**NAME- RAJ PANDHI

**ASSIGNMENT- MODULE (1) SE: - OVERVIEW OF IT INDUSTRY.

**INSTITUTE- TOPS TECHNOLOGIES

Q.1- What is software? What is software engineering?

- Software is generally made up of data or computer instructions that give the computer how to function. It consists of programs, procedures and its documentation concerned with a machine or more specifically an automated data processing system. In other words, software is all the code running in a computer system.
- Software is essential for making the hardware functional.
- Software engineering is a discipline that uses systematic approaches to design, create and maintain software systems. It nested approach to make software reliable, efficient and meet the specified requirement. It involves different types of methodologies, tools and techniques which can be used to control the challenges faced due complexities in software projects while trying to make high-quality software within a given time frame with the allocated budget. These include requirements, software architecture. implementation and testing, and then deployment/ maintenance concerns.

Q.2- Explain types of software.

- The types of software's are as follows:

1. SYSTEM SOFTWARES: this software is designed to manage and control hardware components which helps to run applications. ex- Operating System (OS), Device Drivers and utilities for ex- antivirus software.
2. APPLICATION SOFTWARES: this software is designed for the users to do specific type of tasks or applications. Ex- Productivity software's like (word, excel, PowerPoint). Media players like (VLC media player, MX player). Web browsers like (google chrome, Microsoft edge, Firefox). And Games like (PUBG, candy crush, etc.)
3. DEVELOPMENT SOFTWARES: this software gives tools and environments for growing software's. Ex- Programming languages like (JAVA, C++, PYTHON). Integrated Development Environments (IDEs) this includes

code editors, debuggers, compilers, etc. like (VS Code, etc). And Version Control System like (git, GitHub, etc).

4. EMBEDDED SOFTWARES: this software's are written to manipulate machines or device which are not considered as computers. It is specialized for the unique hardware which runs on and has its own time and memory. Ex- Firmware- it can be said as a particular form of embedded software that offers a low-degree device manage for devices unique hardware. It can be found in routers and cameras.

This are some types of software's which can be used in a daily life.

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

- SDLC means Software Development Life Cycle, it is a scientific process for developing software program via a based collection of levels. It ensures that super software program is evolved, meets or exceeds client expectations, and is completed inside time and price estimates.
- Phases of SDLC:

1. Planning:

   - Objective: Define the task scope, goals, and feasibility.
   - Activities: Identify resources, set up mission schedules, estimate costs, and expand a venture plan.
   - Outcome: Project constitution, feasibility look at, and mission plan.

2. Requirements Analysis:

   - Objective: Gather and examine commercial enterprise requirements.
   - Activities: Conduct stakeholder interviews, surveys, and record analysis to collect requirements. Create requirement specs.
   - Outcome: Requirement's specification record (RS), use case diagrams, and consumer stories.

3. Design:

   - Objective: Define the software program architecture and layout.

- Activities: Create device architecture, statistics models, interface designs, and distinct layout specifications.
- Outcome: System layout record, architectural layout, and database schema

4. Implementation (Coding):

- Objective: Transform design files into functional software program.
- Activities: Write code, carry out unit checking out, and integrate specific modules.
- Outcome: Source code, unit check reviews, and software builds.

5. Testing:

- Objective: Ensure the software is disorder-free and meets the specified requirements.
- Activities: Execute diverse testing sorts, such as unit testing, integration testing, machine testing, and person acceptance checking out (UAT).
- Outcome: Test plans, check instances, computer virus reports, and take a look at summary report.

6. Deployment:

- Objective: Deploy the software program to a production environment.
- Activities: Install and configure the software program, perform very last trying out, and ensure the gadget is prepared for use.
- Outcome: Deployed software program, installation courses, and deployment reviews.

7. Maintenance:

- Objective: Provide ongoing guide and improvements.
- Activities: Monitor the gadget for troubles, perform trojan horse fixes, and enforce improvements based totally on user remarks.
- Outcome: Updated software program versions, maintenance reviews, and enhancement files.

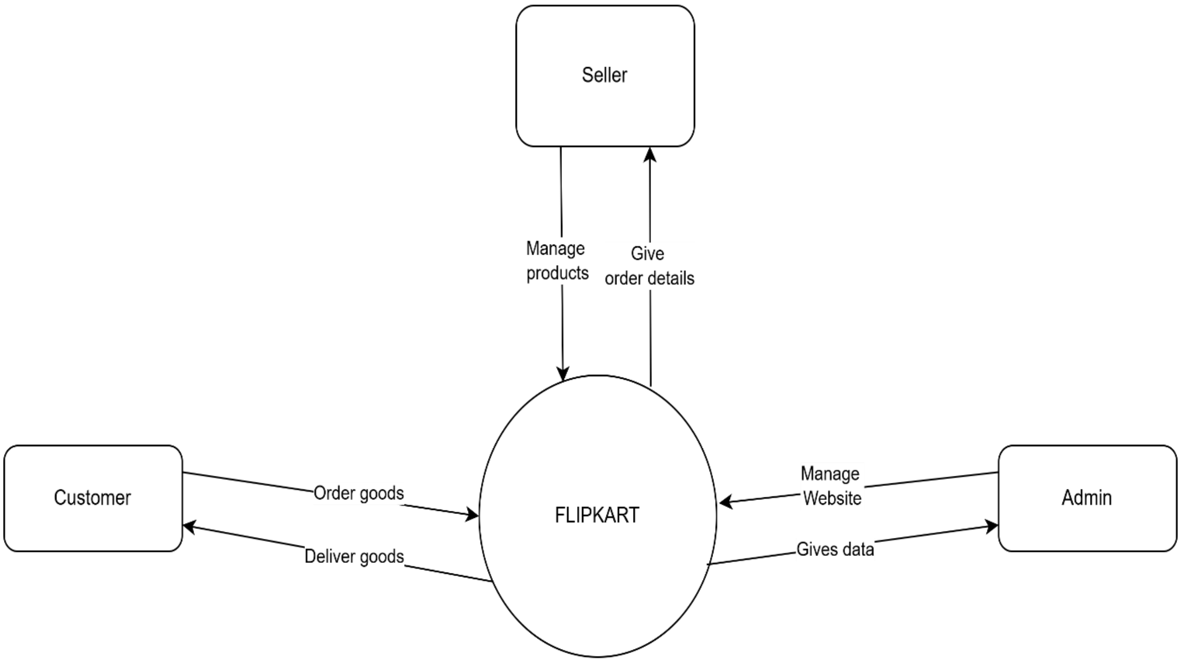
Q.4- What is DFD? Create a DFD diagram on Flipkart

- DFD means Data Flow Diagram, it is a graphical illustration of the float of statistics inside an app or website. It is used to visualize how the data moves via any device, inclusive of the inputs, tactics, storage, and outputs of information. DFDs assist in knowledge how data is processed and transformed in a machine, making it less difficult to pick out the components and interactions inside the device.
- There are different levels of DFD:

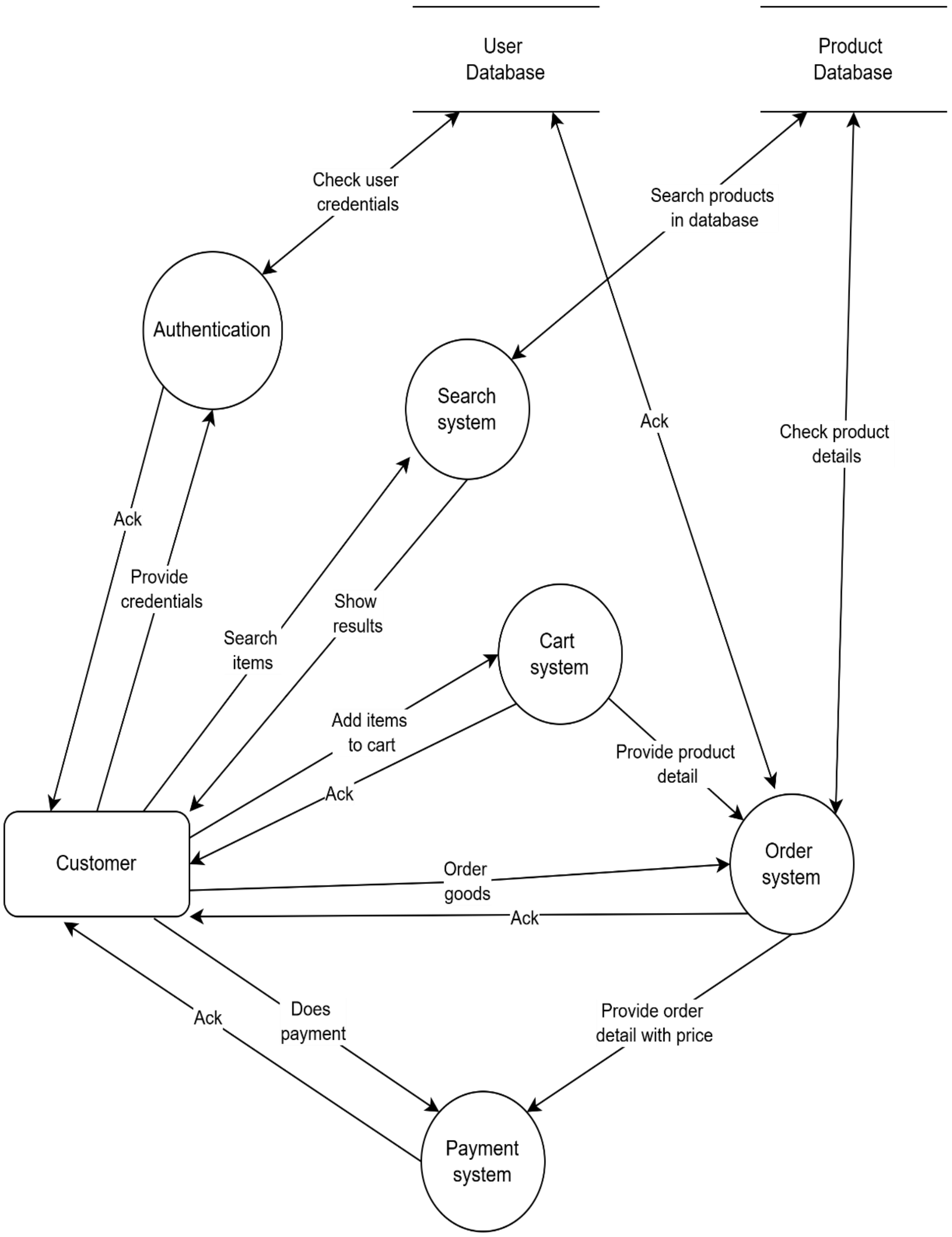
1. Level 0 DFD (Context Diagram): it Provides an excessive-level overview of the machine. It also shows the system as a single method with external entities and statistics flows. Also useful for knowledge the device's obstacles and interactions with external entities.
2. Level 1 DFD: Breaks down the hidden process from the Level zero DFD into sub-tactics. It also provides more detail about the essential methods and data flows inside the system. Also helps in expertise the primary practical regions of the machine.
3. Level 2 (and above) DFDs : it further decomposes the strategies from the Level 1 DFD into greater specified sub-techniques. It also provides an in-depth view of specific elements of the gadget. It is useful for detailed analysis and design of complex systems.

- DFD diagram of FLIPKART with all three levels example:

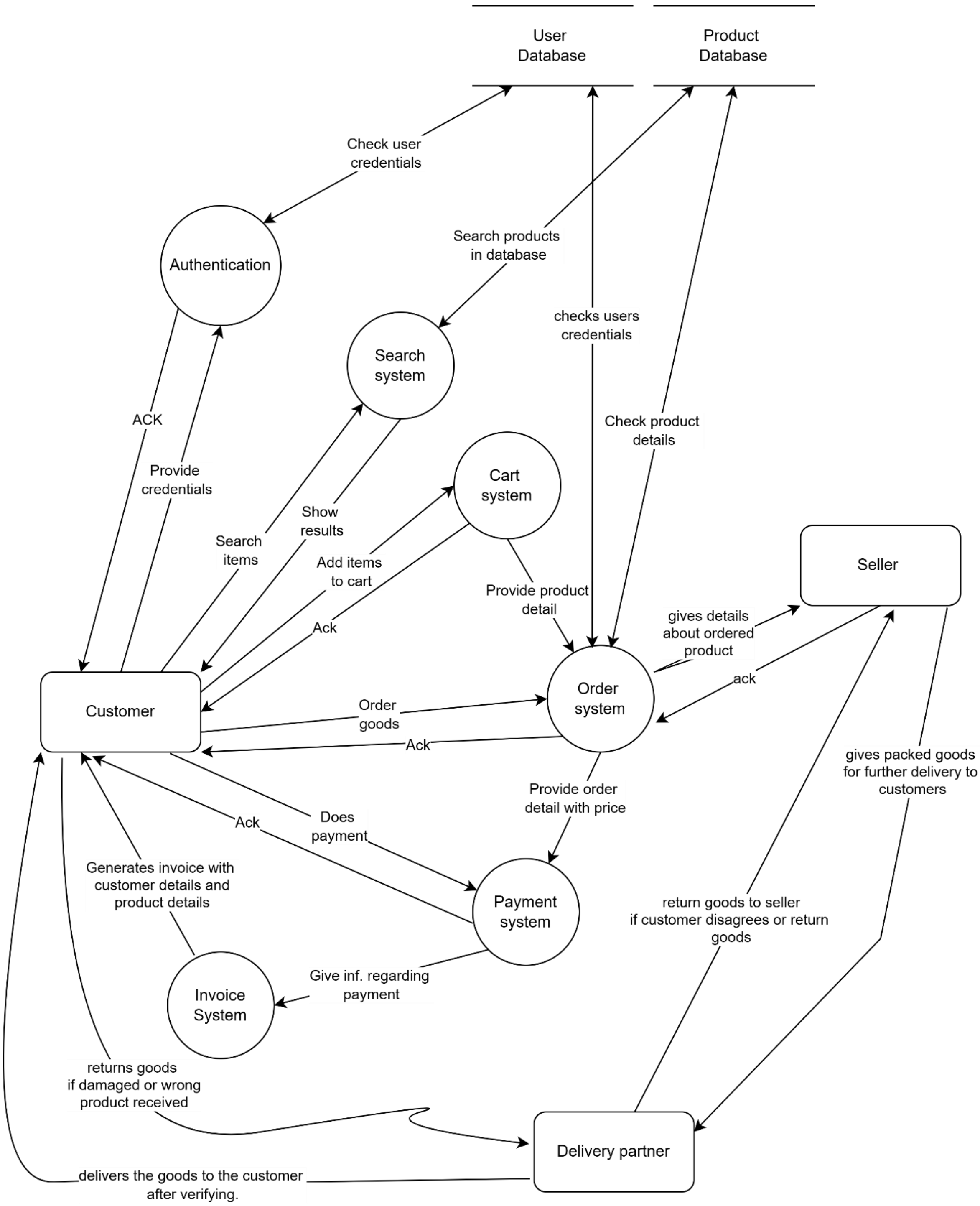
FLIPKART DFD-0 FLOW DIAGRAM



FLIPKART DFD-1 FLOW DIAGRAM



FLIPKART DFD-2 FLOW DIAGRAM



User Database
Product Database
Check user credentials
Authentication
Search products in database
Search system
checks users credentials
Check product details
ACK
Provide credentials
Cart system
Search items
Show results
Add items to cart
Ack
Provide product detail
Seller
gives details about ordered product
ack
Customer
Order goods
Order system
Ack
Provide order detail with price
gives packed goods for further delivery to customers
Ack
Does payment
Generates invoice with customer details and product details
Payment system
return goods to seller if customer disagrees or return goods
Give inf. regarding payment
Invoice System
returns goods if damaged or wrong product received
Delivery partner
delivers the goods to the customer after verifying.

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

- A flowchart is a diagrammatic representation of a process or workflow. It uses standardized symbols and arrows to show the sequence of steps and the flow of control in a process, making it easier to visualize, understand, and communicate how a particular task is performed. Flowcharts are commonly used in various fields, including computer programming, business process modelling, and project management, to illustrate procedures, algorithms, and systems.
- Key components of Flow Diagram:

1. Start/end:

    - Symbol- oval
    - Purpose- indicates the start or end points in process.

2. Process:

    - Symbol- rectangle
    - Purpose- indicates a step or action in the process.

3. Input/output:

    - Symbol- parallelogram
    - Purpose- indicates an input or output in the process.

4. Decision:

    - Symbol- diamond
    - Purpose- indicates a decision point where the flow can be branched based on yes/no or true/false.

5. Flow lines (arrows):

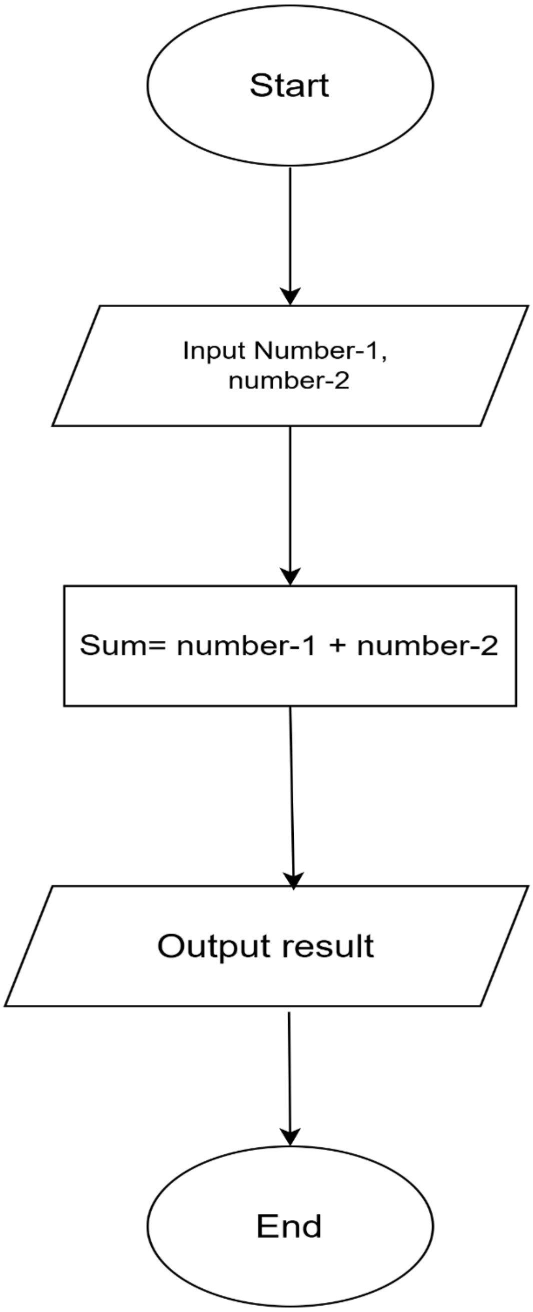
    - Symbol- arrow lines
    - Purpose- indicates to show the directions from one step to the next.

6. Connector:

- Symbol- circle

Purpose- used to connect the different parts of the flow chart. Especially when the diagram is of multiple pages.

- <u>Flowchart of additions of two numbers.</u>



<u>Q.6</u> What is Use case Diagram? Create a use-case on bill payment on Paytm.

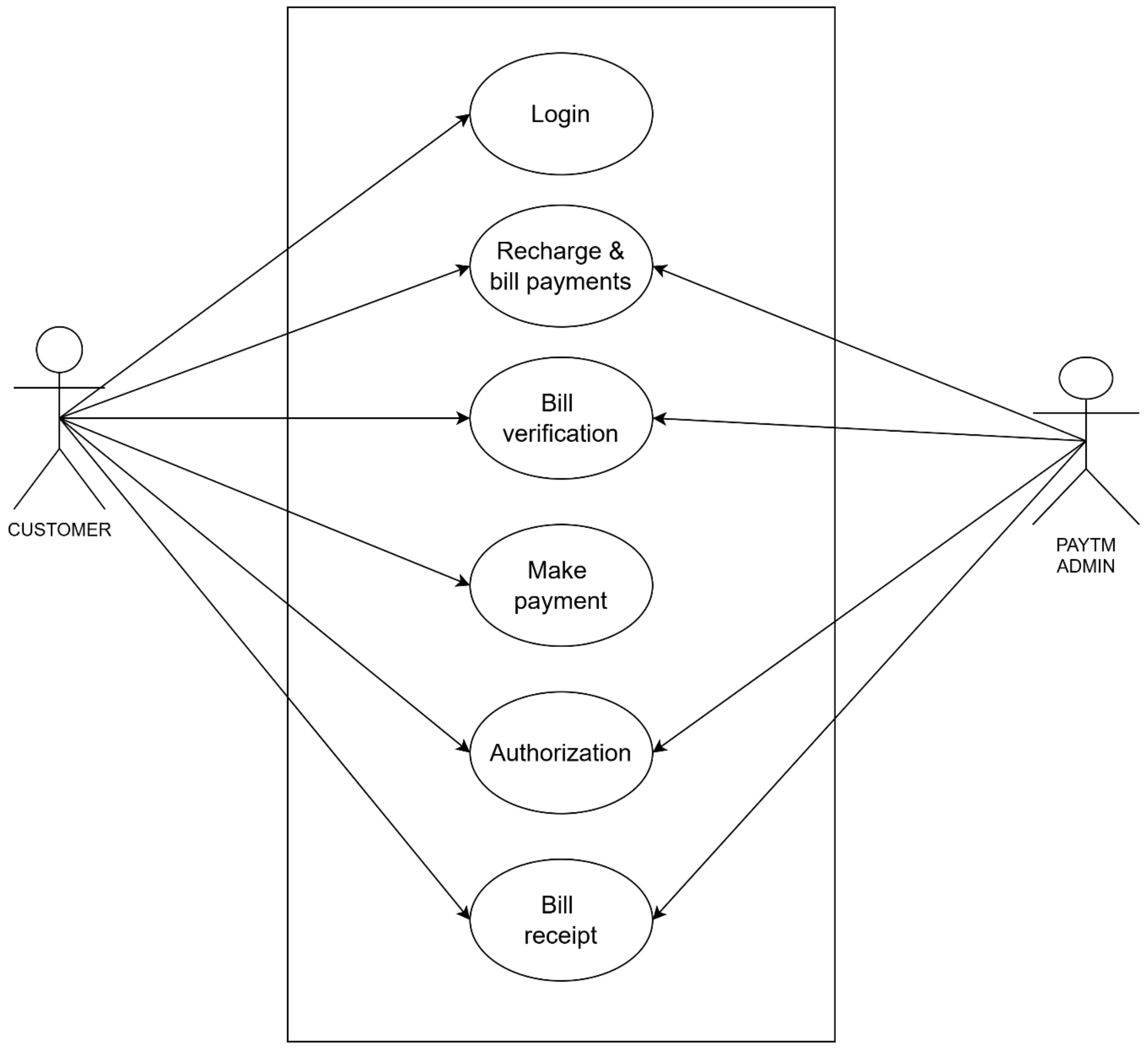
> A use case diagram is a visual representation of the functional requirements of a system. It shows the interactions between users (actors) and the system itself, depicting the different ways users can utilize the system to achieve specific goals. Use case diagrams are part of Unified Modelling Language (UML) and are commonly used in software engineering to capture and communicate the functional aspects of a system.

➔ Key components of a Use Case Diagram:

1. Actors- Represent the users or other systems that interact with the system being modelled. It is depicted as stick figures. It can be primary (initiating the interaction) or secondary (participating in the interaction).
2. Use Cases- Represent the actions or services the system provides to the actors. it is depicted as ovals. Also described with a brief verb-noun phrase, such as "Login," "Place Order," or "Generate Report.
3. System boundary- Represents the scope of the system. It is depicted as a rectangle enclosing the use cases. It also separates the internal functions of the system from external actors.

➔ Use case diagram on bill payment of Paytm:

BILL PAYMENT SYSTEM OF PAYTM


Login
Recharge & bill payments
Bill verification
Make payment
Authorization
Bill receipt
CUSTOMER
PAYTM ADMIN