

Module (HTML5) -3

**MODULE: 1**

**SE – Overview of IT Industry**

1. **What is software? What is software engineering?**

***Software***

Software is basically a set of instructions or commands that tell a computer what to do. In other words, the software is a computer program that provides a set of instructions to execute a user’s commands and tell the computer what to do. For example, like MS-Word, MS-Excel, PowerPoint, etc.

***Software engineering***

Software Engineering is the process of designing, developing, testing, 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 of software**

Software can be categorized into several types based on various criteria such as its purpose, functionality, and licensing model. Here are some common types of software:

* **System Software:** This type of software is essential for the operation of computer hardware and provides a platform for running application software. Examples include operating systems like Windows, macOS, and Linux, as well as device drivers, firmware, and utility programs like antivirus software and disk management tools.
* **Application Software:** Application software is designed to perform specific tasks or provide specific functionality for end-users. It can range from productivity tools like word processors, spreadsheets, and presentation software to multimedia applications, graphics design programs, and web browsers.
* **Programming Software:** Programming software consists of tools and applications used by software developers to create, debug, and maintain other software. This includes integrated development environments (IDEs), text editors, compilers, debuggers, and version control systems.
* **Middleware:** Middleware acts as an intermediary between different software applications, enabling communication and data exchange between them. Examples include web servers, database management systems, and enterprise application integration (EAI) software.
* **Embedded Software:** Embedded software is specialized software that is built into hardware devices to control their functions. It is commonly found in consumer electronics, automotive systems, industrial machines, and IoT devices.
* **Enterprise Software:** Enterprise software is designed to support large organizations or businesses in managing their operations and processes. This includes enterprise resource planning (ERP) systems, customer relationship management (CRM) software, and supply chain management (SCM) solutions.
* . **Proprietary Software:** Proprietary software is owned and controlled by a single company or entity, and its source code is typically not available to the public. Users must usually purchase a license to use proprietary software. Examples include Microsoft Office, Adobe Photoshop, and Oracle Database.

1. **What is SDLC? Explain each phase of SDLC**

SDLC stands for Software Development Life Cycle. It is a structured approach to software development that outlines the stages and activities involved in building software systems. The SDLC consists of several phases, each with its own set of goals, deliverables, and activities.

* Planning:

The first step in the software development life cycle is planning. It's the team set goals, and identify risks. At this stage, the team will work together to devise a set of business goals, requirements, specifications, and any high-level risks that might hinder the project's success.

* Analyze Requirements:

Once come up with some ideas, it's time to organize them into a cohesive plan and design. This requires a lot of research and planning to ensure that your final product meets expectations (and those of customers). The big step is creating a detailed project plan document and work breakdown structure that outlines the requirements.

* Design:

Once design plans is ready, it's time for wireframing and mockups. This step builds upon the planning stage, building out the tasks need to do in the work breakdown schedule. There are plenty of tools available, such as Adobe XD or InVision, that make this process much easier than ever before.

* Develop the Code:

The development phase is where coding begins to take place. It is one of the most time-consuming phases in the SDLC. This phase often requires extensive programming skills and knowledge of databases. The team will build functionality for the product or ser

vice, which includes creating a user interface and building the database so users can store information in your system.

* Test the product:

Before releasing the mockups into final production, you'll need to test it to ensure it is free of bugs and errors. Any issues need to be fixed before moving forward with deployment. You'll also need to manage how the system will integrate into existing systems, software, and processes.

* Implementations:

Once you've completed all testing phases, it's time to deploy your new application for customers to use. After deployment, the launch may involve marketing your new product or service so people know about its existence. If the software is in-house, it may mean implementing the change management process to ensure user training and acceptance.

* Maintenance:

The final stage of the software development life cycle is maintenance and operations. This is one of the most critical stages because it's when your hard work gets put to the test.

Maintenance involves updating an existing software product to fix bugs and ensure reliability. It can also include adding new features or functionality to a current product. Operations refer to the day-to-day running of a software product or service, such as performing backups and other administrative tasks.

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

A Data Flow Diagram (DFD) is a graphical representation of the flow of data within a system. It illustrates how data is input, processed, stored, and output in a system. DFDs are commonly used in the analysis and design of information systems to model the system's functionality at different levels of abstraction.

Creating a DFD for Flipkart, an e-commerce platform, involves identifying the main processes and data flows within the system. Here's a high-level DFD for Flipkart:

Flipkart

Customer

Seller

Catalog Management

Inventory Management

Order System

Cart System

Payment Gateway

**Explanation of the DFD elements:**

* **Customer and Seller**: These represent the external entities interacting with the Flipkart system. Customers browse products, place orders, and make payments. Sellers manage their product listings, inventory, and receive orders.
* **Catalog Management:** This process involves managing the product catalog, including adding new products, updating existing products, and removing discontinued products.
* **Inventory Management:** This process handles inventory-related tasks, such as updating stock levels, tracking product availability, and managing warehouse logistics.
* **Order System:** This process manages the order lifecycle, including order placement, order processing, order fulfillment, and order tracking.
* **Cart System:** This process manages the shopping cart functionality, allowing customers to add/remove items, view cart contents, and proceed to checkout.
* **Payment Gateway:** This external entity facilitates secure online payments, handling payment processing, transaction authorization, and funds transfer between customers and sellers.
* **Data Flows:** Arrows represent the flow of data between processes, entities, and data stores. For example, data flows from the Catalog Management process to the Order System process when a customer selects a product to purchase.

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

A flowchart is a graphical representation of a process or algorithm, using various shapes and arrows to illustrate the steps involved and the flow of control. It's commonly used in software development, business processes, and other fields to visually communicate the steps of a procedure or workflow

Here's a simple flowchart to illustrate the process of adding two numbers:

**Input Number 1, Number 2**

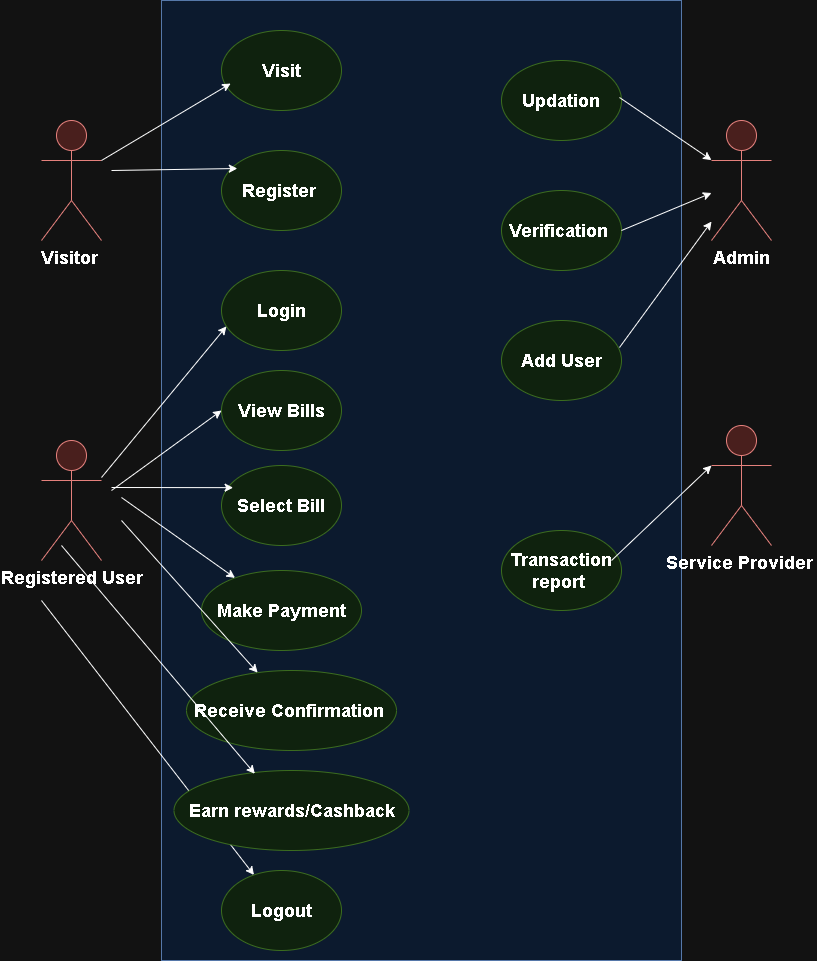
**Sum=Number 1 +Number 2**

**Print Sum**

1. **What is Use case Diagram? Create a use-case on bill payment on paytm.**

A Use Case Diagram is a visual representation of the functional requirements of a system from the perspective of its users. It illustrates the interactions between users (actors) and the system to achieve specific goals or tasks.

**USE-CASE OF PAYTM**



**Use-Case for payment on Paytm**