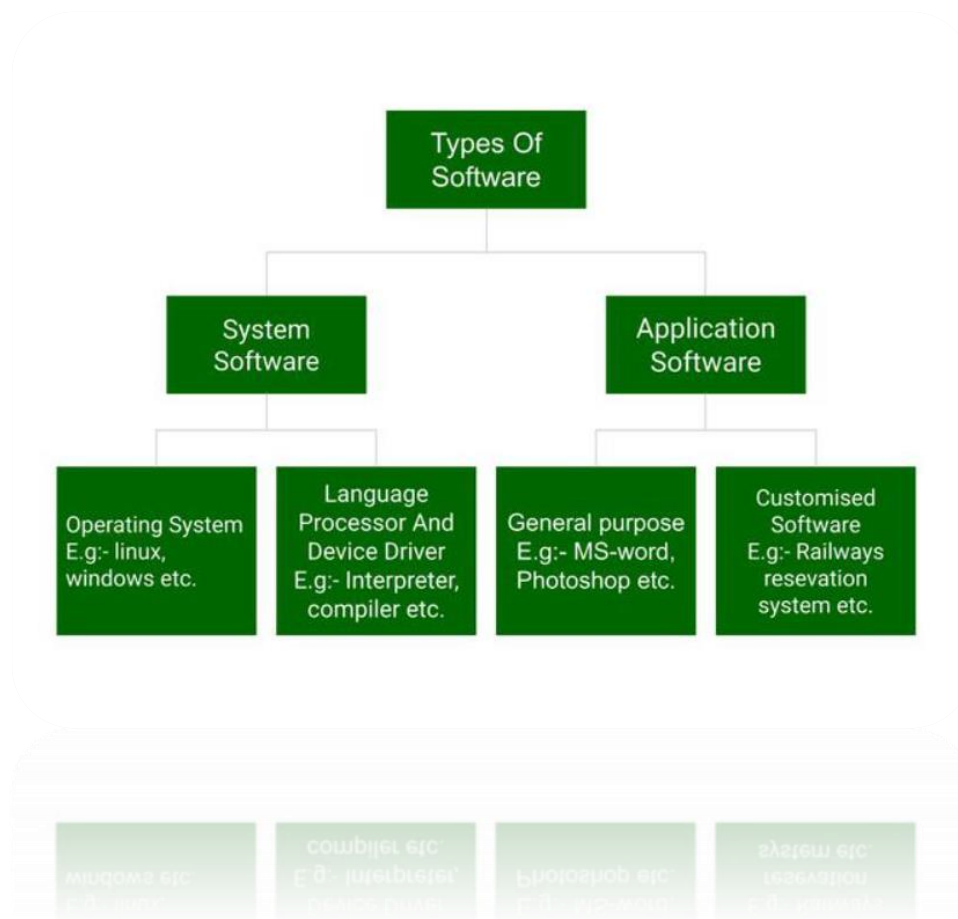# 1. What is software? What is software engineering? ⌑

> ➢ **Software** is a program that can provide some functionalityto perform operation on computer.
> ➢ **Software engineering** is a process of analysing user requirements and then designing, building, and testing software application which will satisfy those requirements.

2. Explain types of software

1.System Software:

- Operating system
- Language processor
- Device Driver

2.Application Software:

- General purpose software
- Customize Software
- Utility software

1> System 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.

1.1 Operating System:

The operating system (OS) manages all the software and hardware on the computer. It performs basic tasks such as file, memory and process management, handling input and output, and controlling peripheral devices such as disk drives and printers.

1.2 Language Processor:

A language processor, or translator, is a computer program that translates source code from one programming language to another. They also identify errors during translation. Computer programs are

usually written in high-level programming languages (like C++, Python, and Java).

1.3 Operating System:

Device Drivers are essential for a computer system to work properly because without a device driver the hardware fails to work accordingly, which means it fails in doing the function/action it was created to do. Most use the term Driver, but some may say Hardware Driver, which also refers to the Device Driver.

2> Application Software: Application software is a type of computer program that performs a specific personal, educational, and business function. Each application is designed to assist end-users in accomplishing a variety of tasks, which may be related to productivity, creativity, or communication.

2.1 General purpose software:

A general-purpose software application is software which has been designed to solve user's common or general tasks using a computer system. Examples include word processors, spreadsheets and photo editing applications for example.

2.2 Customize Software:

Customised software (also known as bespoke software or tailor-made software) is software that is developed specifically for some specific organization or other user.

2.3 Utility software:
Antivirus, backup software, file manager, and disk compression tool all are utility software.

3. What is SDLC? Explain each phase of SDLC

A software life cycle model (also termed process model) is a pictorial and diagrammatic representation of the software life cycle. A life cycle model represents all the methods required to make a software product transit through its life cycle stages.

- Stage 1: Project Planning
- Stage 2: Gathering Requirements & Analysis
- Stage 3: Design
- Stage 4: Coding or Implementation
- Stage 5: Testing
- Stage 6: Deployment
- Stage 7: Maintenance

1. Requirements gathering and analysis: This phase involves gathering information about the software requirements from stakeholders, such as customers, end-users, and business analysts.

2. Design: In this phase, the software design is created, which includes the overall architecture of the software, data structures, and interfaces. It has two steps:

  ➢ High-level design (HLD): It gives the architecture of software products.

➢ Low-level design (LLD): It describes how each and every feature in the product should work and every component.

3. Implementation or coding: The design is then implemented in code, usually in several iterations, and this phase is also called as Development.

things you need to know about this phase:

➢ This is the longest phase in SDLC model.
➢ This phase consists of Front end + Middleware + Back-end.
➢ In front-end: Development of coding is done even SEO settings are done.
➢ In Middleware: They connect both the front end and back end.
➢ In the back-end: A database is created.

4. Testing: The software is thoroughly tested to ensure that it meets the requirements and works correctly.

5. Deployment: After successful testing, The software is deployed to a production environment and made available to end-users.

6. Maintenance: This phase includes ongoing support, bug fixes, and updates to the software.

There are different methodologies that organizations can use to implement the SDLC, such as Waterfall, Agile, Scrum, V-Model and DevOps.

4. 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. It is a graphical tool, useful for communicating with users ,managers and other personnel. it is useful for analyzing existing as well as proposed system.
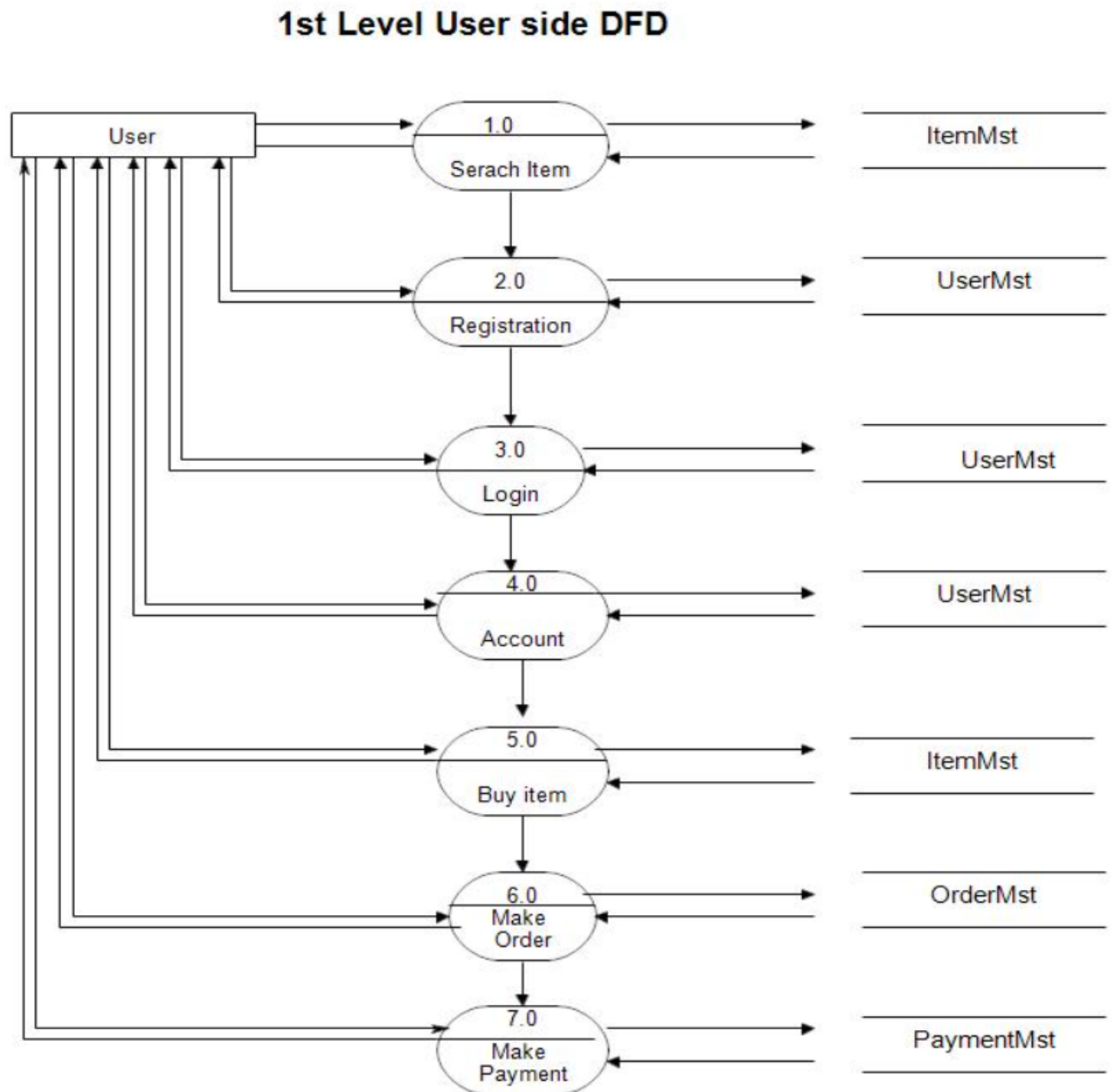
0 level DFD :


1st Level Admin Side DFD:

The admin side DFD describe the functionality of Admin, Admin is an owner of the website. Admin can first add category of item and then add items by

category wise. and admin can manage order and payment detail.

1st Level User side DFD

User → 1.0 Serach Item	→	ItemMst	
→ 2.0 Registration	→	UserMst	
→ 3.0 Login	→	UserMst	
→ 4.0 Account	→	UserMst	
→ 5.0 Buy item	→	ItemMst	
→ 6.0 Make Order	→	OrderMst	
→ 7.0 Make Payment	→	PaymentMst	

5. 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.
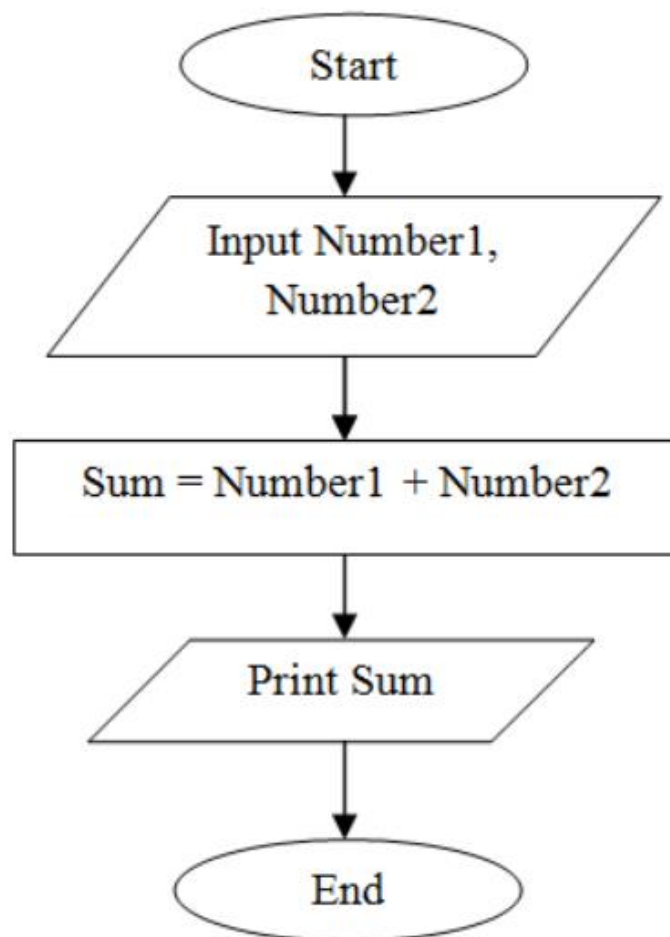
6. What is Use case Diagram? Create a use-case on bill payment on Paytm.

A use case is a written description of how users will perform tasks on your website. It outlines,

from a user's point of view, a system's behaviour
as it responds to a request. Each use case is
represented as a sequence of simple steps,
beginning with a user's goal and ending when that
goal is fulfilled.