**MODULE: 1 (SDLC) Assignment**

**1. What is Software? What is Software engineering?**

**Software is a set of instructions or programs that allow a computer to perform a specific task. Software can be divided into two categories: system software and application software. System software is responsible for running the computer's operating system and managing its resources. Application software is used to perform specific tasks, such as word processing or email**.

**Software is a program or set of programs containing instructions that provide desired functionality. And Engineering is the process of designing and building something that serves a particular purpose and finds a cost-effective solution to problems.**

**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. Software engineering includes a variety of techniques, tools, and methodologies, including requirements analysis, design, testing, and maintenance.**

**2. Explain types of Software.**

**Software, which is abbreviated as SW or S/W, is a set of programs that enables the hardware to perform a specific task. All the programs that run the computer are software. The software types: system software, application software, middleware and programming software**.

**System software:-**

* **Software is a set of instructions, which is designed to perform a defined task, and it tells the computer how to work.it is of mainly two types, namely system software and Application software. System software is a set of computer programs that is designed to manage system resources. It is a collection of such files and utility programs that are responsible for running and smooth functioning of your computer system with other hardware. Moreover, it is solely responsible for running the operating system and managing the computer device entirely.**
* **System software acts as a platform for other software to work, such as antivirus software, OS, compiler, disk formatting software, etc. In this topic, we will know more about the system software. Let's start with the definition of System Software.**

**Operating system:-**

* **An Operating System is the most basic type of System Software that helps to manage computer hardware and software. It is the central part of any computer system which is responsible for the smooth functioning of any computer device.**
* **An Operating system primarily operates your computer when you start it. If you haven't installed the operating system on your computer, then you will not be able to start your computer. Some most common examples of OS are macOS, Linux, Android, and Microsoft Windows.**

### ****Device Driver:-****

* **Device Drivers are the types of system software that reduce the troubleshooting issues in your system. The operating system communicates with hardware components internally. This communication can easily be managed and controlled with the help of device drivers.**
* **The operating system contains a number of device drivers to drive the hardware components. Most of the device drivers, such as a mouse, keyboards, etc., are already installed in the computer system by the computer manufacturing companies. However, in case of any new device for the operating system, users can install them through the internet also.**

**Firmware:-**

* **Firmware is an operational software stored in a flash, ROM, EPROM for the OS to identify it. The firmware provides instructions on how the device should be operated. Unlike other software, firmware can not be manipulated, changed, or deleted by the end-users. They remain on the device.**

#### **BIOS and UEFI:-**

* **BIOS (Basic Input/Output System) or the new UEFI (Unified Extended Firmware Software) gets the computer system started when we turn it on. BIOS also manages the flow of information between operating systems and the attached devices.**

#### **Utilities:-**

* **Utilities are the type of system software that is present between user and application software. These are the programs designed to configure, analyze, optimize and maintain tasks of the computer. Their task varies from disk fragmentation to data security.**

**Application Software:-**

* **Application software is a set of programs designed to perform a specific task. It does not control the working of a computer as it is designed for end-users. Application software can be easily installed or uninstalled as required. It can be a single program or a collection of small programs. Microsoft Office Suite, Adobe Photoshop, and any other software like payroll software or income tax software are application software. As we know, they are designed to perform specific tasks.**
* **Application Software helps in providing a graphical user interface to the user to operate the computer for different functionality. The user may use the computer for browsing the internet, accessing to email service, attending meetings, and playing games.**
* **Different high-level languages are used to build application software. They basically lie over system software. They are really utilized by end-user just as have explicit usefulness or undertakings which they are intended to perform. This application software is regularly evolved through custom software improvement, in light of prerequisites of clients.**

**Word Processing Software:-**

* **This software allows users to create, edit, format, and manipulate the text and more. It offers lots of options for writing documents, creating images, and more. For example, MS Word, WordPad, Notepad, etc.**

**Spreadsheet Software:-**

* **It is designed to perform calculations, store data, create charts, etc. It has rows and columns, and the data is entered in the cell, which is an intersection of a row and column, e.g., Microsoft Excel.**

**Multimedia Software:-**

* **These software are developed to perform editing of video, audio, and text. It allows you to combine texts, videos, audio, and images. Thus, you can improve a text document by adding photos, animations, graphics, and charts through multimedia software. For example, VLC player, Window Media Player, etc.**

**Enterprise Software:-**

* **These software are developed for business operational functions. It is used in large organizations where the quantum of business is too large. It can be used for accounting, billing, order processing and more. For example, CRM (Customer Relationship Management), BI (Business Intelligence), ERP (Enterprise Resource Planning), SCM (Supply Chain Management), customer support system, and more.**

**Middleware Softwere:-**

* **Middleware is software that lies between an operating system and the applications running on it. Essentially functioning as hidden translation layer, middleware enables communication and data management for distributed applications.**
* **It’s sometimes called plumbing, as it connects two applications together so data and databases can be easily passed between the “pipe.” Using middleware allows users to perform such requests as submitting forms on a web browser, or allowing the web server to return dynamic web pages based on a user’s profile.**
* **Common middleware examples include database middleware, application server middleware, message-oriented middleware, web middleware, and transaction-processing monitors. Each program typically provides messaging services so that different applications can communicate using messaging frameworks like simple object access protocol (SOAP), web services, representational state transfer (REST), and JavaScript object notation (JSON).**
* **While all middleware performs communication functions, the type a company chooses to use will depend on what service is being used and what type of information needs to be communicated. This can include security authentication, transaction management, message queues, applications servers, web servers, and directories.**
* **Middleware can also be used for distributed processing with actions occurring in real time rather than sending data back and forth.**

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

**The software development lifecycle (SDLC) is the cost-effective and time-efficient process that development teams use to design and build high-quality software. The goal of SDLC is to minimize project risks through forward planning so that software meets customer expectations during production and beyond. This methodology outlines a series of steps that divide the software development process into tasks you can assign, complete, and measure**.

* **Software development can be challenging to manage due to changing requirements, technology upgrades, and cross-functional collaboration. The software development lifecycle (SDLC) methodology provides a systematic management framework with specific deliverables at every stage of the software development process. As a result, all stakeholders agree on software development goals and requirements upfront and also have a plan to achieve those goals.**

**Here are some benefits of SDLC:**

* **Increased visibility of the development process for all stakeholders involved**
* **Efficient estimation, planning, and scheduling**
* **Improved risk management and cost estimation**
* **Systematic software delivery and better customer satisfaction**
* **The SDLC Process:-**



**Planning:-**

* **The planning phase typically includes tasks like cost-benefit analysis, scheduling, resource estimation, and allocation. The development team collects requirements from several stakeholders such as customers, internal and external experts, and managers to create a software requirement specification document.**
* **The document sets expectations and defines common goals that aid in project planning. The team estimates costs, creates a schedule, and has a detailed plan to achieve their goals.**

**Analysis:-**

* **The analysis phase in SDLC is the first activity and the most important phase for a product manager. It involves cooperating with all the stakeholders to get a clear understanding of the business requirements and the system specifications. It also includes a feasibility study and identifies any potential risks. The outcome of this phase is the software requirement specification (SRS), which defines the system of functions and is critical for product acceptance by customers.**

**Design:-**

* **In the design phase, software engineers analyze requirements and identify the best solutions to create the software. For example, they may consider integrating pre-existing modules, make technology choices, and identify development tools. They will look at how to best integrate the new software into any existing IT infrastructure the organization may have**.

**Implementation:-**

* **In the implementation phase, the development team codes the product. They analyze the requirements to identify smaller coding tasks they can do daily to achieve the final result.**
* **Implementation starts once the developer gets the Design document. The Software design is translated into source code. All the components of the software are implemented in this phase.**

**Testing:-**

* **Once the software is complete, and it is deployed in the testing environment. The testing team starts testing the functionality of the entire system. This is done to verify that the entire application works according to the customer requirement.**
* **During this phase, QA and testing team may find some bugs/defects which they communicate to developers. The development team fixes the bug and send back to QA for a re-test. This process continues until the software is bug-free, stable, and working according to the business needs of that system.**

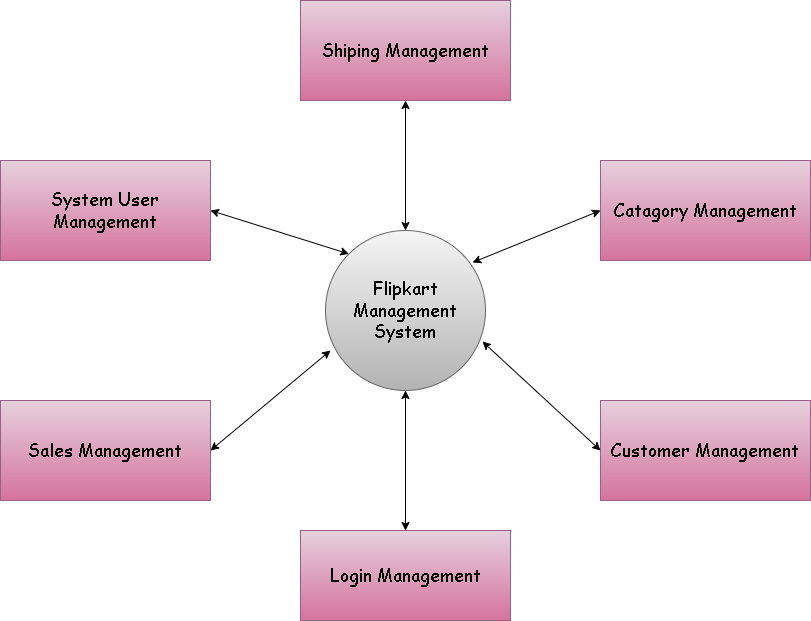
**Maintenance:-**

* **In the maintenance phase, among other tasks, the team fixes bugs, resolves customer issues, and manages software changes. In addition, the team monitors overall system performance, security, and user experience to identify new ways to improve the existing software.**
* **Bug fixing – bugs are reported because of some scenarios which are not tested at all**
* **Upgrade – Upgrading the application to the newer versions of the Software**
* **Enhancement – Adding some new features into the existing software**
* **The main focus of this SDLC phase is to ensure that needs continue to be met and that the system continues to perform as per the specification mentioned in the first phase.**

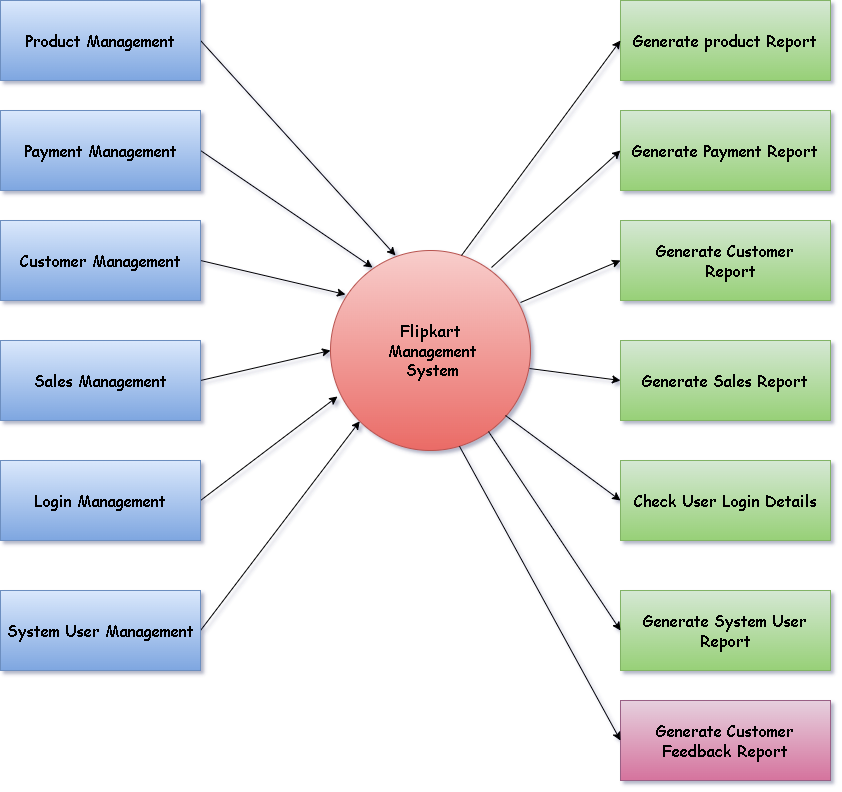
**4. What is DFD? Create a DFD diagram on Flipkart**

**A Data Flow Diagram (DFD) is a traditional way to visualize the information flows within a system. A neat and clear DFD can depict a good amount of the system requirements graphically. It can be manual, automated, or a combination of both.**

* **It shows how information enters and leaves the system, what changes the information and where information is stored. The purpose of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communications tool between a systems analyst and any person who plays a part in the system that acts as the starting point for redesigning a system.**
* **It is usually beginning with a context diagram as level 0 of the DFD diagram, a simple representation of the whole system. To elaborate further from that, we drill down to a level 1 diagram with lower-level functions decomposed from the major functions of the system. This could continue to evolve to become a level 2 diagram when further analysis is required. Progression to levels 3, 4 and so on is possible but anything beyond level 3 is not very common. Please bear in mind that the level of detail for decomposing a particular function depending on the complexity that function.**

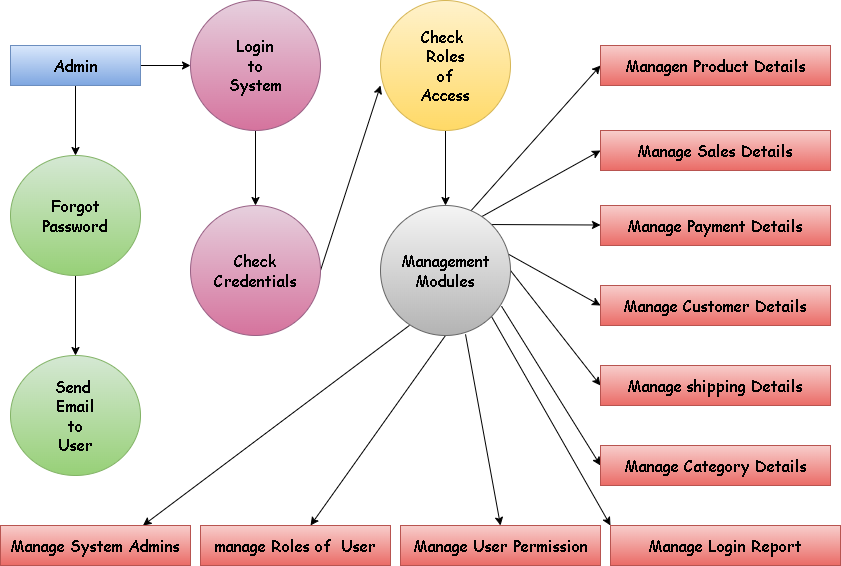
****

**Zero-Level Flipkart DFD Diagram**



**First-Level** **Flipkart DFD Diagram**

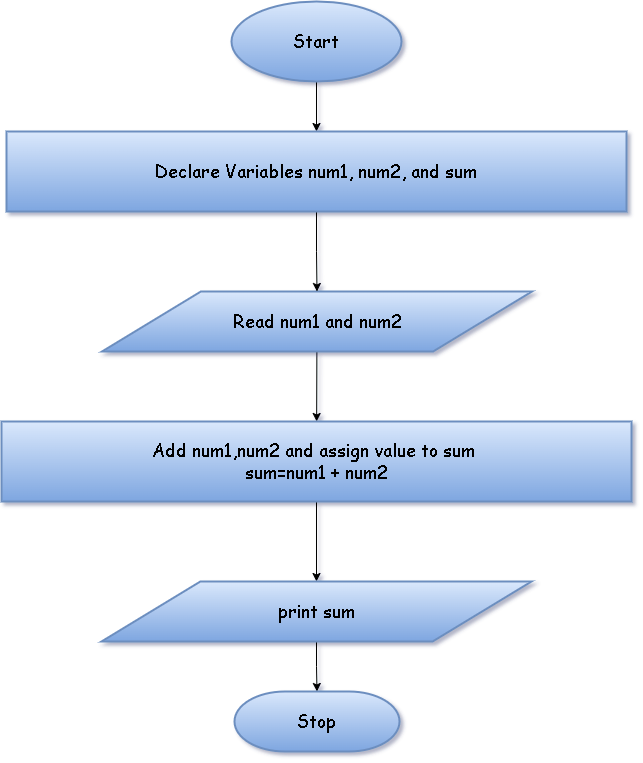
**Second-Level Flipkart DFD Diagram**



**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. Flowcharts, sometimes spelled as flow charts, use rectangles, ovals, diamonds and potentially numerous other shapes to define the type of step, along with connecting arrows to define flow and sequence.**

**They can range from simple, hand-drawn charts to comprehensive computer-drawn diagrams depicting multiple steps and routes. If we consider all the various forms of flowcharts, they are one of the most common diagrams on the planet, used by both technical and non-technical people in numerous fields. Flowcharts are sometimes called by more specialized names such as Process Flowchart, Process Map, Functional Flowchart, Business Process Mapping, Business Process Modeling and Notation (BPMN),  or Process Flow Diagram (PFD). They are related to other popular diagrams, such as Data Flow Diagrams (DFDs) and Unified Modeling Language (UML) Activity Diagrams.**

****

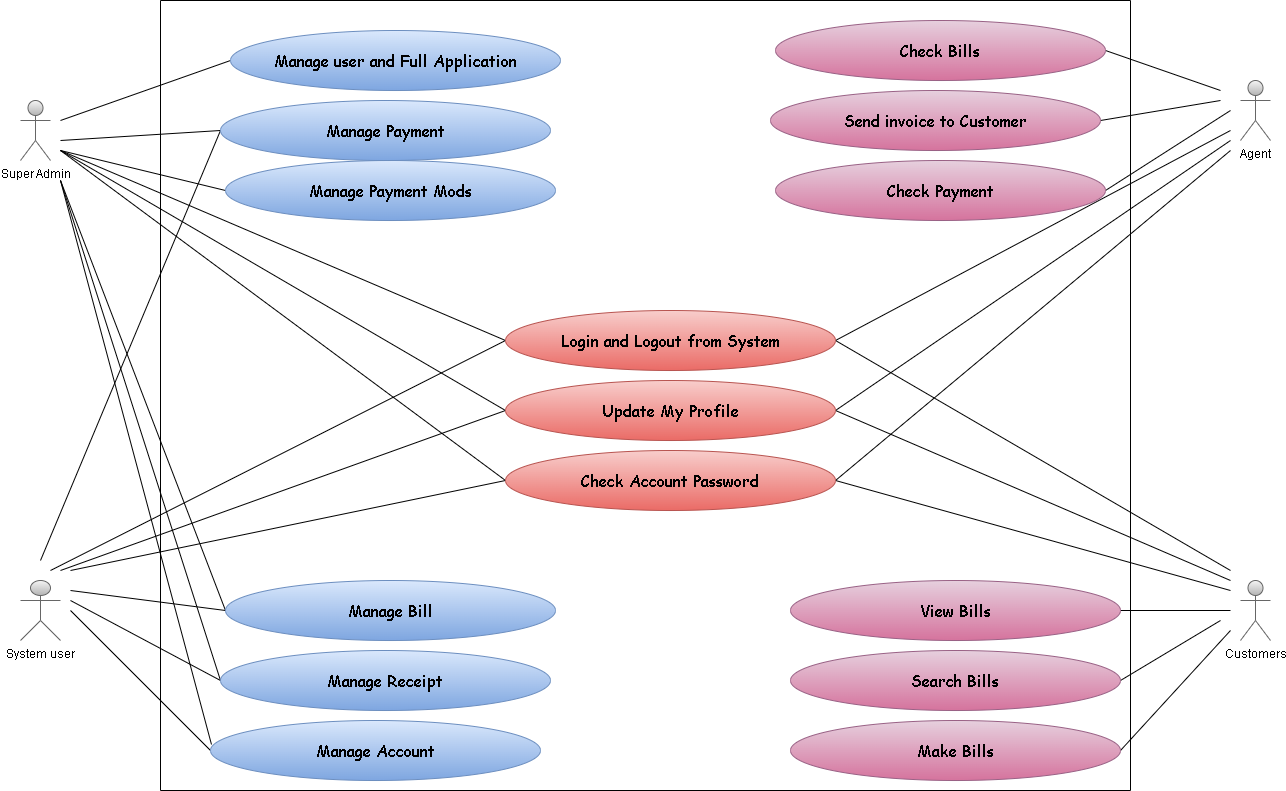
**Two addition number Flowchart**

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

**A use case diagram is used to represent the dynamic behavior of a system. It encapsulates the system's functionality by incorporating use cases, actors, and their relationships. It models the tasks, services, and functions required by a system/subsystem of an application. It depicts the high-level functionality of a system and also tells how the user handles a system.**

* **main purpose of a use case diagram is to portray the dynamic aspect of a system. It accumulates the system's requirement, which includes both internal as well as external influences. It invokes persons, use cases, and several things that invoke the actors and elements accountable for the implementation of use case diagrams. It represents how an entity from the external environment can interact with a part of the system.**

1. **A simple and complete use case diagram should be articulated.**
2. **A use case diagram should represent the most significant interaction among the multiple interactions.**
3. **At least one module of a system should be represented by the use case diagram.**
4. **If the use case diagram is large and more complex, then it should be drawn more generalized.**

****

**Use-case Diagram Paytm Bill Payment**