**MODULE: 1**

SE – Overview of IT Industry

1. What is software? What is software engineering?

* 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 is the process of designing, developing, testing, and tools to the design, development, and maintaining software. It is a systematic and disciplined approach to software development that aims to create high-quality, reliable, and maintainable software.

1. Explain types ofsoftware

Among the various categories of software, the most common typesinclude the following:

* 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.

* 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 include the firmware, computer language translators and system utilities.

* Driver software:

Also known as device drivers, this software is often considered a type of system software. Device drivers control the devices and peripherals connected to a computer, enabling them to perform their specific tasks. Every device that is connected to a computer needs at least one device driver to function. Examples include software that comes with any nonstandard hardware, including special game controllers, as well as the software that enables standard hardware, such as USB storage devices, keyboards, headphones and printers.

* Middleware:

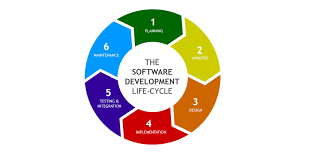
The term middleware describes software that mediates between application and system software or between two different kinds of application software. For example, middleware enables Microsoft Windows to talk to Excel and Word. It is also used to send a remote work request from an application in a computer that has one kind of OS, to an application in a computer with a different OS. It also enables newer applications to work with legacy ones.

* 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.

1. 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. SDLC consists of several distinct phases, each with its own set of activities and objectives. The phases of SDLC typically include:



1. Planning:Decide what software you want to create, what it should do, and who will use it.

2. Analysis: Understand in detail what the software needs to do and how it will work. The software development team works closely with stakeholders to understand the needs of the end-users and to define the system's functionality.

3. Design:Plan out how the software will look and how all the different parts will fit together. In this phase, the software architecture is designed based on the requirements and analysis conducted in the previous stages. This includes defining the system's structure, components, interfaces, and data models.

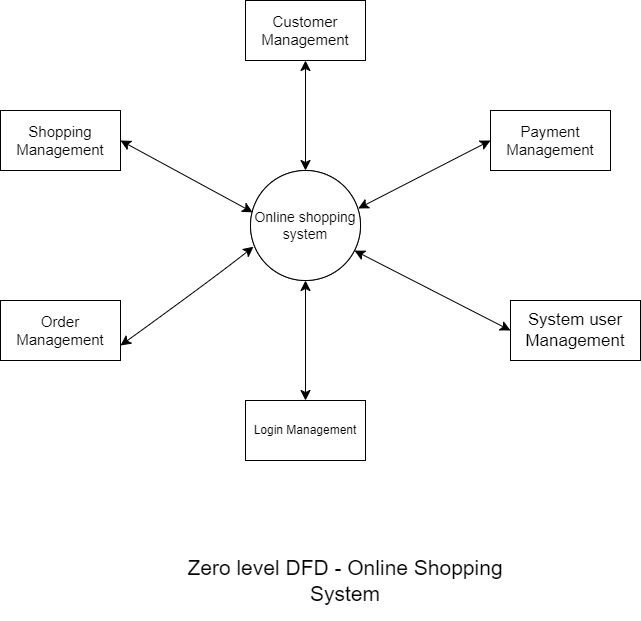
4. Implementation:Write the actual code for the software based on the design. Also known as the coding phase, this is where the actual software is developed. Programmers write code according to the design specifications, following coding standards and best practices.

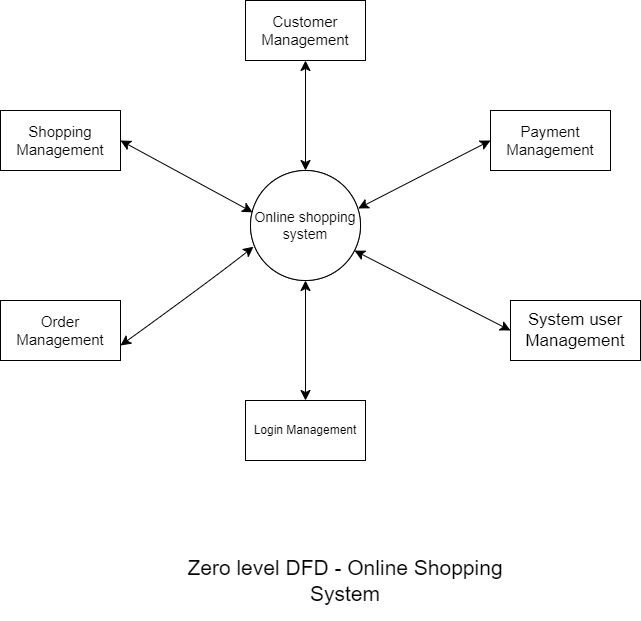
5. Testing:Check the software to make sure it works correctly and meets the specified requirements. Various testing techniques, such as unit testing, integration testing, and system testing, are used to identify and fix defects.

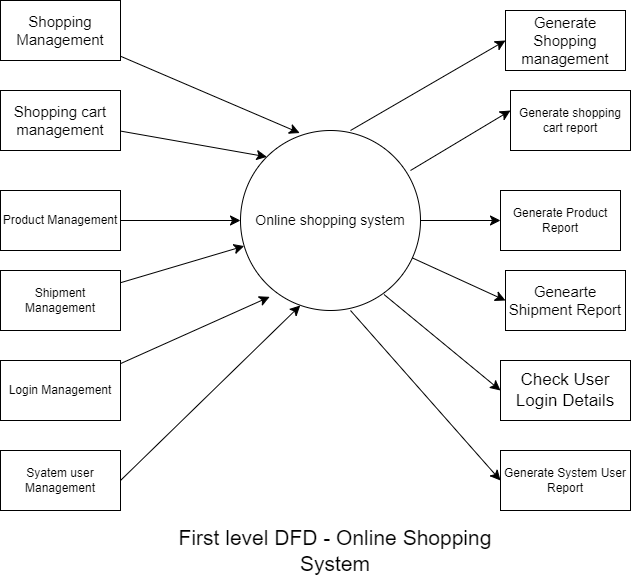
6. Deployment:Put the finished software into use, whether it's installing it on computers or making it available for people to download.

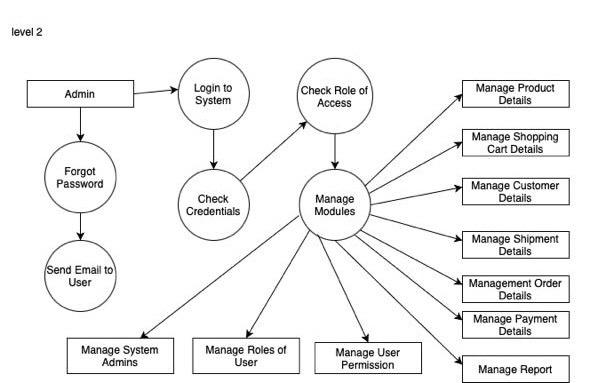
7. Maintenance:Keep the software running smoothly by fixing any problems that come up and making improvements over time. This phase may involve fixing bugs, adding new features, or optimizing performance based on user feedback and changing requirements. Following these steps helps ensure that the software is made correctly, works well, and can be updated and improved as needed.

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



Login Management





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

Flow chart is graphical representation of algorithm:

Algorithm:

1: Start

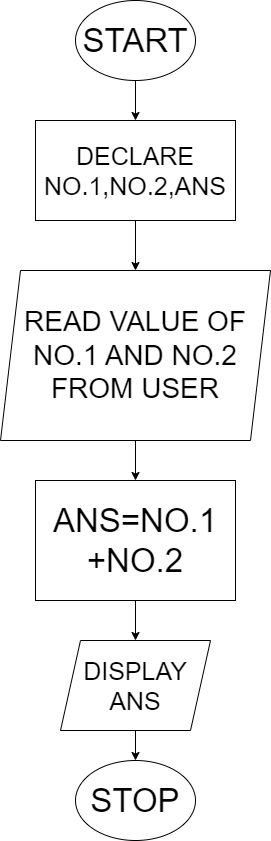
2: Declare no.1, no.2, ans

3: Read value of no.1 and no.2 from users

4: ans = no.1 +no.2

5: Display ans

6: Stop



1. What is Use case Diagram? Create a use-case on bill payment on paytm
2. Top of Form

* A use case diagram is a visual representation in Unified Modeling Language (UML) that depicts how users interact with a system to accomplish specific tasks.

