Software Engineering Assignment

------------------------------------------------------------------------------------------------

MODULE: 1 (SDLC)

------------------------------------------------------------------------------------------------

1. What is Software? What is Software Engineering?

Ans. **Software** is a set of instructions, data or programs used to operate computers and execute specific tasks.

-The term software engineering is the product of two words, software and engineering.

-The software is a collection of integrated programs.

-Software subsists of carefully-organized instructions and code written by developers on any of various particular computer languages.

-Computer programs and related documentation such as requirements, design models and user manuals.

-Engineering is the application of scientific and practical knowledge to invent, design, build, maintain, and improve frameworks, processes, etc.

-**Software Engineering** is an engineering branch related to the evolution of software product using well-defined scientific principles, techniques, and procedures. The result of software engineering is an effective and reliable software product.

------------------------------------------------------------------------------------------------

2. Explain types of Software?

Ans. There are three types of Software: -

1.Application Software

2.System Software

3.Programming Software

1. 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.

-Examples of modern applications include office suites, graphics software, databases and database management programs, web browsers, word processors, software development tools, image editors and communication platforms.

2. System Software: -

-These software programs are designed to run a computer's application programs and hardware. System software coordinates the activities and functions of the hardware and software.

-In addition, it controls the operations of the computer hardware and provides an environment or platform for all the other types of software to work in. The OS is the best example of system software; it manages all the other computer programs.

-Other examples of system software includes the firmware, computer language translators and system utilities.

3. 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.

-Examples of programming software include assemblers, compilers, debuggers and interpreters.

------------------------------------------------------------------------------------------------

3. What is SDLC? Explain each phase of SDLC

Ans. SDLC stands for "Software Development Life Cycle".

-SDLC is a collection of Processes which are followed to develop a software.

-SDLC is a Methodology (Lot of Methods) that defines some processes which are followed to develop a high-quality software.

-It covers the detailed plan for building, deploying and maintaining the software.

-The main aim of SDLC is to define all the tasks required for developing and maintaining the software.

-It is followed for a software project within a software developing Organizations.

==> Phases of SDLC:

1. Requirement:

-It is first phase of SDLC in which all the necessary information is collected from the customer to develop the software as per their expectation.

-Some important question like: what is the need of software, who will be the end-users, what is the future scope of that software, etc. are discussed.

-The main aim of this phase is to collect the details of requirement of the customer so that the developers will clearly understand what they are developing and how to fulfil the customer's requirements.

-This phase gives a clear picture of what we are going to build.

2. Analysis and Feasibility Study:

-It is the second phase of SDLC in which an organization discusses about the cost and benefits of the software.

-It is an important phase because profit from the software plays an important role as if cost is very high then company may face loss.

-After the feasibility study, the project may be accepted, accepted with modifications or rejected.

-It measures how much benefits the product is for the organization.

3. Design:

-It is third phase in which architects start working on logical designing of the software.

-In this phase a SRS (System Requirement Specification) document is created which contains all logical details like how the software will look like, which language will be used, database design, modular designs etc.

-This phase provides a prototype of the final product.

-Basically, all it includes is design of everything which has to be coded.

4. Coding:

-When the designing of the software is completed, then, a group of developers starts coding of the design using a programming language.

-The interface of the software and all its internal working according to design phase is implemented in coding phase.

-A number of developers code the modules and then all modules are arranged together to work efficiently.

-It is the longest phase of SDLC.

5. Testing:

-Once the software development is completed, then it is sent to the testers. The testing team starts testing the functionality of the entire system.

-In this phase, the software is checked for bugs or errors.

-Whenever a bug is found, then the software is resent to the coders to fix it and then overall software is re-tested.

-This is done to verify that the entire application works according to the customer requirement.

6. Maintenance:

-The maintenance team look over the software usage and user’s feedbacks.

-Maintenance is necessary to eliminate errors in the system during its working life and to tune the software.

-The bug fixing, upgrade and enhancement of the software is looked over by the maintenance team.

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

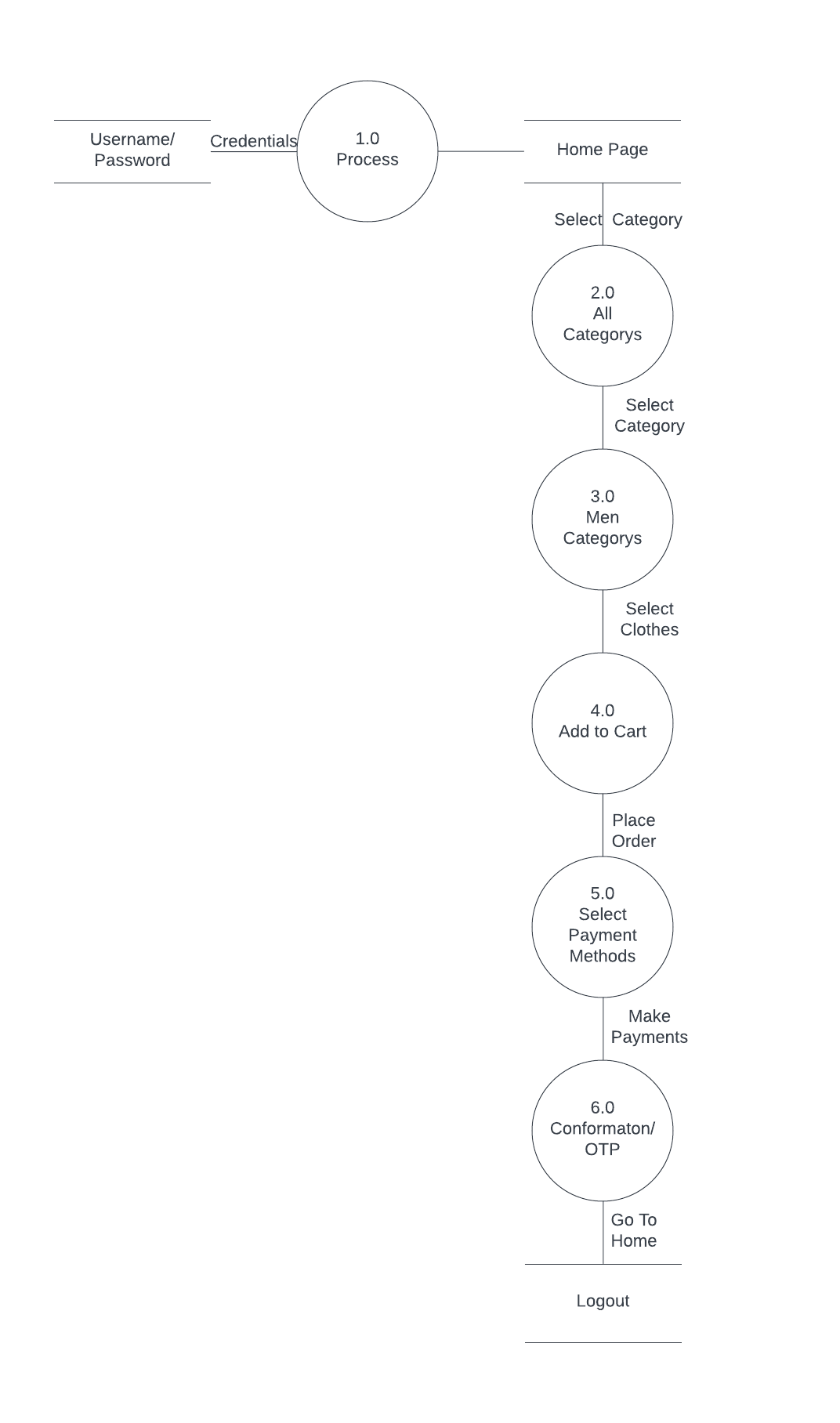
Ans. A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It can be manual, automated, or a combination of both.

-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. The DFD is also called as a data flow graph or bubble chart.

**Components of Data Flow Diagram: -**

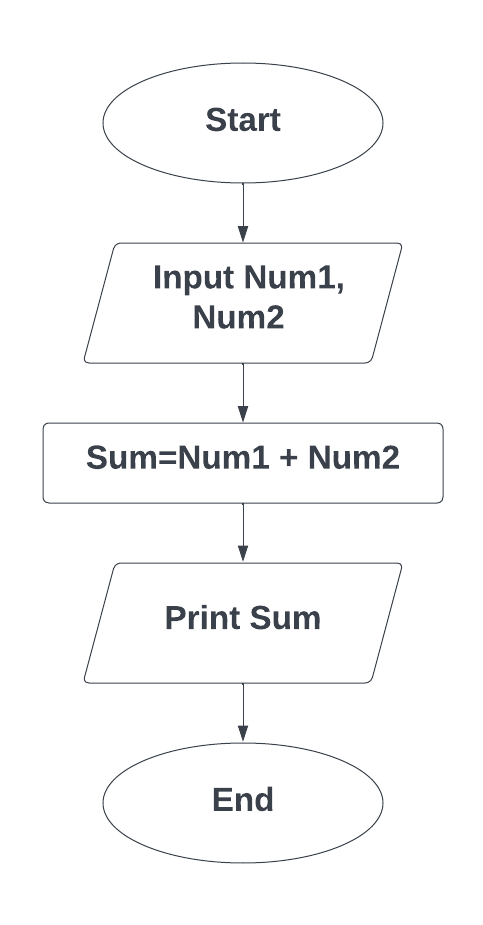
* **Entities:**  
  Entities include source and destination of the data. Entities are represented by rectangle with their corresponding names.
* **Process:**  
  The tasks performed on the data is known as process. Process is represented by circle. Somewhere round edge rectangles are also used to represent process.
* **Data Storage:**  
  Data storage includes the database of the system. It is represented by rectangle with both smaller sides missing or in other words within two parallel lines.
* **Data Flow:**  
  The movement of data in the system is known as data flow. It is represented with the help of arrow. The tail of the arrow is source and the head of the arrow is destination.
* **DFD on Filpkart:-**



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

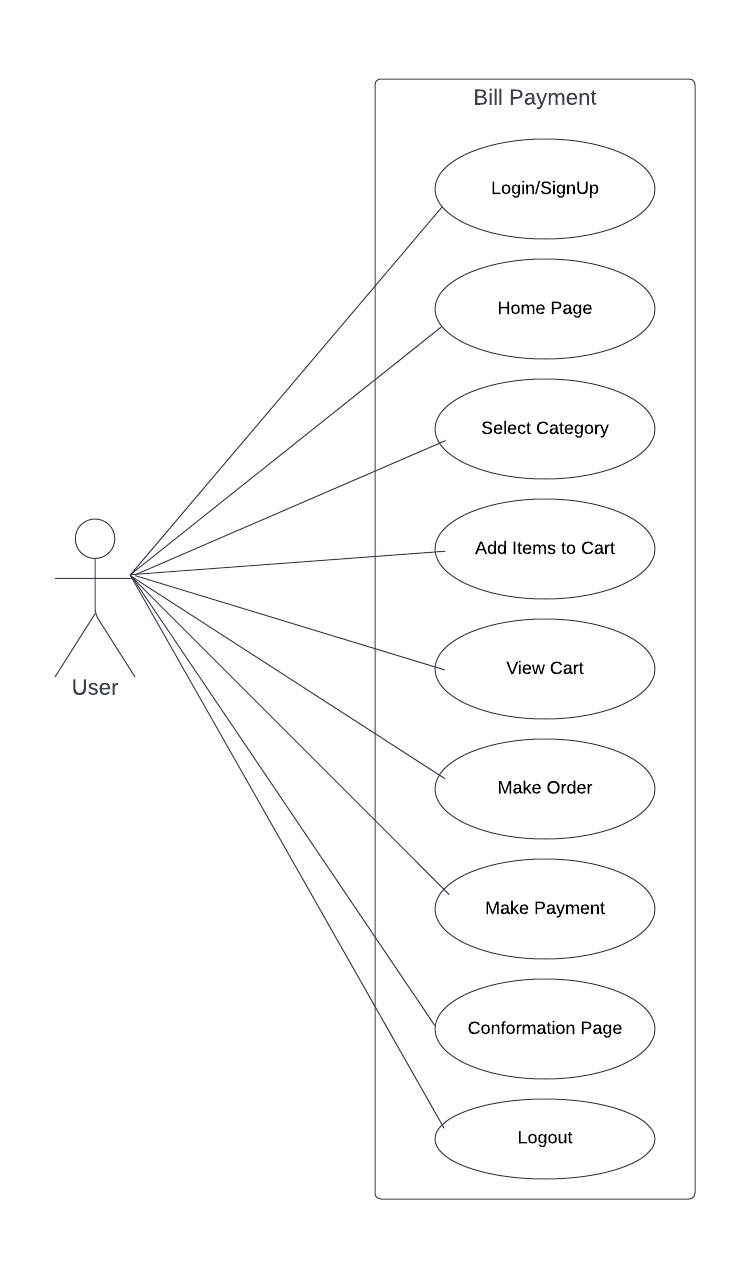
Ans. A flowchart is a type of diagram that represents a workflow or process. A flowchart can also be defined as a diagrammatic representation of an algorithm, a step-by-step approach to solving a task.

**Flowchart to make addition of two numbers: -**



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

Ans. A use case diagram is a visual summarization of interactions and relationships within a system. These diagrams show a very broad view of a system. They may show systems in computer software, businesses or customer experiences. A use case diagram shows a model scenario in which individuals interact with a system using a series of specialized symbols and connectors.



------------------------------------------------------------------------------------------------