software is the bottom layer: it sits between the hardware and the application software.

Operating systems like Windows, macOS, Android and iOS are examples of system software. Operating systems are loaded into RAM when the device starts up, and have access to the hard drive.

2. Utility software

Utility software is part of the system software and performs specific tasks to keep the computer running. Utility software is always running in the background. Examples of utility software are security and optimisation programs.

Security programs include anti-virus software that scans and removes viruses. Most computers will include some sort of anti-virus software, but you can add your own.

3. Application software

This is everything else! Anything that is not an operating system or a utility is an application or app. So a word processor, spreadsheet, web browser, and graphics software are all examples of application software, and they can do many specific tasks.

You can remove and add applications on your computer using the operating system.

Application software like a word processor regularly directs the operating system to load and save files from and to the hard drive. When you are working on a file, it is saved temporarily in the RAM. It is only when you choose to save it that it is written to the hard drive

3) What is SDLC? Explain each phase of SDLC

Answer: SDLC stands for Software Development Life Cycle. It is a structured approach to software development that outlines the processes and stages involved in building software. The SDLC aims to produce high-quality software that meets customer expectations, is completed on time and within budget, and complies with regulations.

The phases of SDLC typically include:

1. **Requirements Gathering and Analysis:**

   Purpose: Understand and document the software requirements from stakeholders (clients, users, etc.).

   Activities: Gather information, analyze requirements for feasibility and clarity, and document detailed requirements in a Software Requirements Specification (SRS) document.

   Output: SRS document, which serves as a blueprint for the entire project.

2. **System Design:**

   Purpose: Design the architecture of the software based on the requirements gathered.

   Activities: Design the overall system architecture, database schema, user interface, modules, and other system components.

Output: Design documents (e.g., High-Level Design (HLD) and Low-Level Design (LLD)) that guide the development team in implementing the system.

3. **Implementation (Coding):**

Purpose: Translate the design into actual code.

Activities: Writing code according to the design specifications, integrating different modules, and ensuring the code is well-documented and maintainable.

Output: Source code, executables, and other related documentation.

4. **Testing:**

Purpose: Verify that the software works as expected and meets the defined requirements.

Activities: Develop and execute test cases, report defects, and validate fixes.

Types of Testing: Includes unit testing, integration testing, system testing, acceptance testing, and sometimes performance testing and security testing.

Output: Test reports, defect logs, and a verified product ready for deployment.

5. **Deployment:**

Purpose: Release the software into the production environment.

Activities: Plan the deployment process, install the software, configure it according to requirements, and ensure all necessary components are in place.

Output: Deployed software that is ready for use

6. **Maintenance and Support:**

Purpose: Ensure the software continues to operate correctly and efficiently over time.

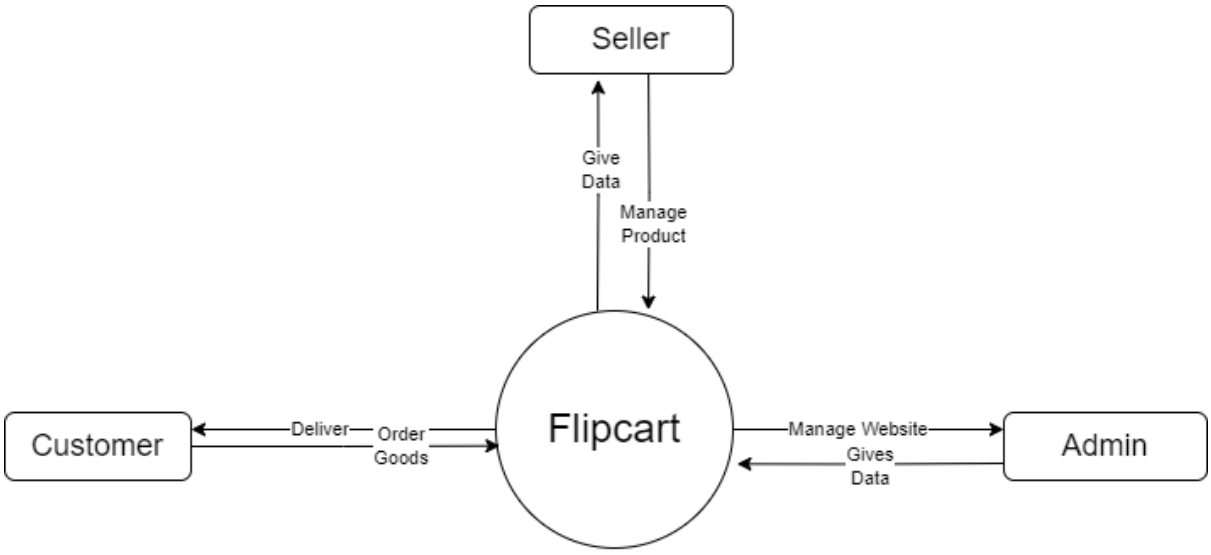
Activities: Address defects (bugs) discovered post-deployment, make improvements or enhancements as needed, and provide user support.

Output: Updates, patches, and ongoing support documentation.

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

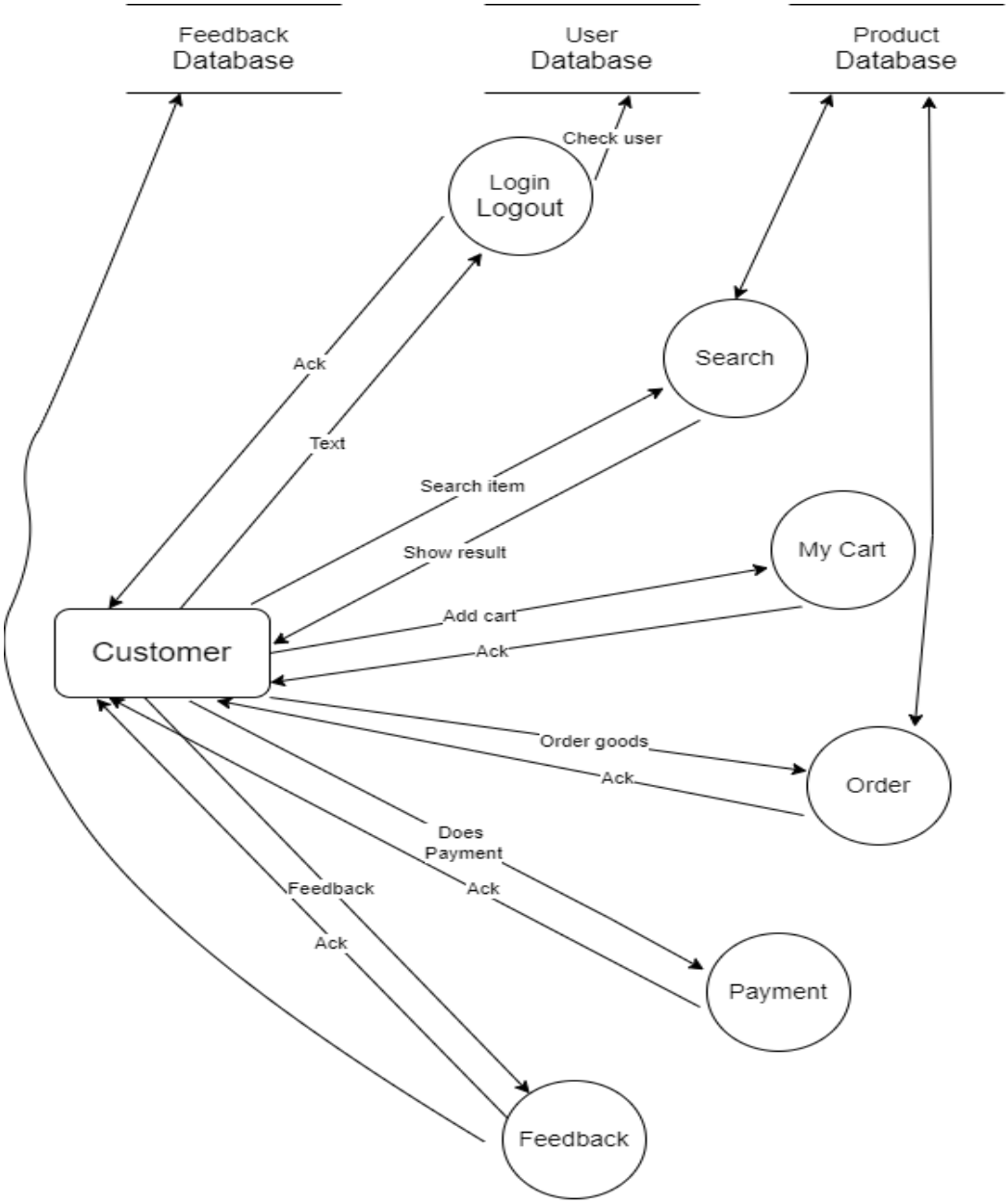
Answer: A data flow diagram (DFD) is a graphical or visual representation using a standardized set of symbols and notations to describe a business's operations through data movement. They are often elements of a formal methodology such as Structured Systems Analysis and Design Method (SSADM).

DFD Level = 0



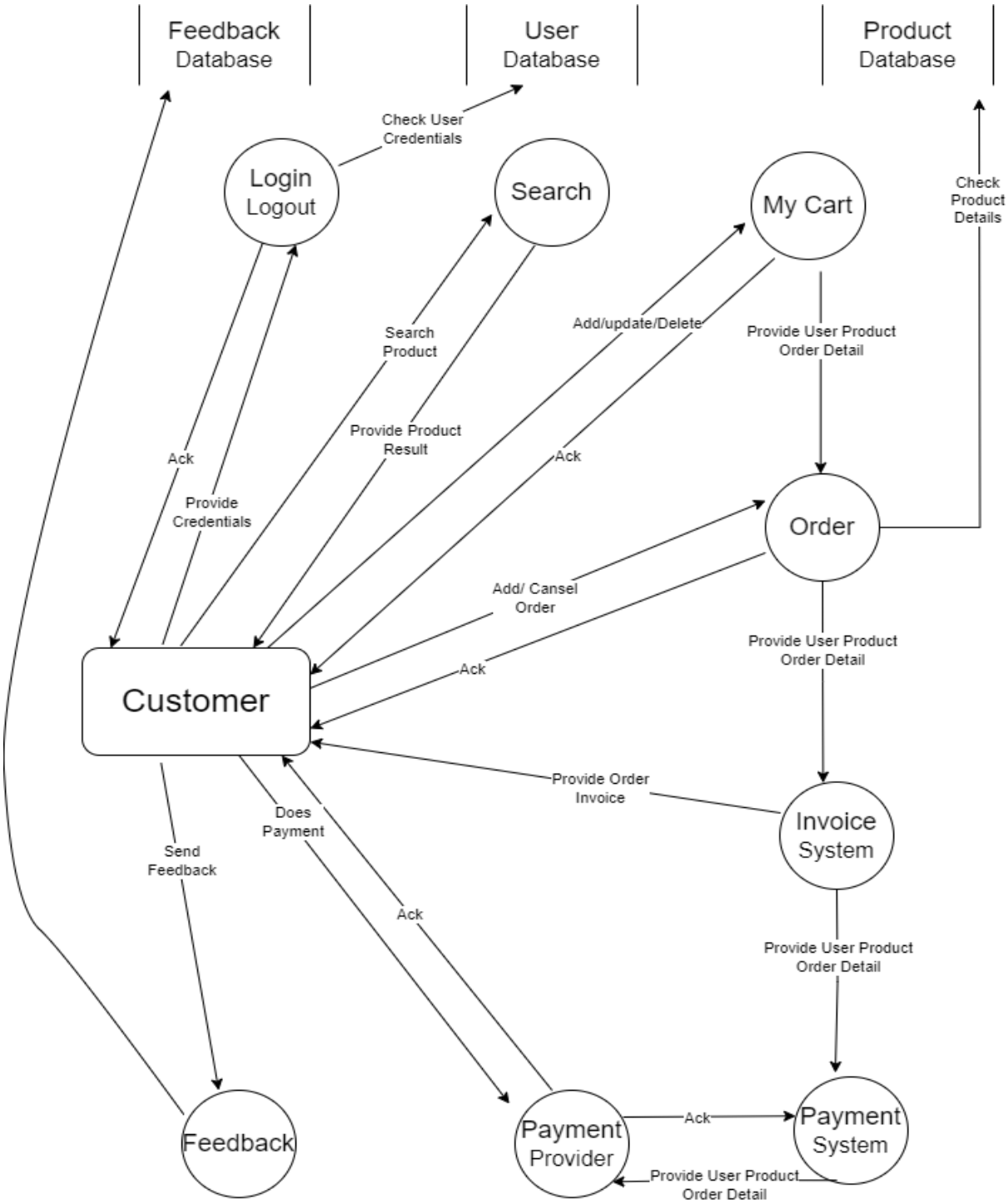
DED Level 0

DFD Level = 1



DED Leval 1

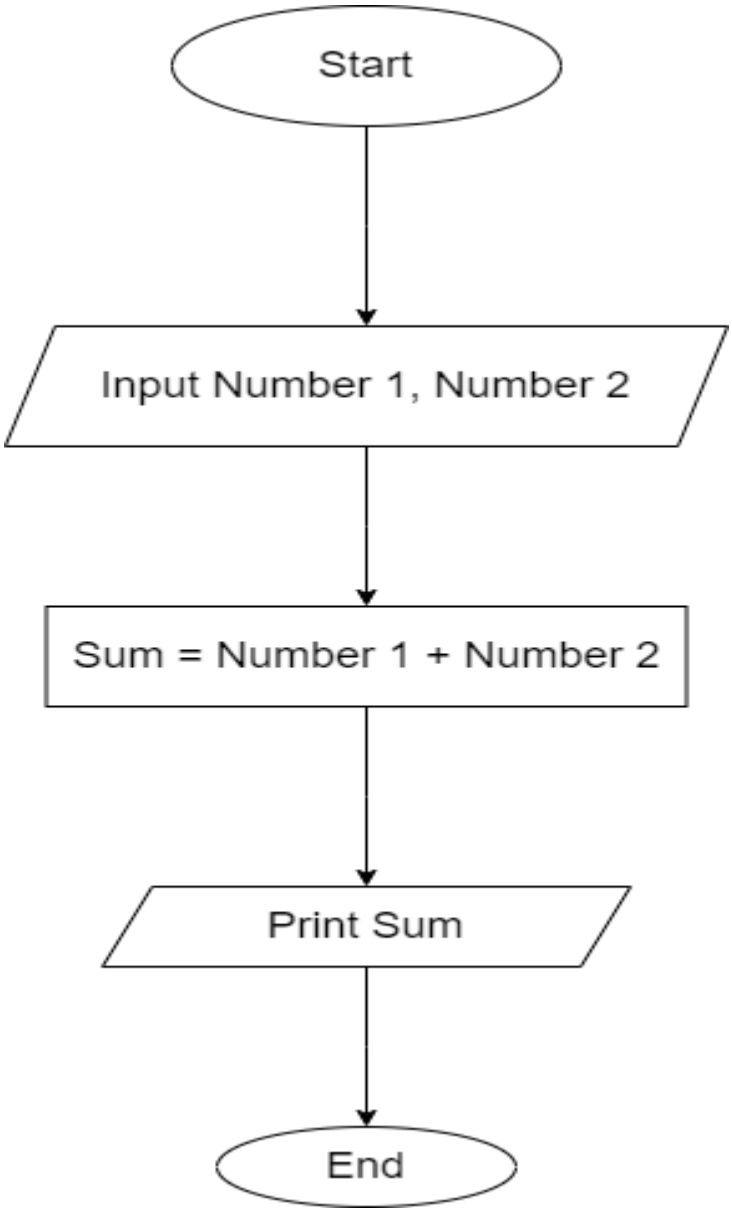
DFD Level = 2



DFD Level 2

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

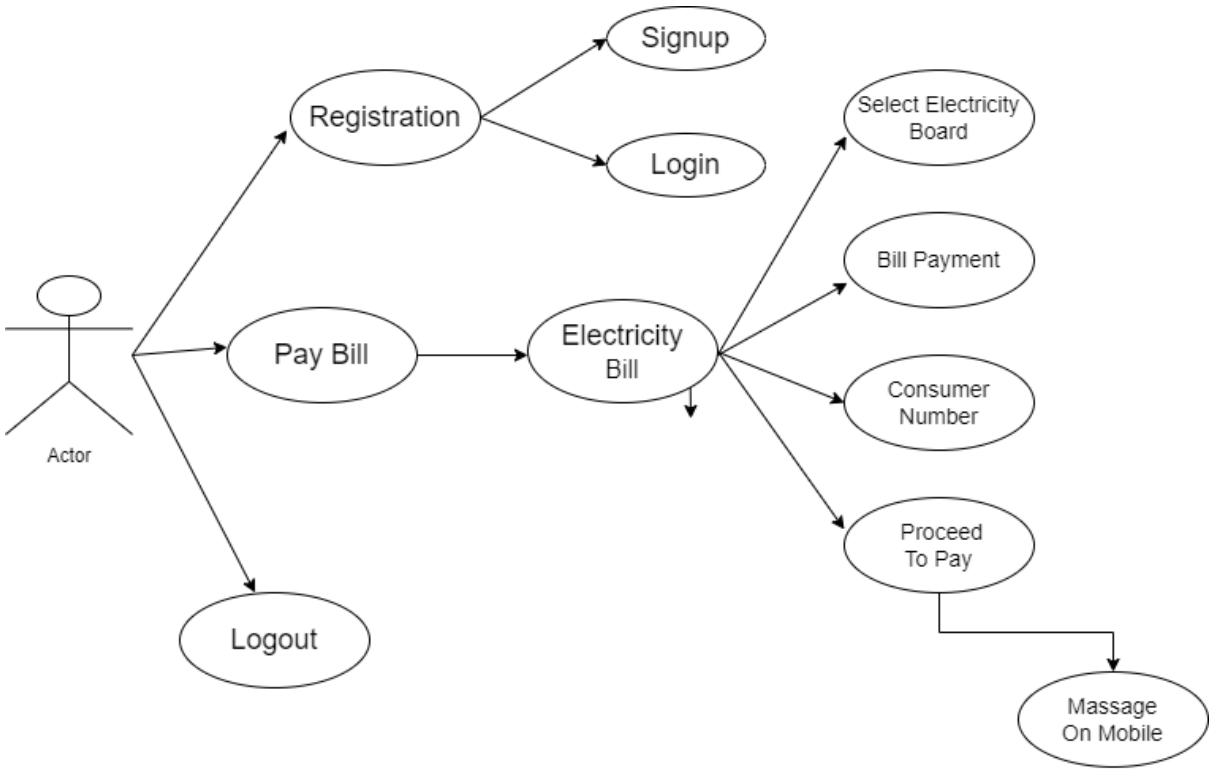
Answer: A flowchart is a diagram that depicts a process, system or computer algorithm. They are widely used in multiple fields to document, study, plan, improve and communicate often complex processes in clear, easy-to-understand diagrams.



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

Answer: Use-case diagrams describe the high-level functions and scope of a system. These diagrams also identify the interactions between the system and its actors. The use cases and actors in use-case diagrams describe what the system does and how the actors use it, but not how the system operates internally.

Use Case Diagram



Use Case Diagram