ASSIGNMENT: -1

1.What is software? What is software engineering?

- Software is collection of data, information and instruction.
- Software Engineering is a technique through we can developed or created software for computer system, mobile system and any another device.

2. Explain Types of Software.

- There are three types of software.
  1.system software
  2.application software
  3.programming software
- **system software**
- System software is the software that runs the system.
- System software is software designed to provide an operating system to another device like mobile, laptop and computer.
- Operating system is the best example of system software.
- Android, iOS, windows, ubuntu, Linux are operating system.
- **Application software**
- Application software is the software that helps you to do a specific type of work.
- Three type of application software.

  1.desktop application
    - In this type of application software, software is work only on desktop.
    - Software runs on a computer device like a desktop or a computer device.
    - Example: notepad, Microsoft access.

  2.web app
    - Web application can be used by anyone who has access to the web browser.
    - Example: web browser like chrome, Mozilla Firefox.

  3.moblie application
    - In this type of application, software is work only on mobile or tab.

- Software runs on a mobile or tab.
- Example: snapchat, Instagram.

Programming software

- Programming software is a software which helps the developed the software.
- Compiler and debugger are the programming software.

  There are two type of programming language.

  1.high-level programming language

  - In this programming language, mediator is converting program to machine language.
  - This mediator is compiler or interpreter.
  - There are four types of programming language.

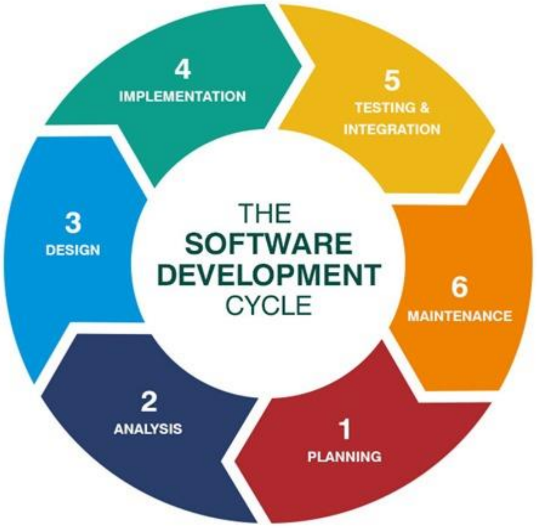
  i) Procedural Oriented programming Language
  ii) Object oriented Programming Language
  iii) Functional Programming Language
  iv) Logical Programming Language

  2. low-level programming language

  I. Machine language
  II. Assembly language

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

- SDLC stands for “Software Development Life Cycle”.
- it describes the sequences of phases or steps to develop any software.
- SDLC life cycle focuses on the following phases of software development:



I. Planning

- Planning for the quality assurance requirements and identifications of the risks associated with the projects is also done at this stage.
- Business analyst and Project organizer set up a meeting with the client to gather all the data like what the customer wants to build, who will be the end user, what is the objective of the product. Before creating a product, a core understanding or knowledge of the product is very necessary.

II. Requirements Analysis

- All necessary information is collected from the customer to develop the software as per their expectation.
- What is need of software, who will be the end-user, what is the future scope of that software etc. are discussed.
- The main aim of this phase is to collect the details of each requirement of the costumer so that the developer will clearly understand what they are developing and how to satisfied costumer’s requirement.
- This phase is given clear image of software or application.

III. Designing

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

- In this phase an SRS 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
- All it includes is design of everything which has to be coded.

IV. **Implementation**

- The actual development begins, and the programming is built. The implementation of design begins concerning writing code.
- Developers have to follow the coding guidelines described by their management and programming tools like compilers, interpreters, debuggers, etc. are used to develop and implement the code.

V. **Testing & Integration**

- 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 created software is checked for bugs or errors.
- Whenever a bug is found, then the software is resent to the order 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.

VI. **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 Flipkart.

- DFD means Graphical representation of flow of data inside application.
- It is Used for communicating with users, managers and other person.
- It is focus on the movement of data between external entities and processes, and between processes and data stores.
- DFD Elements:

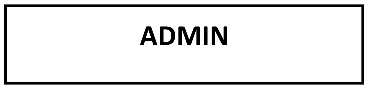
  External Entities

  Data Flows

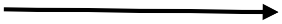
  Processes

  Data Stores

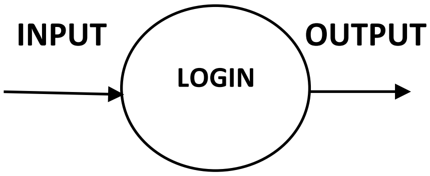
1) External Entities
   - Can be user or external system that performs some process or activity in project Symbolized with rectangle.
   - It is source of system inputs or sink of output data.
   - If we have entity ‘admin’ then symbol will be



2) Data Flows
   - It Can be used to show input and output of data Should be named uniquely and don’t include word ‘data’ Names can be ‘payment’, ‘order’, ’complaint’ etc.
   - It is Symbolized as

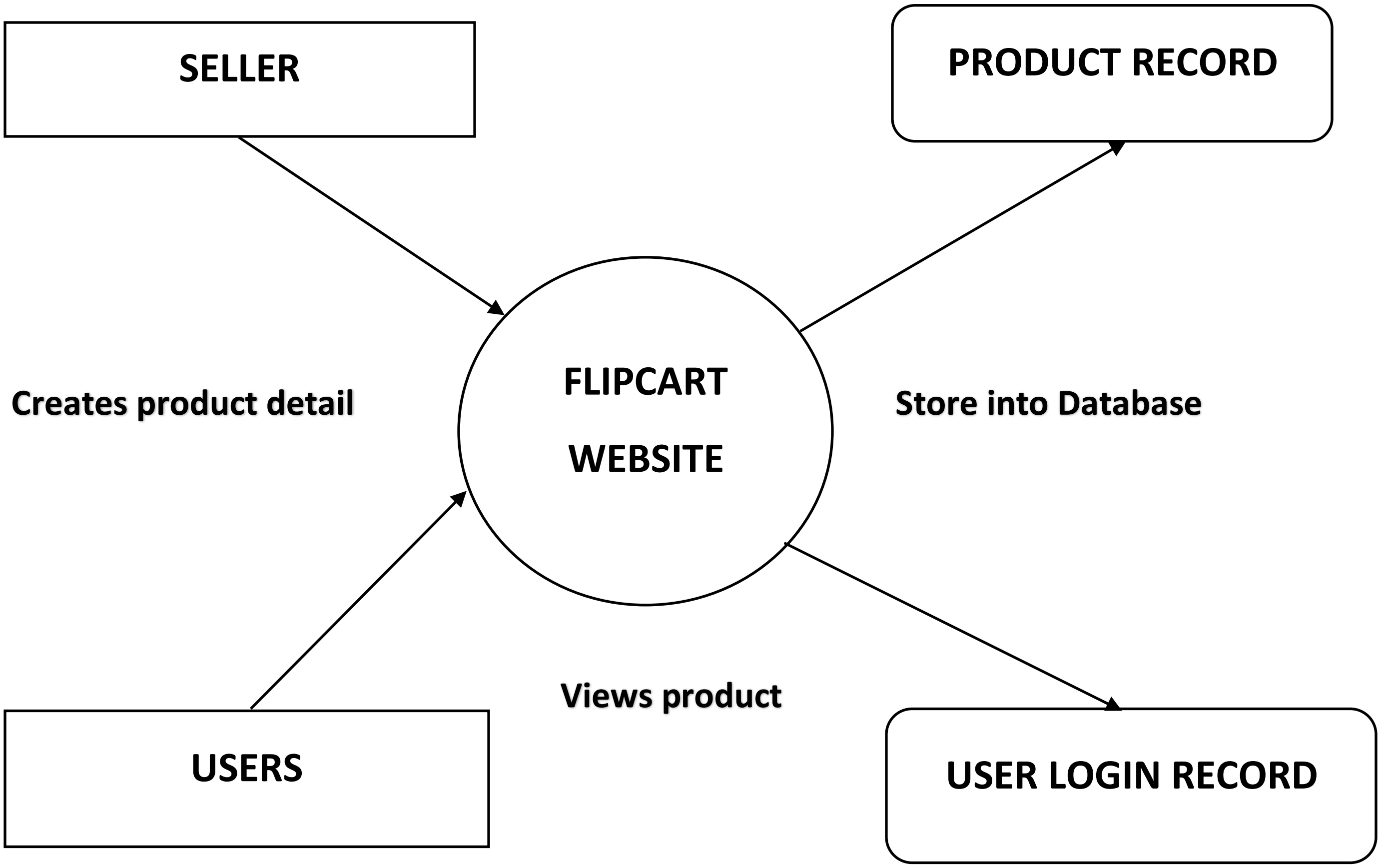


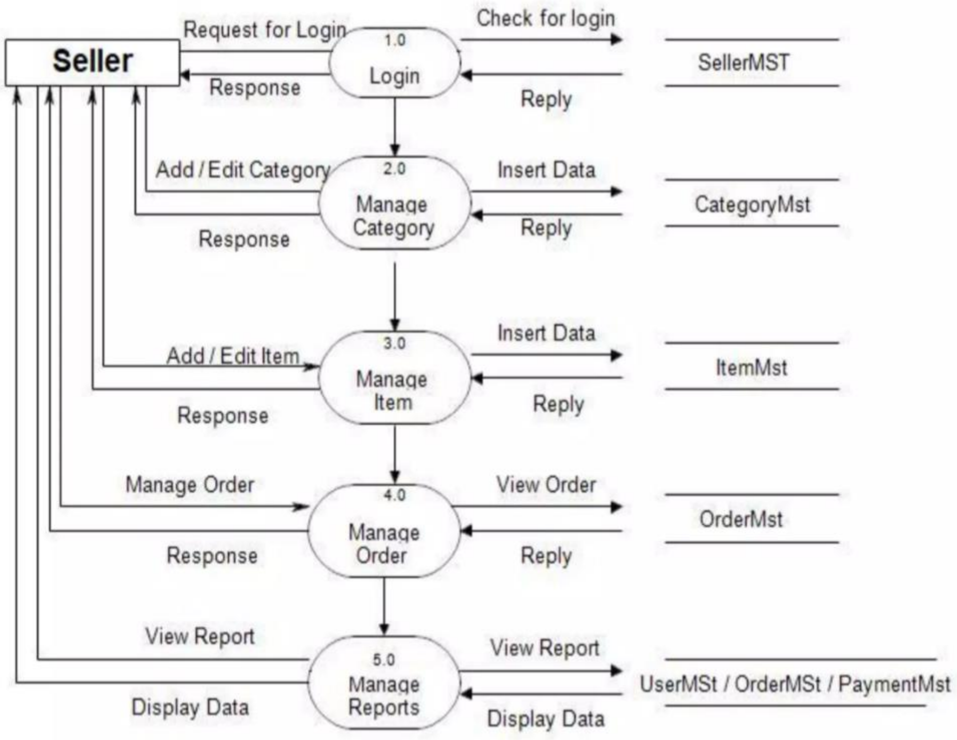
3) Processes
   - Work or action taken on incoming data to product output Each process must have input and output.
   - It is performing transformation of input data to output data.
   - It is Symbolized as

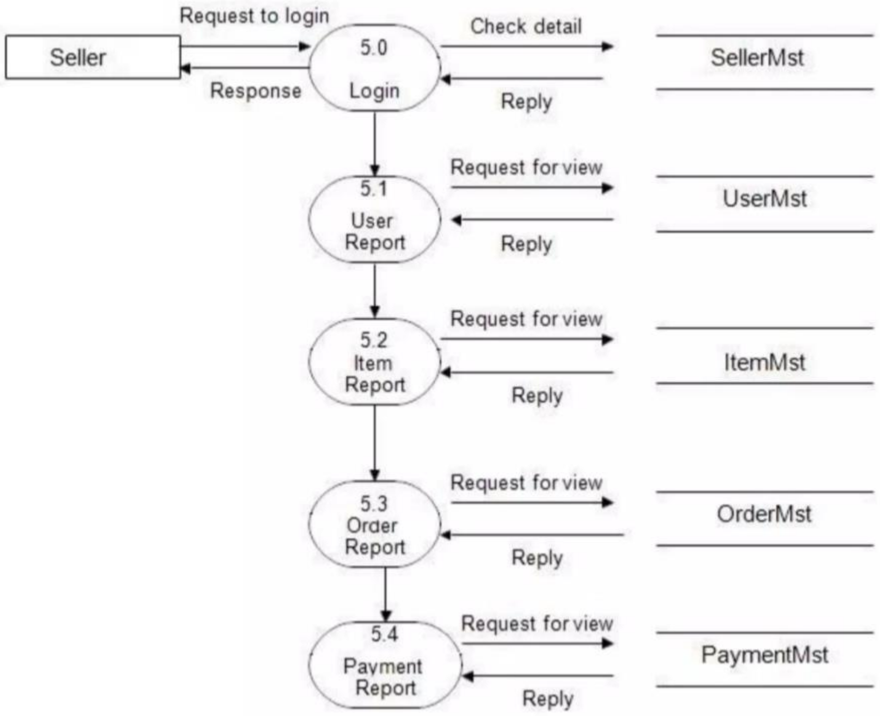


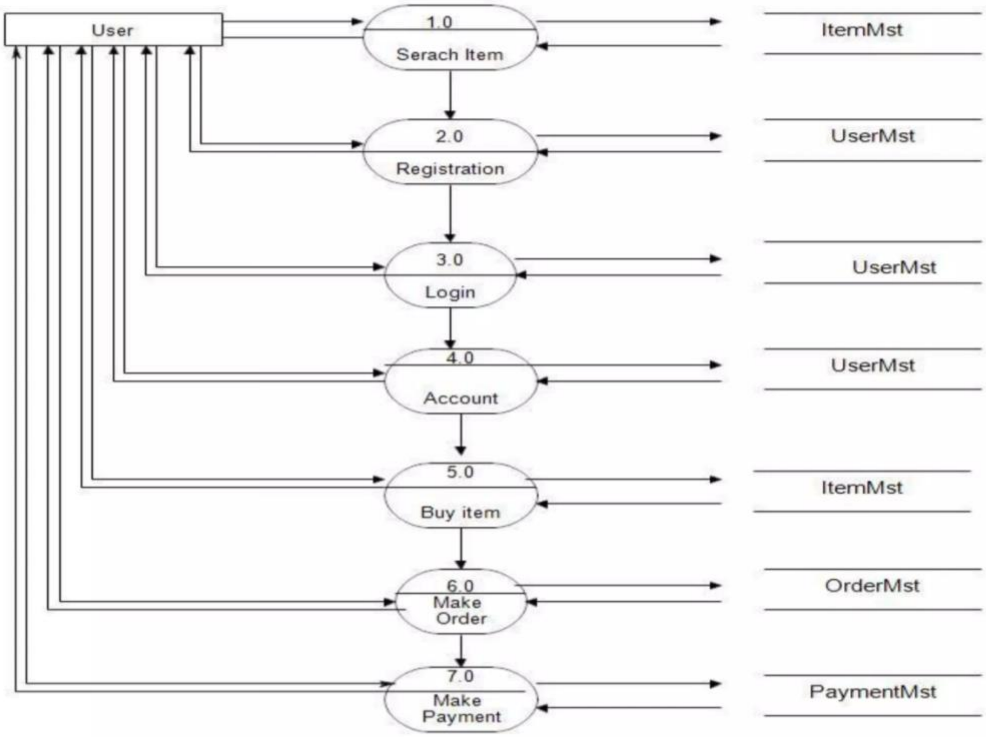
4) Data Stores
   - It Can be used to show database tables Only process may connect data stores.
   - There can be two or more process sharing same data store.
   - It is Symbolized as

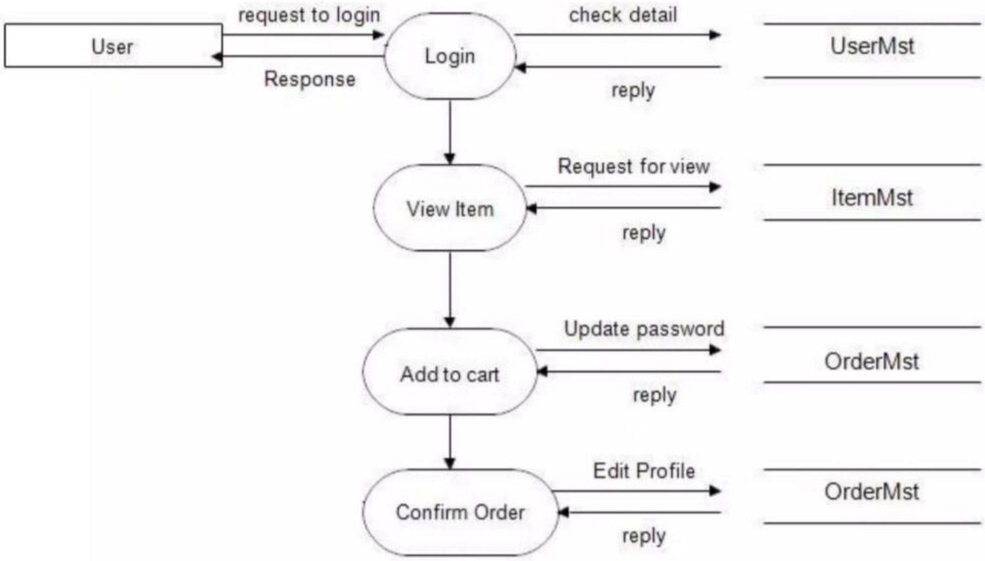
Flip cart DFD Diagram:




Seller
Request for Login
Response
1.0
Login
Check for login
Reply
SellerMST
Add / Edit Category
Response
2.0
Manage
Category
Insert Data
Reply
CategoryMst
Add / Edit Item
Response
3.0
Manage
Item
Insert Data
Reply
ItemMst
Manage Order
Response
4.0
Manage
Order
View Order
Reply
OrderMst
View Report
Display Data
5.0
Manage
Reports
View Report
Display Data
UserMSt / OrderMSt / PaymentMst


Seller
Request to login
Response
5.0
Login
Check detail
Reply
SellerMst
5.1
User
Report
Request for view
Reply
UserMst
5.2
Item
Report
Request for view
Reply
ItemMst
5.3
Order
Report
Request for view
Reply
OrderMst
5.4
Payment
Report
Request for view
Reply
PaymentMst

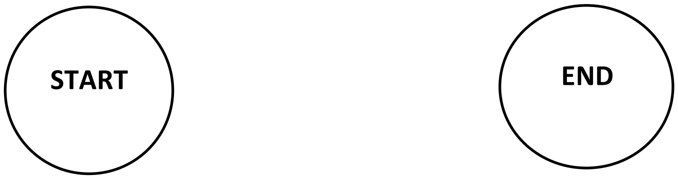

User
1.0
Serach Item
ItemMst
2.0
Registration
UserMst
3.0
Login
UserMst
4.0
Account
UserMst
5.0
Buy item
ItemMst
6.0
Make
Order
OrderMst
7.0
Make
Payment
PaymentMst


User
request to login
Response
Login
check detail
reply
UserMst
View Item
Request for view
reply
ItemMst
Add to cart
Update password
reply
OrderMst
Confirm Order
Edit Profile
reply
OrderMst

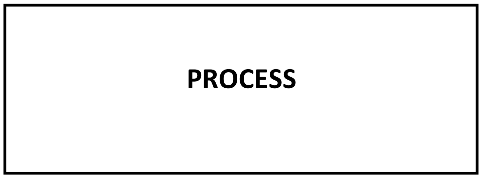
5.What is Flow chart? Create a flowchart to make addition of two numbers

- Flow chart is Used to show algorithm or process.
- It Can give step solution to the problem.
- Flowcharts are generally drawn in the early stages of formulating computer solutions.
- Flowcharts facilitate communication between programmers and business people.
- These flowcharts play a vital role in the programming of a problem and are quite helpful in understanding the logic of complicated and lengthy problems.
- Once the flowchart is drawn, it becomes easy to write the program in any high-level language.
- Often, we see how flowcharts are helpful in explaining the program to others. Hence, it is correct to say that a flowchart is a must for the better documentation of a complex program.
- **FLOW CHART SYMBOLS:**

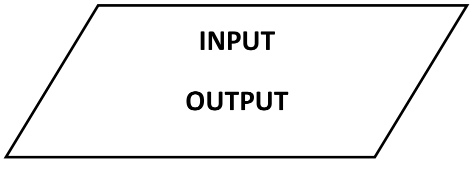
1. **Start Or End:**
    - Show starting or ending of any flow chart.
    - It can be Symbolized as:



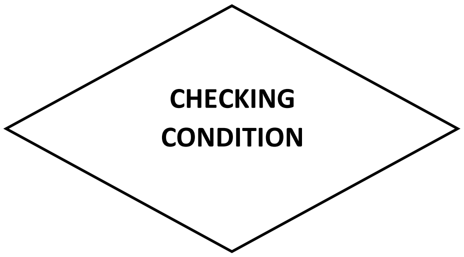
2. **Process:**
    - Defines a process like defining variables or initializing variable or performing any computation.
    - It can be Symbolized as



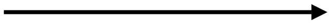
3. **Input or Output:**
    - Used when user have to get or initialize any variable Like get num1 and num2 from user.
    - It can be Symbolized as



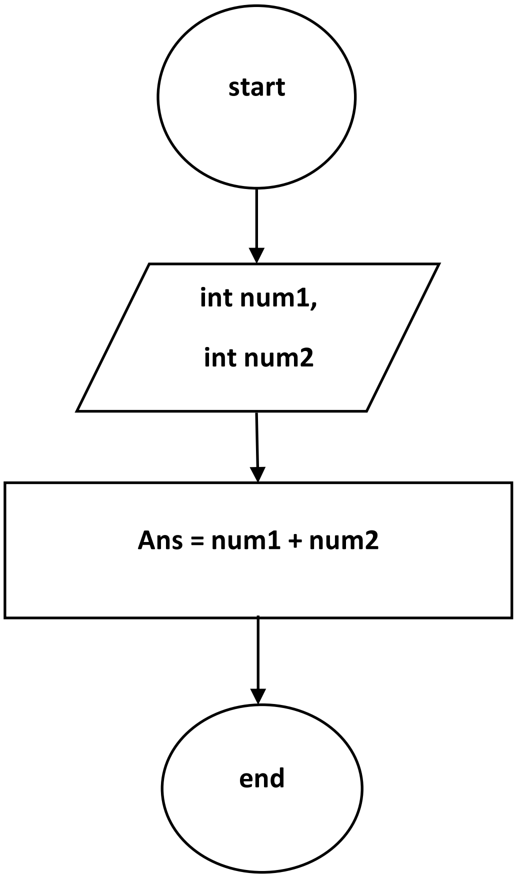
4. **Decision Making**
    - For checking condition this symbol can be used Like if num1 is greater than num2.
    - it Can be symbolized as



5. **Flowlines:**
    - Lines showing flow of data and process Showing flow of instructions.
    - It Can be symbolized as

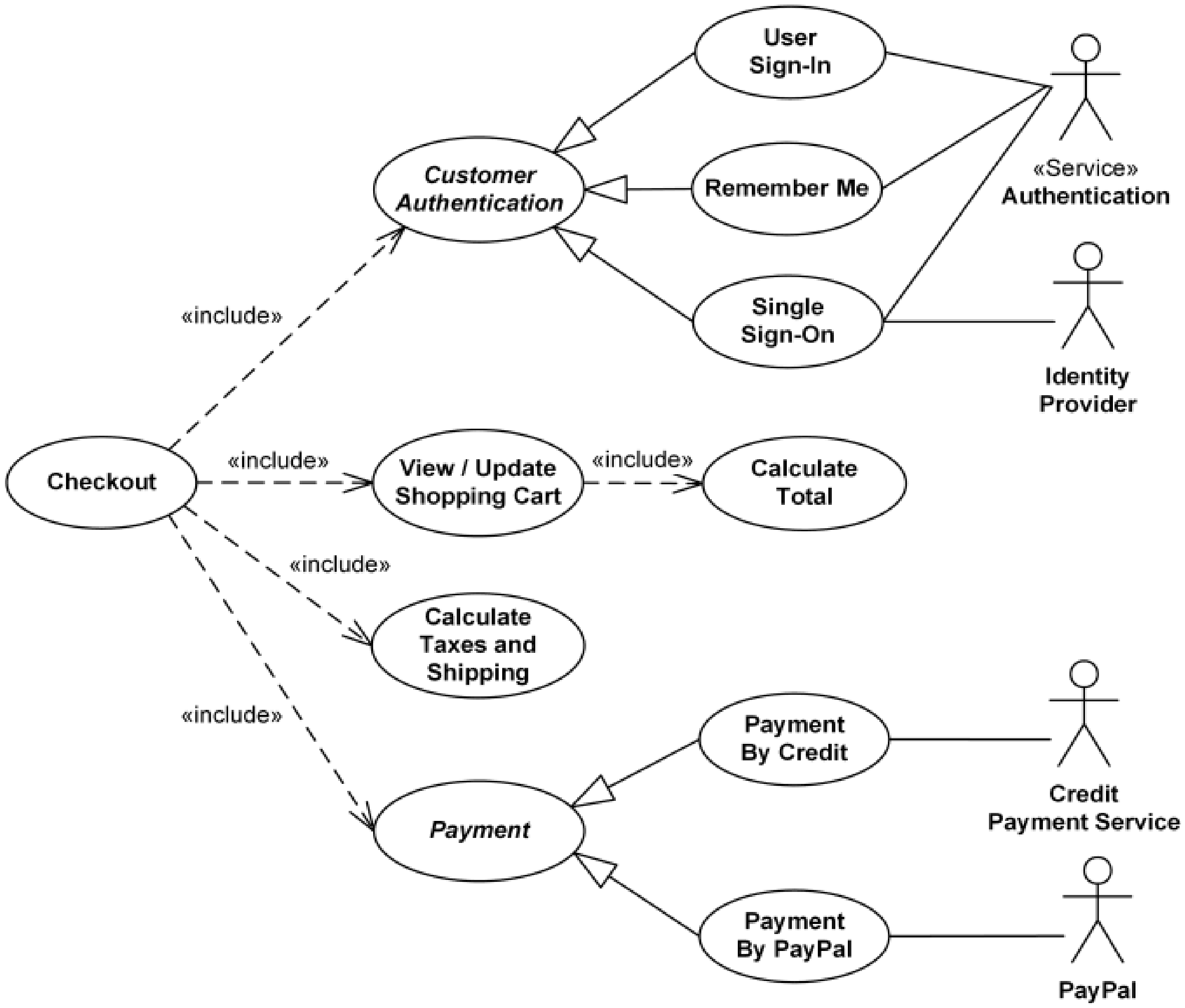


- **flowchart of two numbers addition:**



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

- 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.
- The main purpose of a use case diagram is to portray the dynamic aspect of a system.
- It gathers the system's needs.
- It depicts the external view of the system.
- It recognizes the internal as well as external factors that influence the system.
- It represents the interaction between the actors.


User Sign-In
Customer Authentication
Remember Me
«Service» Authentication
«include»
Single Sign-On
Identity Provider
Checkout
«include»
View / Update Shopping Cart
«include»
Calculate Total
«include»
Calculate Taxes and Shipping
«include»
Payment By Credit
Credit Payment Service
Payment
Payment By PayPal
PayPal