Chittagong University of Engineering and Technology

Department of Computer Science and Engineering

Course ID: CSE-354

Course Name: System Analysis and Design (Sessional)

Organization: Bank Asia (Noapara Branch)

Group members:

Name	Student ID
Shahriar Hossain	2004069
Sha Newaz Mahmud	2004081
Shahriar Hasan	2004093
Md. Tahfim Juwel Chowdhury	2004094
Md. Minhazul Islam Minar	2004095
Md. Nur Siddik Ruman	2004098

Submitted to:

Dr. Kaushik Deb
Professor,
Dept. of CSE,CUET

Sabiha Anan
Assistant Professor,
Dept. of CSE,CUET

Remarks

Introduction:

A method for organizing and describing the structure and flow of data via a system's processes is called process modeling. Data flow diagrams, or DFDs, are the basic building blocks of a system analysis process model.

A data flow diagram (DFD) is a way of representing a flow of a data of a process or a system (usually an information system). The DFD also provides information about the outputs and inputs of each entity and the process itself.

Context diagram gives a broad overview at a glance. The successive expansion of a DFD from the context diagram to give more details is known as leveling of DFD. Data flow diagrams (DFDs) are categorized as either logical or physical.

Objectives:

- Analyzing the process
- To draw Logical and Physical Data Flow diagram
- Representing the data flow diagram in several levels

SDLC:

A method for planning, developing, testing, and deploying an information system is referred to as the systems development life cycle (SDLC), sometimes known as the application development lifecycle, in the fields of systems engineering, information systems, and software engineering. A system can be made up of only software, only hardware, or a combination of both, hence the systems development life-cycle idea is applicable to a variety of hardware and software combinations.

Systems engineers and systems developers employ a systems development life cycle, which is made up of a number of clearly defined and separate work phases, to plan for, design, build, test, and deliver information systems. An SDLC, like anything produced on an assembly line, attempts to develop systems that proceed through each clearly defined phase within scheduled timeframes and cost estimates. These systems should meet or exceed customer expectations and adhere to customer criteria.

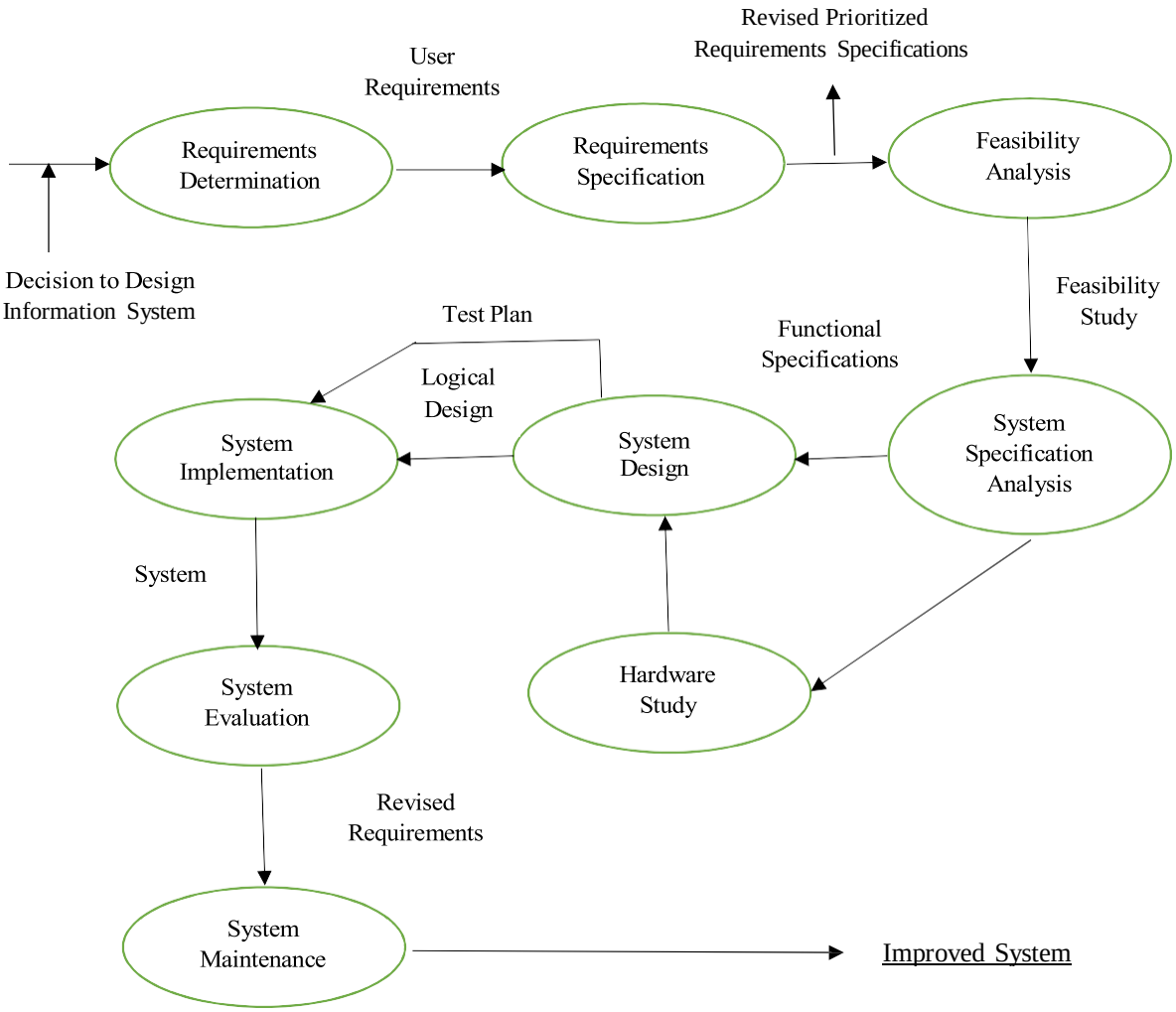

Figure 1: System Development Life Cycle

Context Diagram:

In this context diagram there has five external entity and one process. We have drawn the context diagram and DFD based on the information that we collected from the manager of Bank Asia which are given below:

- The general section is related to account opening, closing and transfer money.
- Customers deposit and withdraw money
- Loan section accepts loan requests and verifies information
- Transaction section is used to accept cheque/money and return money/receipt
- Manager supervise and control whole system
- Customer care service given by the Chatbots and Information desk attendant.
- Deposit and transfer money from banking app to mobile banking services (Bkash, Nagad, Rocket etc)

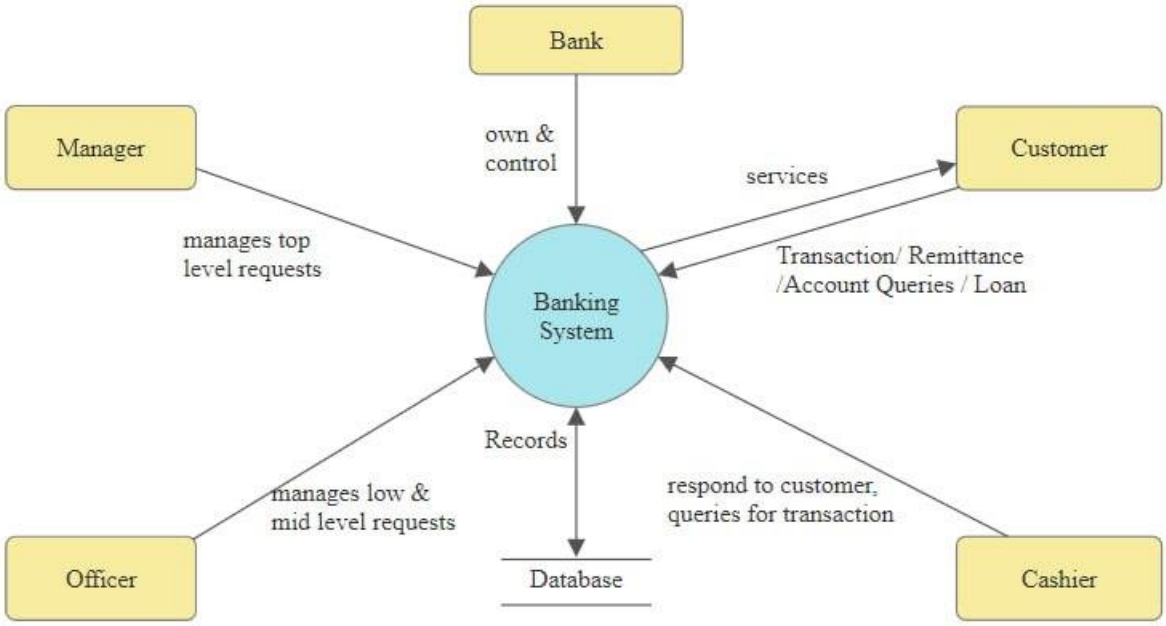

Figure-2: Context Diagram

Data Flow Diagram (DFD):

A Data Flow Diagram (DFD) is a graphical representation of the flow of data within a system. It is used to visualize how data is processed by a system in terms of inputs and outputs. DFDs help in understanding the functioning of a system and identifying the flow of data between different processes, data stores, and external entities. They are widely used in the analysis and design phase of system development to depict the boundaries of a system and the data interactions within it.

Four Components of a Data Flow Diagram:

1. **Processes:**
    - **Definition:** Processes represent the actions or functions that transform incoming data flows into outgoing data flows.
    - **Notation:** They are typically depicted as circles or rounded rectangles.

2. **Data Flows:**
    - **Definition:** Data flows illustrate the movement of data between processes, data stores, and external entities.
    - **Notation:** They are shown as arrows, with the arrowhead indicating the direction of data movement.
3. **Data Stores:**
    - **Definition:** Data stores represent places where data is stored within the system.
    - **Notation:** They are depicted as open-ended rectangles or parallel lines.
4. **External Entities:**
    - **Definition:** External entities are outside systems, organizations, or individuals that interact with the system by sending or receiving data.
    - **Notation:** They are usually represented as rectangles.

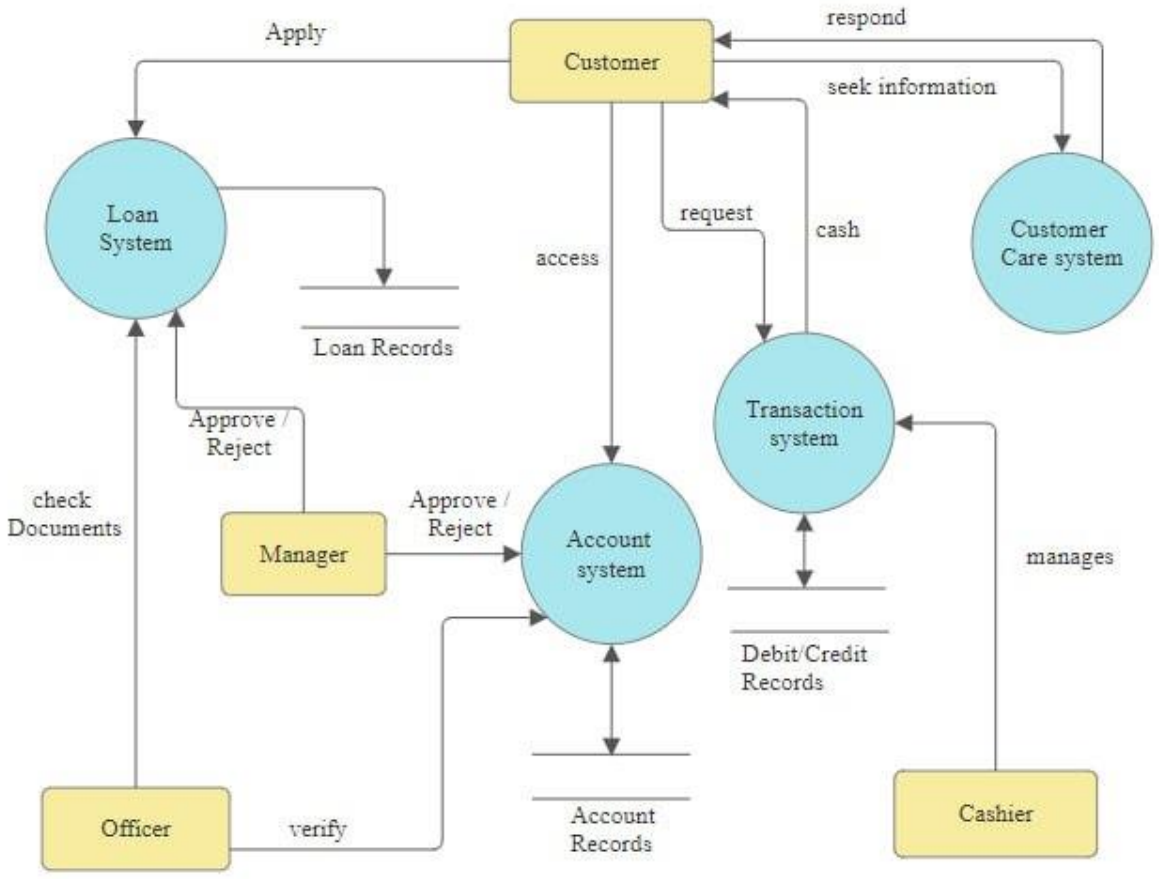

Figure-3: DFD on Banking System of Bank Asia

There are two types of Data Flow Diagram:

Logical DFD:

A Logical Data Flow Diagram (DFD) focuses on the business and its activities. While any data flow diagram maps out the flow of information for a process or system, a logical DFD provides a higher-level view that illustrates what the system must do to support the business functions. It emphasizes the business operations, the information that flows between them, and the relationships between different entities within the business. Logical DFDs are used during the analysis phase of system development to capture and communicate the requirements of a system without being concerned with how these requirements will be implemented technically. They help in understanding the business processes and the data that move through them, identifying the essential elements and interactions of the system, and defining the information requirements that support the business processes.

Physical DFD:

A Physical Data Flow Diagram (DFD) focuses on how a system is implemented. While a logical DFD provides a high-level view of the business processes and information flow, a physical DFD provides a detailed view of how these processes will be technically implemented. It includes details such as hardware, software, databases, manual procedures, and people involved in the system. The physical DFD maps out the actual physical implementation of the logical processes, showing how data is moved, stored, and processed in the real world. It includes specifics about the physical components of the system, such as file names, hardware devices, and actual locations of data stores. Physical DFDs are used during the design and implementation phases of system development to guide the technical implementation of the system, ensuring that the logical requirements identified in the logical DFD are met.

Physical DFD of Transaction Management System:

A Physical DFD of a Transaction Management System focuses on the actual implementation details of the system that manages transactions. This includes specific information about hardware, software, data storage, and data flow between the different components involved in the transaction processes.

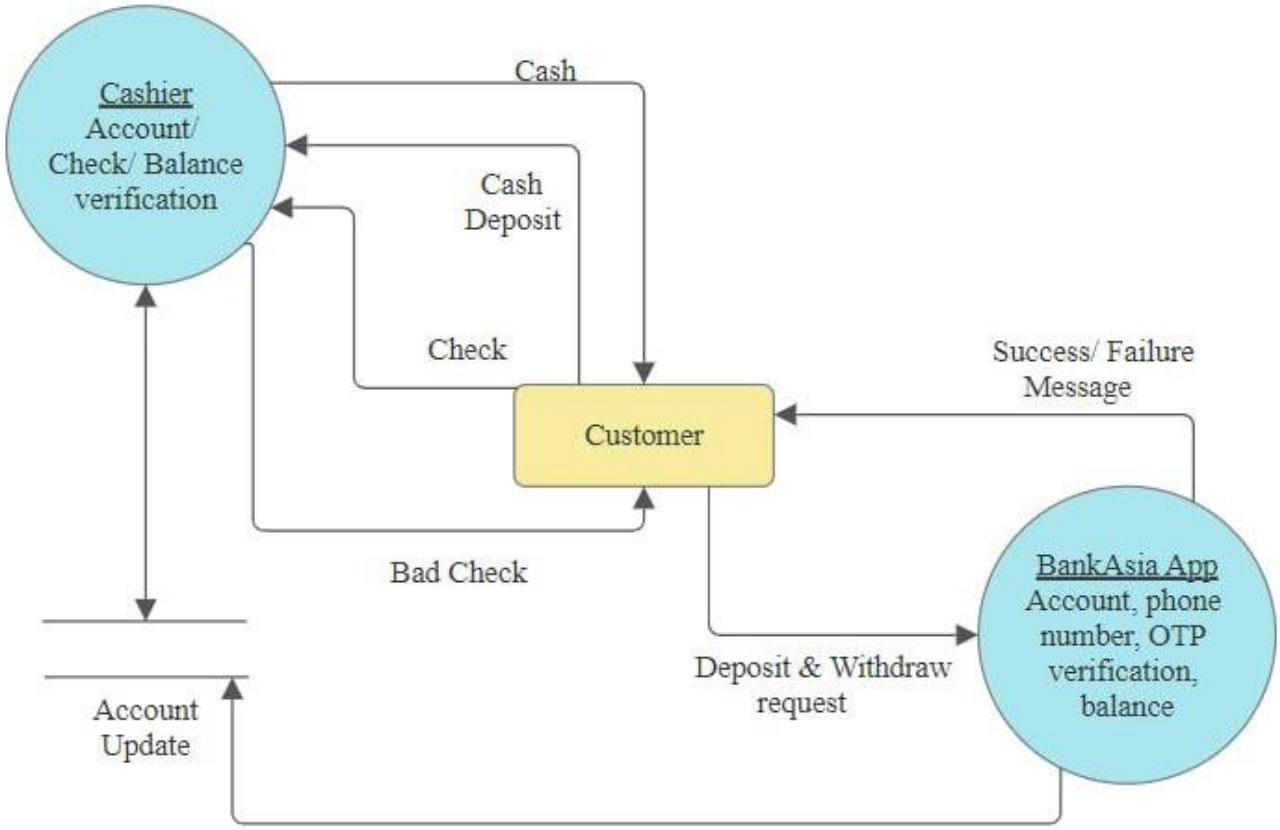

Figure-4: Physical DFD of Transaction Management System

Logical DFD of Transaction Management System:

A Logical Data Flow Diagram (DFD) of a Transaction Management System provides a high-level view of the business processes involved in managing transactions, without delving into the technical implementation details. It focuses on what the system must do to handle transactions effectively, highlighting the flow of information.

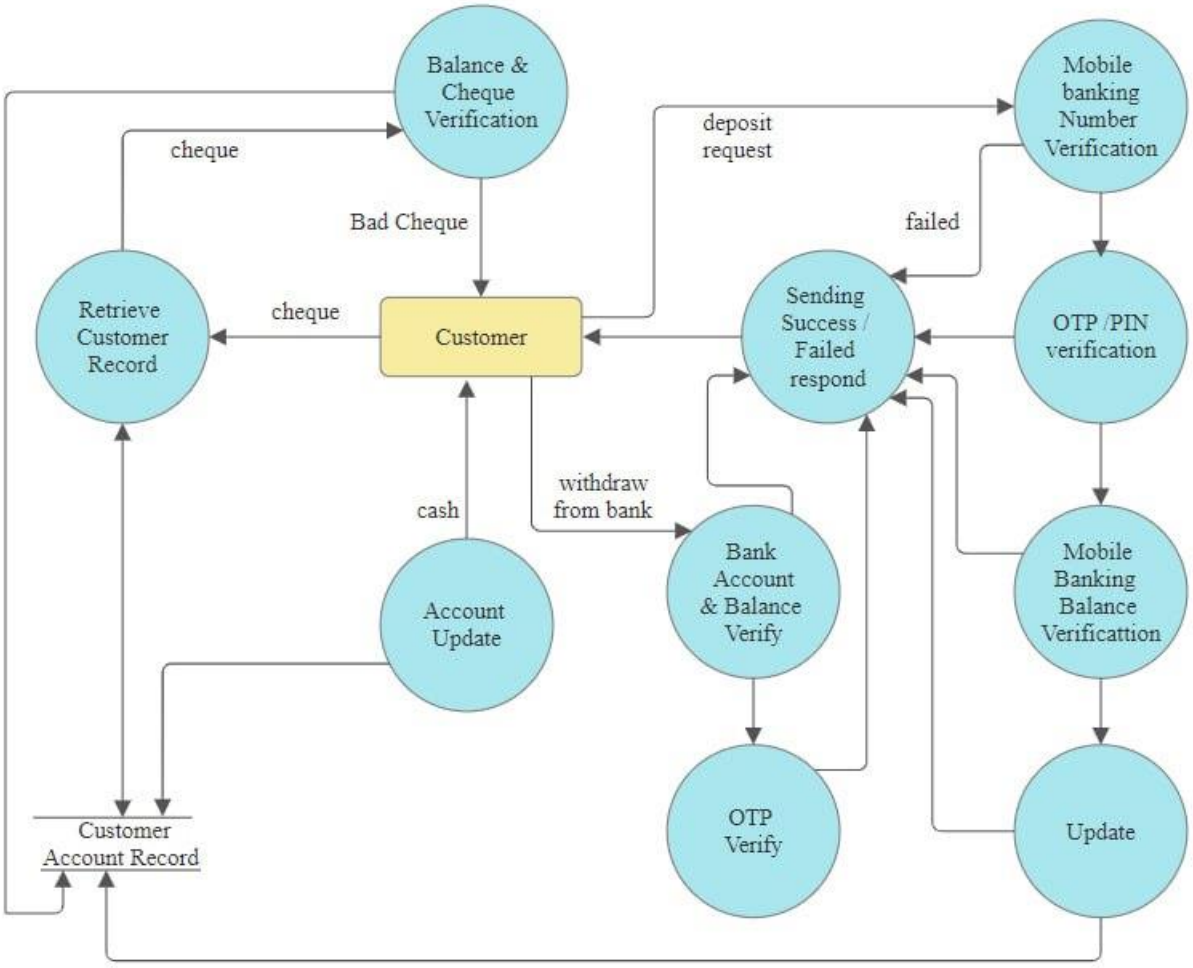

Figure-5: Logical DFD of Transaction Management System

Physical DFD of Customer Care Services:

This DFD outlines how information flows between the attendant, Chatbot, and customers in a customer care service system. It emphasizes both human and automated interactions.

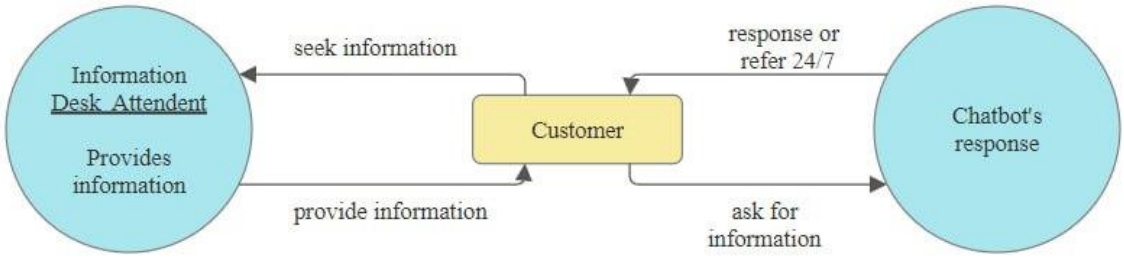

Figure-6: Physical DFD of Customer Care Services

Logical DFD of Customer Care Services:

This DFD illustrates how customer inquiries are handled, whether through automated or manual channels. It's a high-level view of the system's functionality.

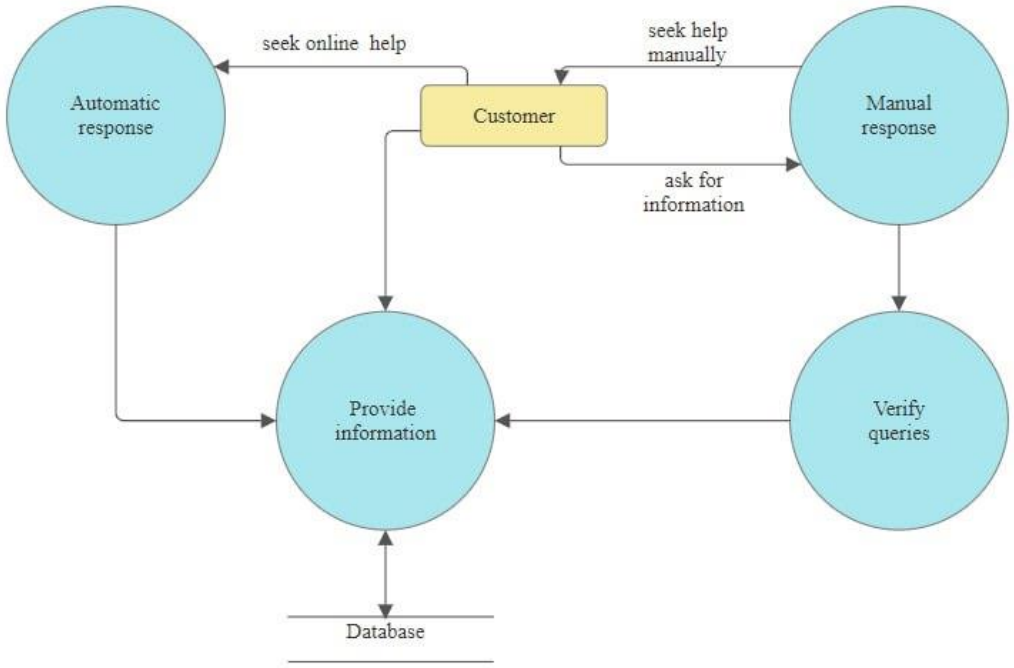

Figure-7: Logical DFD of Customer Care Services

Validating the Data Flow Diagram (DFD):

Data can flow from:

- External entity to process.
- Process to external entity.
- Process to store and back.
- Process to process.

Data cannot flow from:

- External entity to external entity.
- External entity to store.
- Store to external entity.
- Store to store.

Illegal constructs in DFD:

- No loops are allowed in DFD.
- A single data flow should not be split into many flows with different labels.
- No data flow allowed between data stores.

Considering the above rules, we verify our diagrams which are given below:

Context Diagram:

There are no data flows between the external entities in the context diagram, and there is no loop. So, the diagram is valid under the standards outlined above.

Logical and Physical DFDs:

There is no data flow from a database to an external entity or a database to an external entity. In every diagram, a single data flow is not divided into many flows with distinct labels. In these diagrams, there are no loops. Thus, these diagrams are accurate.

Conclusion:

Our study of Bank Asia's Noapara Branch data flow diagram offers insights into its operational framework, illustrating how information circulates within the bank. This analysis highlights the importance of systematic data flow management in enhancing banking processes and optimizing customer service delivery, contributing to scholarly understanding of organization.