**Assignment-1**

Q1> What is Software? What is software engineering?

Ans-> **What is software?**

* **Software** is a collection of data, instructions, and programs that tell a computer how to work and perform specific tasks.

**Main Aspects of Software:**

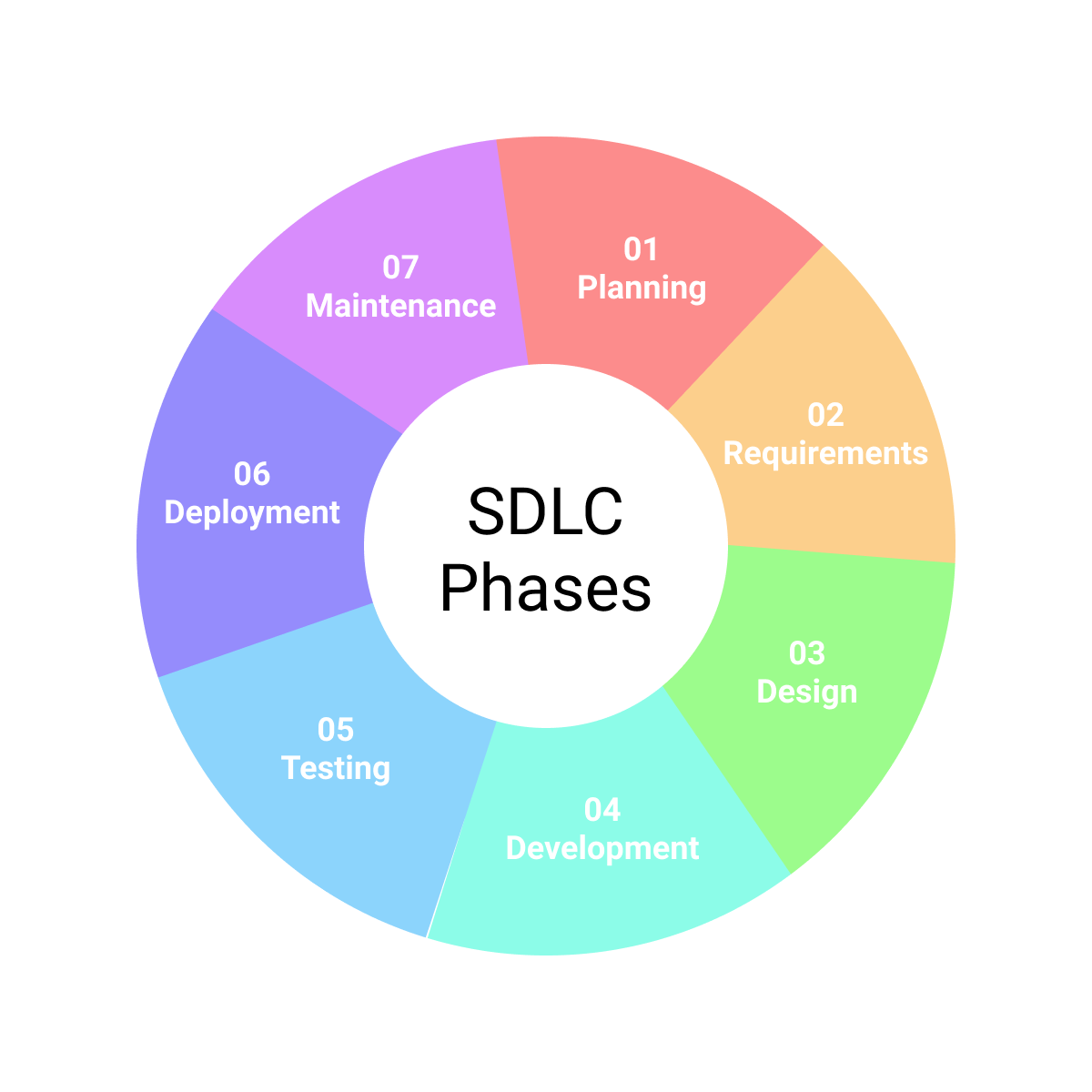
* **Instructions**: Detailed commands that tell the computer what actions to perform.
* **Data**: Information that is processed by the instructions.
* **Programs**: Collections of instructions and data that work together to achieve a specific outcome or perform a specific task.
* **What is software engineering**?
* **Software Engineering** is the systematic application of engineering principles to the design, development, testing, maintenance, and management of software systems.
* Software engineering aims to produce high-quality software in a cost-effective, predictable, and timely manner.
* **Key Aspects of Software Engineering:**
* **Requirements Analysis:** Identifying and documenting the needs and constraints of the users and stakeholders.
* **Design:** Creating a blueprint for the software architecture, components, interfaces, and data flow.
* **Implementation (Coding):** Writing the actual code based on the design specifications.
* **Testing:** Verifying that the software works as intended. Identifying and fixing defects or bugs.
* **Maintenance:** Updating and improving the software after its initial release.
* **Project Management:** Planning, organizing, and managing resources to bring about the successful completion of specific project goals and objectives.

Q2> Explain types of software?

* System Software
* Application Software
* Driver software
* Middleware
* Development Software
* **System Software**: This includes the operating system and all the utilities that enable the computer to function. Examples are Windows, macOS, Linux, and Android.
* **Application Software**: These are programs designed to perform specific tasks for users, such as word processors, spreadsheets, and media players. Examples include Microsoft Office, Adobe Photoshop, and web browsers.
* **Driver software.**Also known as device drivers, this software is often considered a type of system software.
* **Middleware**: Software that connects different applications or services, enabling them to communicate and work together. Examples include database management systems and web servers.
* **Development Software**: Tools and applications used by developers to create, debug, maintain, or otherwise support software development and maintenance. Examples include compilers, debuggers, and integrated development environments (IDEs).

Q3> What is SDLC? Explain each phase of SDLC.

* **Software Development Life Cycle (SDLC)** is a systematic process for planning, creating, testing, and deploying an information system.



* 1. **Planning:** Establish the scope, purpose, and feasibility of the project**.**
  2. **Requirements Analysis:** Gather and document detailed functional and non-functional requirements.
  3. **Design:** Transform requirements into a blueprint for building the software.
  4. **Implementation (Coding):** Translate design specifications into executable software.
  5. **Testing:** Ensure that the software functions correctly and meets requirements.
  6. **Deployment:** Release the software to the production environment and make it available to end-users.
  7. **Maintenance:** Provide ongoing support, fix issues, and enhance the software post-deployment.

Q4>What is DFD? Create a DFD diagram on Flipkart

* A Data Flow Diagram (DFD) is a graphical representation used to visualize the flow of data within a system. It illustrates how data moves from one process to another, as well as how data is stored and accessed. DFDs are commonly used in systems analysis and design to understand and document the processes and data structures of an existing or proposed system.
* Levels of DFD:
* Context Diagram (Level 0 DFD):

 Provides a high-level overview of the system.

 Shows the system as a single process and its interaction with external entities.

 Does not include detailed processes or data stores within the system.

* Level 1 DFD:

 Decomposes the single process in the context diagram into sub-processes.

 Provides more detail about the data flows and data stores within the system.

 Shows the main functions or processes within the system.

* Level 2 (and lower) DFDs:

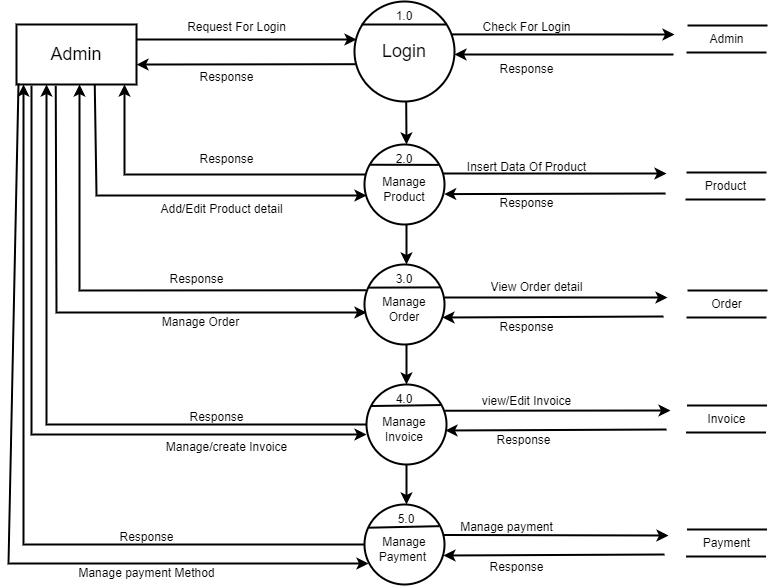
 Further decomposes the processes from Level 1 into more detailed sub-processes.

 Used for complex systems where more detail is required to understand the data flow.

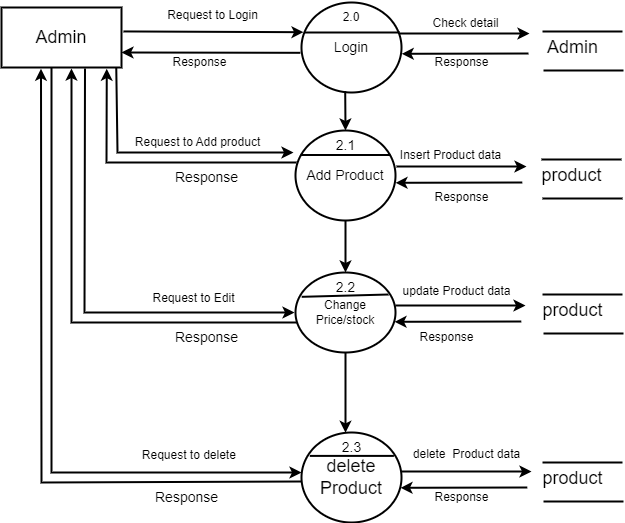
* Level 0 DFD



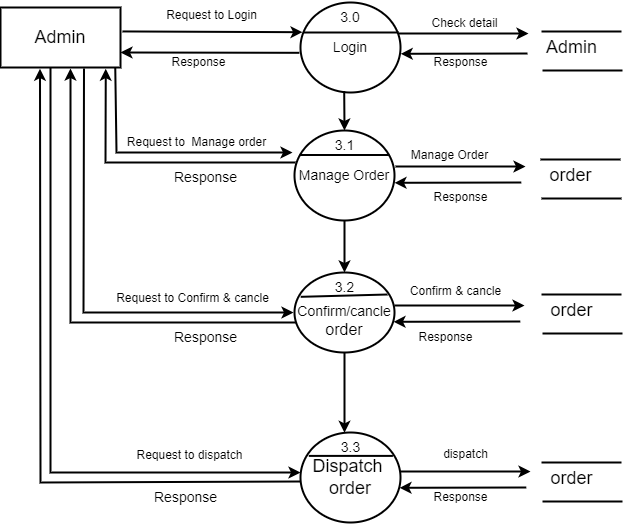
* Level 1 DFD:



* Level 2 DFD(2.0):



* Level 2 DFD(3.0):



Q5>What is Flow chart? Create a flowchart to make addition of two numbers

Flowchart: 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.

Algorithm: A set of finite rules or instruction to be followed in calculation or other problem-solving operations.

#include<stdio.h>

Int main()

{

Int n1,n2,ans;

Printf(“Enter 2 values”);

Scanf(“%d%d”,&n1,&n2);

Ans=n1+n2;

Printf(“%d”,ans);

Return 0;

}

Algorithm:

1)start

2)declare variables n1,n2,ans;

3)display enter 2 values

4)read values of n1,n2 from user

5)ans ->n1+n2;

6)display ans

7)stop

FLOWCHART



Q6>What is Use case Diagram? Create a use-case on bill payment on paytm.

* + A Use Case Diagram is a type of diagram used in systems engineering and software engineering to represent the functional requirements of a system. It captures the interactions between various users (actors) and the system itself to illustrate the system's behavior under various conditions. Here are the key elements and concepts associated with use case diagrams

