MINI PROJECT
ON
BANKING MANAGEMENT SYSTEM

Contributors

Guided by:-

Mr. Anuj Kumar

Created by:-

Created By:[Varun Srivastav]
AFid: [AF04990614]
[Sunny Gupta]
AFid: [AF04991758]
Batch Code:[ANP-D2405] Course Code: [ITPR]

Table of Content

1. Title of Project

Banking management system

2. Introduction

The Banking Management System is a simple software solution that helps manage basic banking activities like creating accounts, depositing money, withdrawing money, and checking balances. It reduces manual work and makes daily banking tasks faster and more accurate.

This system stores all customer information safely in a database and provides quick access whenever needed. It improves efficiency, reduces errors, and makes the overall banking process easier for both customers and bank staff.

3. **Objective**

- To build a banking system that can store and manage customer account details using Java and JDBC.
- To allow users to perform basic banking tasks like deposit, withdrawal, and balance check.
- To connect the Java program to a database using JDBC for safe data storage.
- To make banking operations fast, easy, and accurate through automation.
- To ensure that all transactions are handled securely and updated in the database.
- To provide a simple interface for users to interact with the banking system.

4. Project category

1 Application:Type: Command line base.

2 Domain: Databse Management System.

3 Technology used:

- Java(Core Java+Swing)
- Mysql database
- Maven(build Tool)
- JDBC Connectivity

4 Project Level: Mini Project.

5 Purpose: To manage customer information,accounts,transaction safely and make banking work faster and reduce manual errors.

5. Analysis

1. Modules and Description

Write the main parts (modules) of your project and explain them in short.

a. Account Management Module

This module allows users to create a new bank account, update account details, and view account information.

b. Transaction Module

This module handles money deposits, withdrawals, and fund transfers between accounts.

c. Balance Enquiry Module

This module helps the user check the current balance of their bank account.

d. Customer Management Module

This module stores customer details like name, phone number, address, and account number.

e. Login & Authentication Module

This module allows users/admin to log in using username and password for security.

2. **Database design**

Here you explain how your data will be stored in the database.

a. Customer Table

Field Name	Data Type	Description
customer_id	INT	Unique ID for each customer
name	VARCHAR	Full name of customer
phone	VARCHAR	Mobile number
address	VARCHAR	Customer address
email	VARCHAR	Email address

b. Account table

Stores information about the bank accounts created by customers.

Field Name	Data Type	Description
account_id	INT	Unique account number
customer_id	INT	Links account to a customer
account_type	VARCHAR	Savings/Current
Balance	DOUBLE	Currrent balance in the account Add a little bit of body text
Opening_Date	DATE	Date when account was created

c. Transaction Table

Stores details of all deposits, withdrawals, and transfers.

Field Name	Data Type	Description
transaction_id	INT	Unique transaction ID
account_id	INT	Links transaction to an account
type	VARCHAR	Deposit/Withdrawal/Transfer
amount	DOUBLE	Transaction amount
date	DATE	Date of transaction

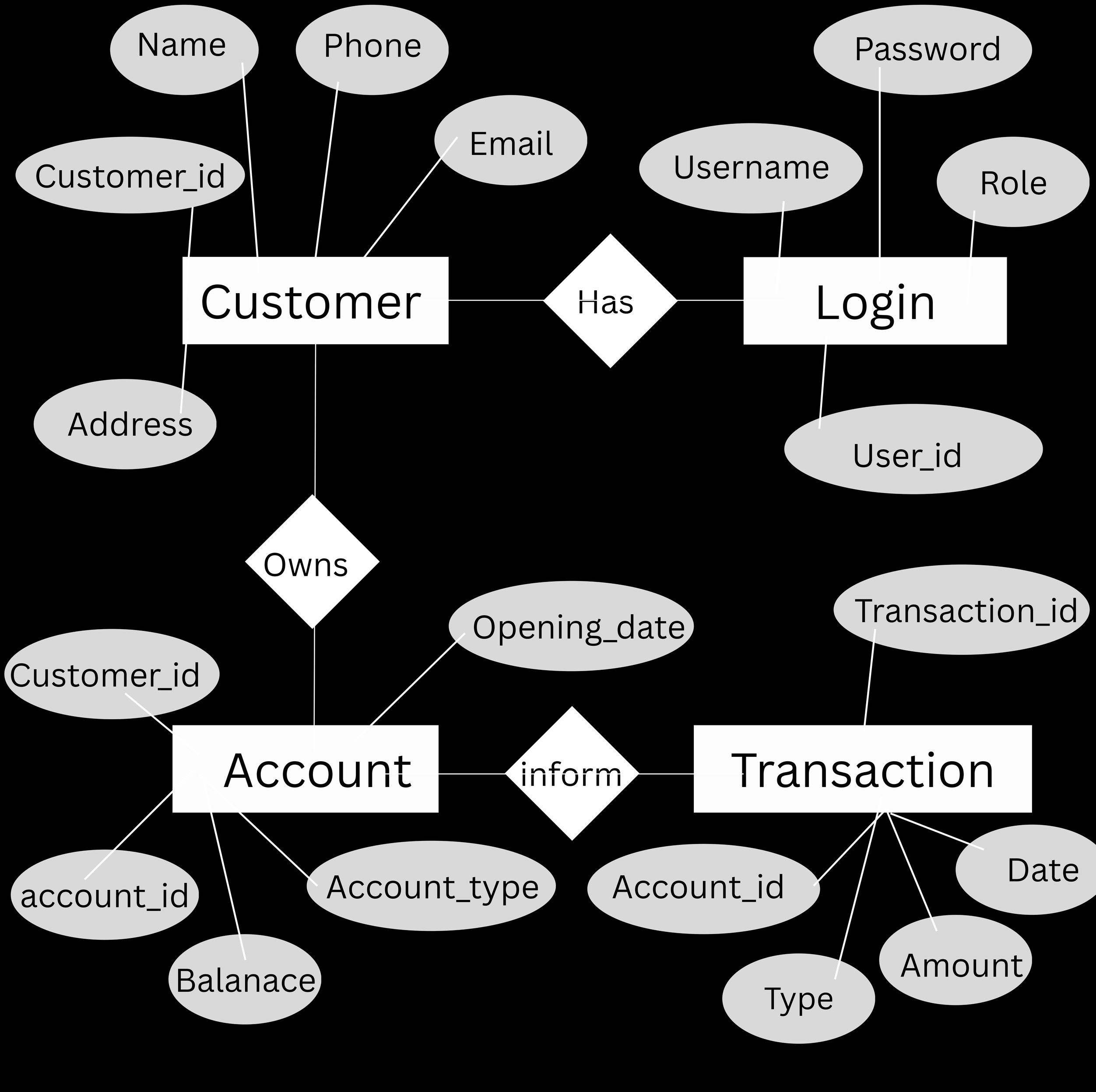
d. Login Table

Used for system authentication

Feild Name	Data type	Description
user_id	INT	Unique ID
username	VARCHAR	Login username
password	VARCHAR	Login password
role	VARCHAR	Admin or User

3.ER Diagram

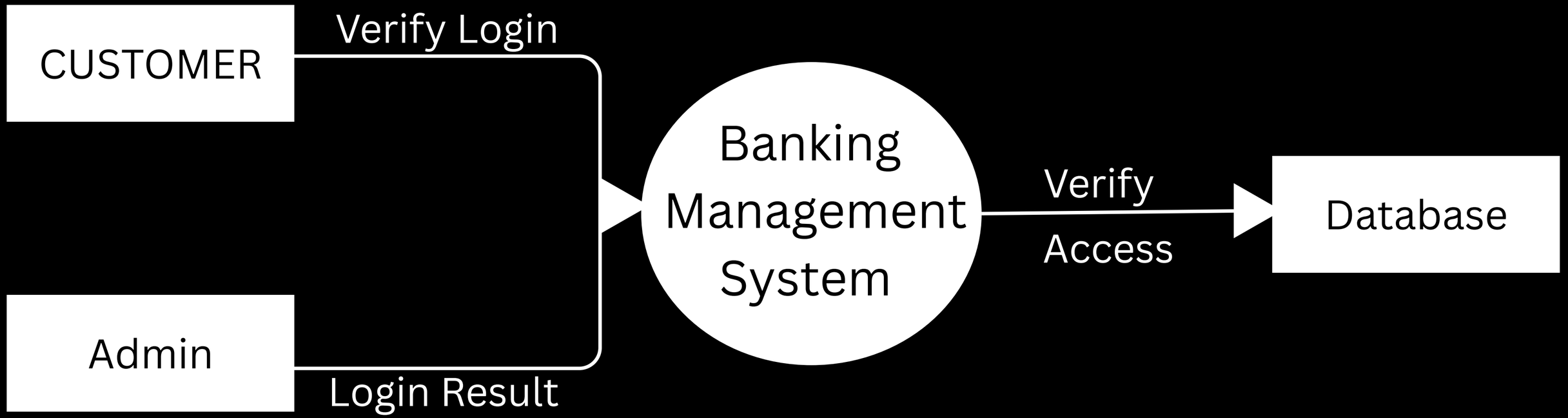
An ER diagram (Entity-Relationship diagram) is a simple picture that shows how different parts of a database are connected. It uses boxes for tables and lines for relationships, helping us understand the structure of data clearly and easily.


Name
Phone
Email
Customer_id
Customer
Has
Login
Password
Username
Role
Address
User_id
Owns
Opening_date
Customer_id
Account
inform
Transaction
Transaction_id
account_id
Account_type
Balanace
Account_id
Date
Amount
Type

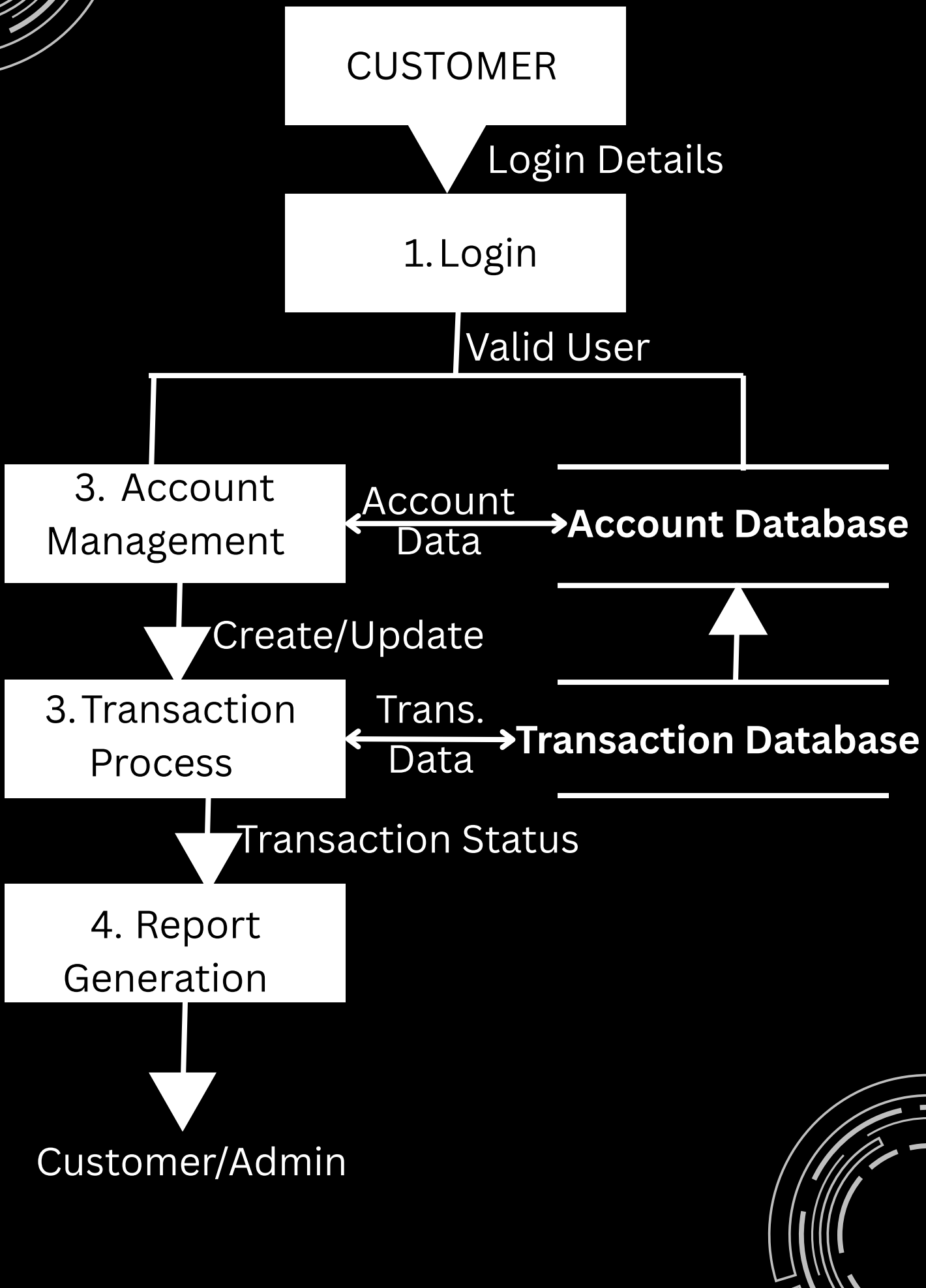
4. Data flow Diagram

A Data Flow Diagram (DFD) of a banking management system shows how information moves between customers, bank staff, accounts, and processes like deposits or withdrawals. It helps understand the system's workflow in a clear and easy way.

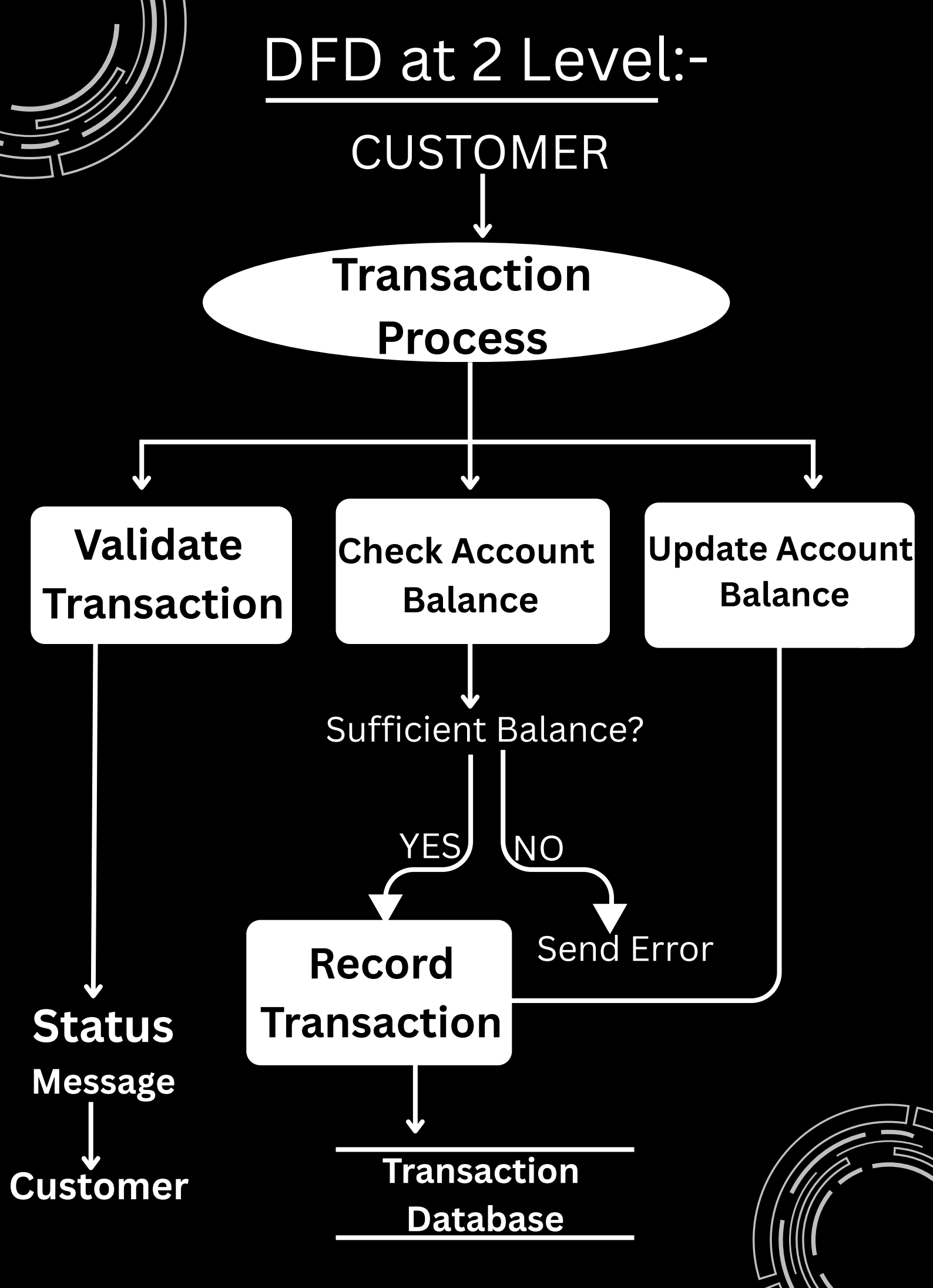
DFD at 0 level:-



DFD at 1 Level:-



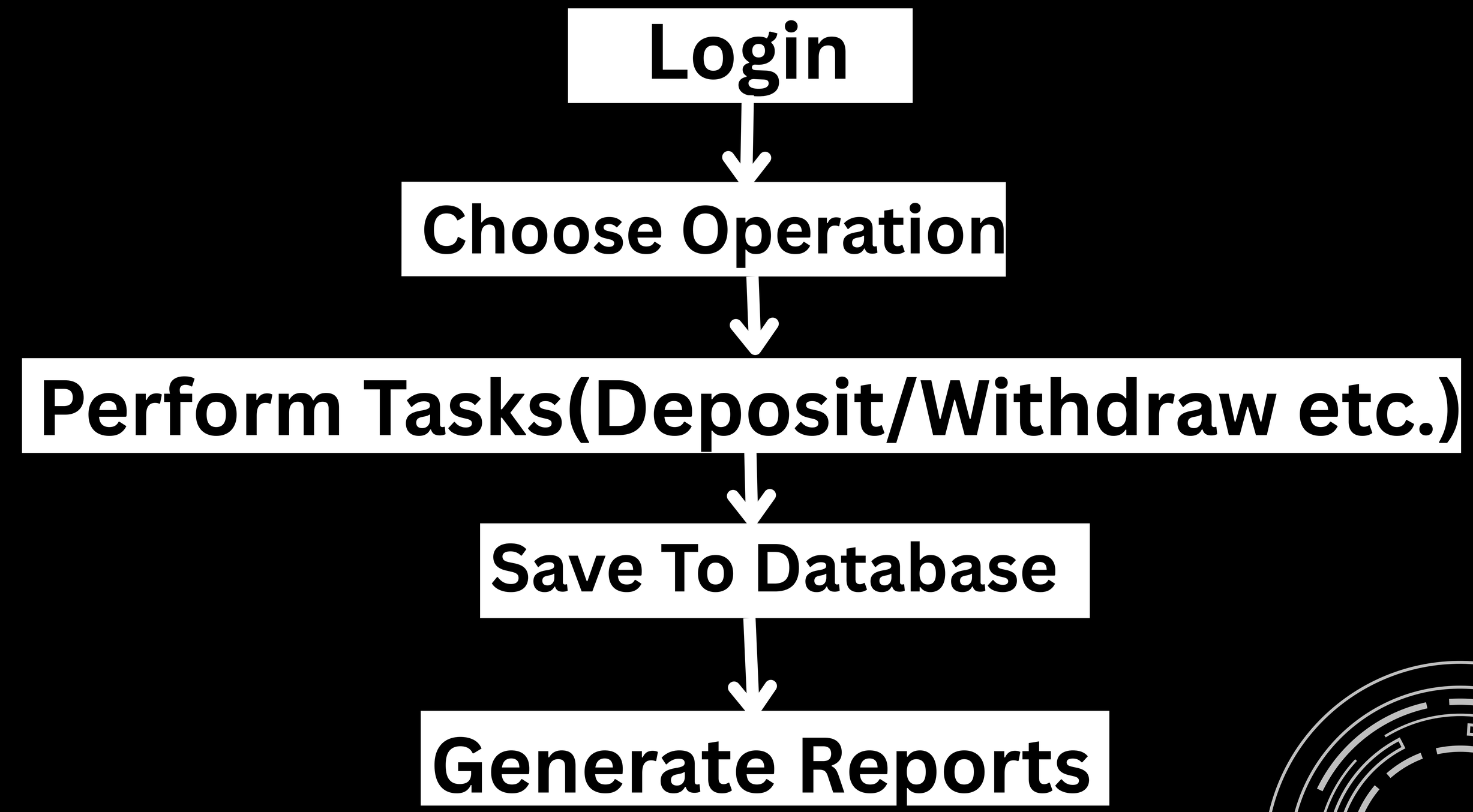
DFD at 2 Level:-



6. **Complete Structure**

Process Logical Diagram

Process Logical Diagram will visually represent the workflow of banking management
detailing between customer and admin withdraw,deposite,transfer money.



7. Platform Used

Hardware Requirements:-

- Processor:corei5
- RAM: 16GB
- Storage: Minimum 516 GB space
- Operating system: Windows 11

Software Requirements:-

1. Java Development Kit (JDK)

- The Java Development Kit provides all the tools required to develop and run Java applications.

- It includes the Java compiler (java c), Java Runtime Environment (JRE), and essential libraries.
- In this project, JDK is used to write and execute the backend logic, forms, and functionalities of the Banking Management System

Why used in this project?

✓ To develop the complete application using Core Java
✓ Provides libraries for OOP-based development

2. Intellij IDE

- Intellij IDE is a popular development environment used for writing, editing, and debugging Java code
- It offers features like auto-completion, error checking, project structure management, and plugin support.

Why used in this project?

✓ Makes Java coding easier with syntax support
✓ Integrated console and debugging tools
✓ Allows well-organized project structure
✓ Supports Maven and plugin extensions

3. Maven (Build Automation Tool)

• Maven is a dependency and project management tool. It automatically downloads required libraries (like MySQL Connector) and manages project versions and builds.

Why used in this project?

✓ Simplifies dependency management
✓ Easy project compilation and packaging
✓ Reduces manual configuration
✓ Creates a clean and structured Java project

4. MySQL Database

- MySQL is a relational database system used for storing all Banking data permanently.
- It stores books, students, issue records, and return records in tables and allows fast data retrieval.

Why used in this project?

✓ Secure and reliable data storage
✓ Supports complex queries
✓ Easy integration with Java
✓ Ensures data consistency and accuracy

5. MySQL Connector/J (JDBC Driver)

- MySQL Connector/J is the official JDBC driver used to connect Java applications to a MySQL database.
- It allows Java programs to execute SQL queries such as insert, update, delete, and fetch operations.

Why used in this project?

✓ Enables communication between Java program and MySQL database
✓ Sends queries and retrieves results
✓ Essential for performing book issue, return, and management operations

8. Future Scope

1. Mobile Banking Improvement

   More features can be added like faster money transfer, bill payments, and secure login.

2. Use of Artificial Intelligence (AI)

   The system can use AI for fraud detection, chatbots, and better customer support.

3. Advanced Security

   Stronger security features like fingerprint login, face detection, and OTP protection.

9. **Bibliography**

I. Intellij IDE – Used for writing, compiling, and running Java code during the development of the banking Management System project.

II. MySQL Database Server – Used for creating the database, tables, and storing all the records related to books, students, admin, and issue/return operations.

III. Google Chrome Browser – Used for searching programming concepts, syntax clarification, documentation, and online references during project development.

IV. Trainer Guidance – Project concepts, database understanding, and coding guidance were provided with the help of the trainer.

V. ChatGPT (OpenAI) – Used for additional explanations, code corrections, debugging help, and better understanding of Java, JDBC, and SQL conceptst

THANK YOU