National University of Computer and Emerging Sciences

Database Systems Lab Project Report

Bank Management System

Group Members:

Name	Roll Number	Section
Mannan Ul Haq (LEAD)	22L-7556	BDS-4A2
Huzaifa Amir	22L-7518	BDS-4A1
M. Hassan Mustansar	22L-7521	BDS-4A2
M. Abdullah Waseem	22L-7522	BDS-4A2

Database Systems Lab

Spring 2024

Department of Data Science

FAST-NU, Lahore, Pakistan

1. Introduction:

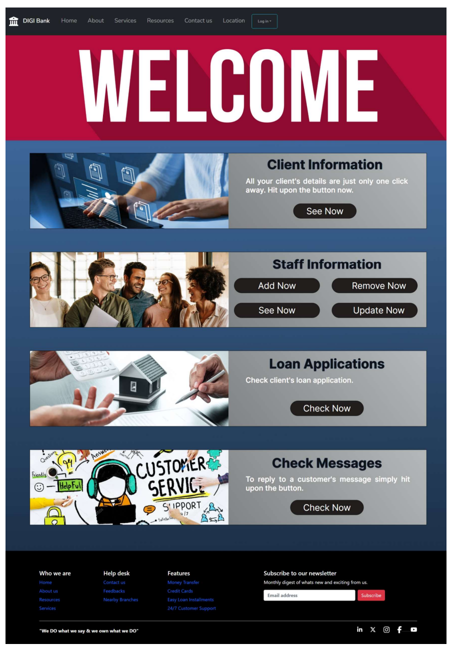
Title: Bank Management System (Website Application)

Application Name: DigiBank

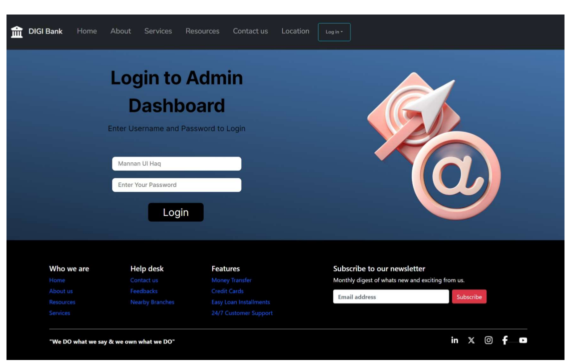
A comprehensive web application for bank management developed for the Database Systems Lab at FAST-NU, Lahore. DigiBank integrates advanced security measures including facial recognition for user authentication and provides a robust interface for managing customer accounts, transactions, loans, and credit cards. The scope of the project includes developing a user-friendly interface with robust features tailored to both customers and bank staff.

2. Main Features:

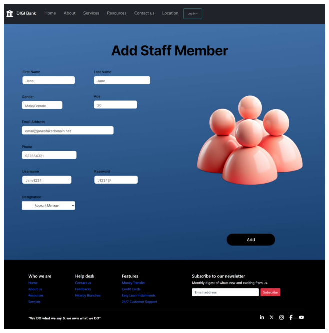
➢ Admin Interface:

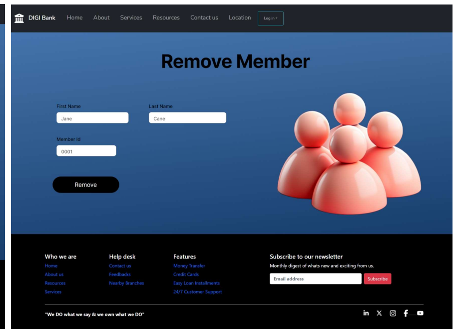


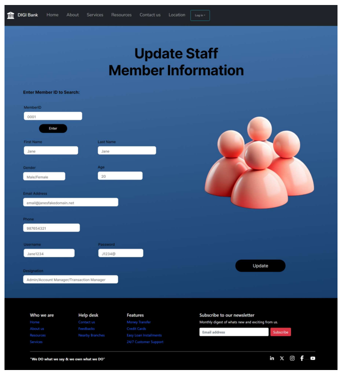
i. Secure Password Login: Ensure secure access for the admin.

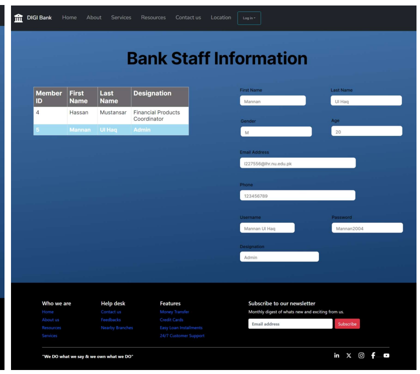


ii. Bank Staff Management: View, manage, and assign roles to bank staff.

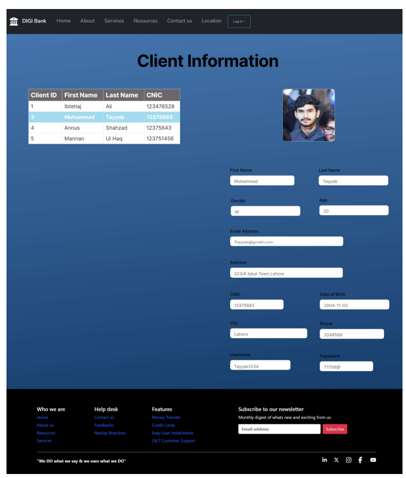




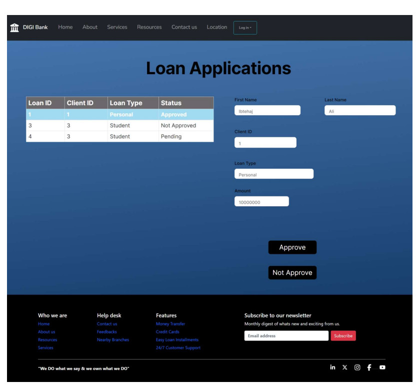




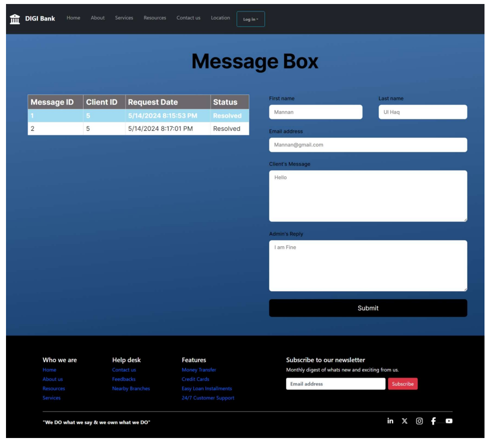
iii. Client Data Viewing: Access and review clients' data.



iv. Loans Management: Accept or reject loan applications.

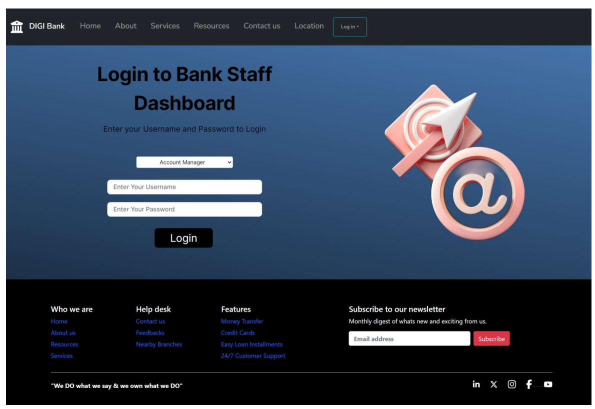


v. Customer Queries: Respond to customer queries and questions via the integrated message box.

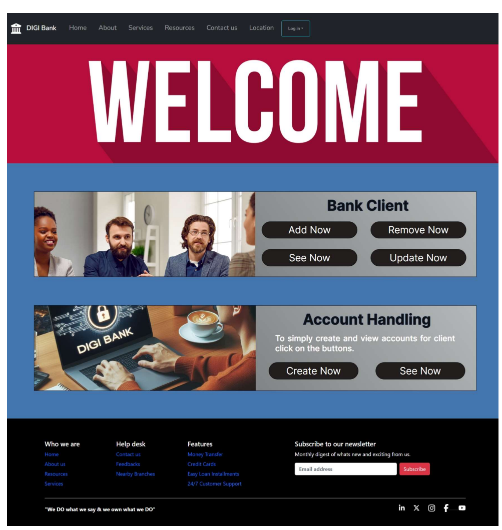


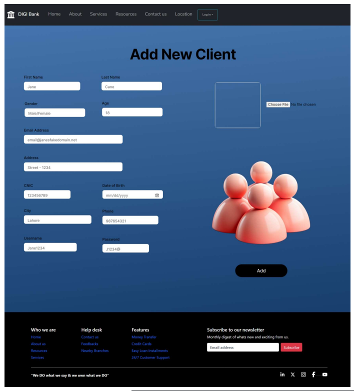
➢ Bank Staff Interface:

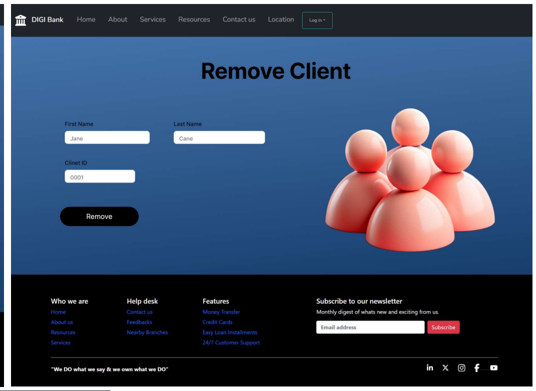
i. Secure Password Login: Ensure secure access for bank staff.

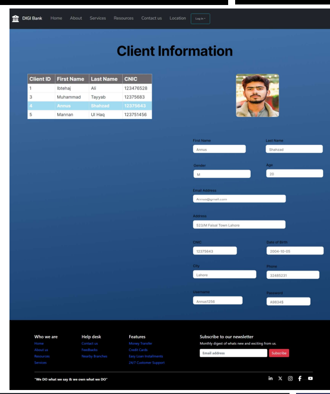


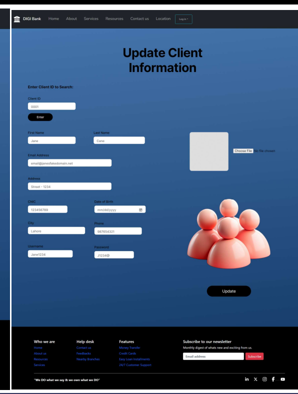
ii. Account Manager: Enter, verify, and update client data and account details.

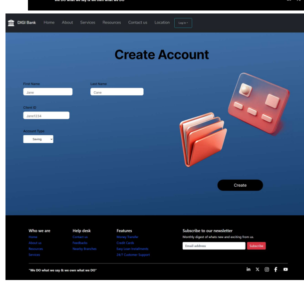


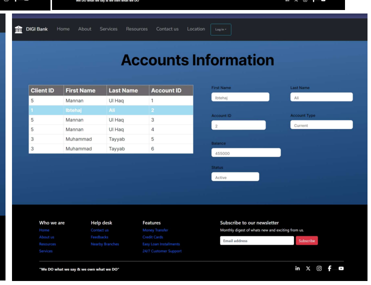




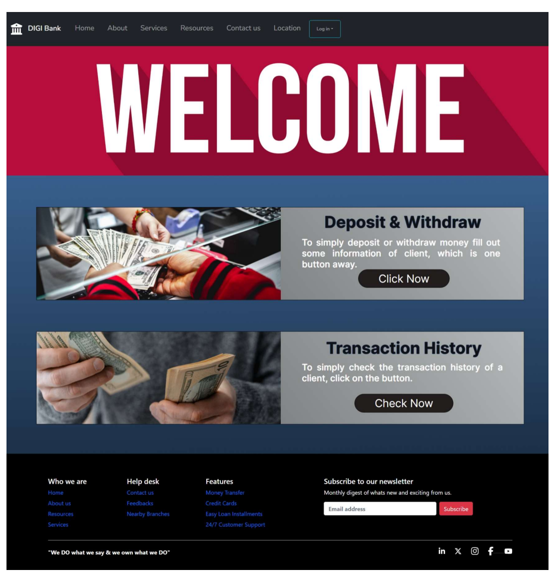


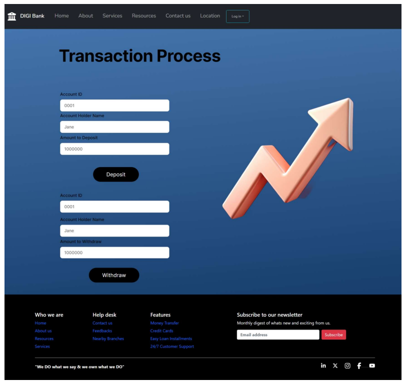


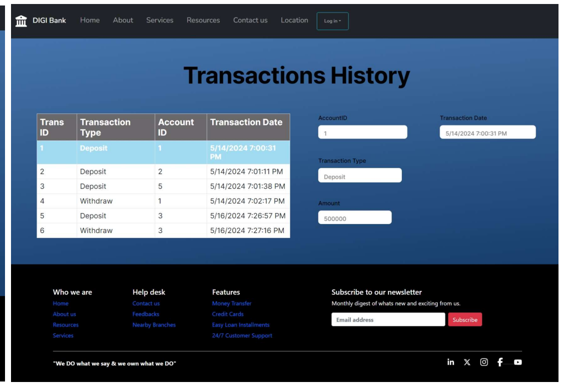




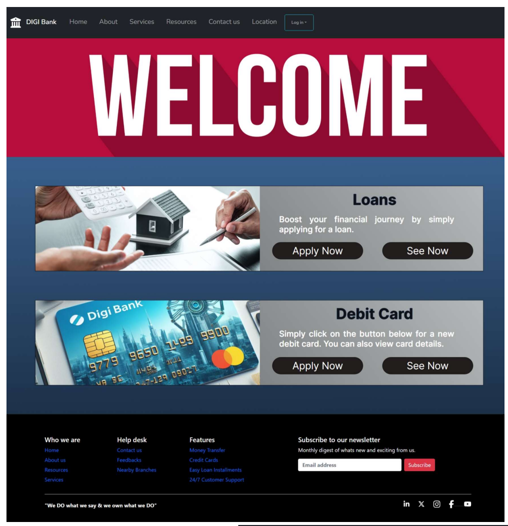
iii. Transaction Manager: Efficiently manage client transaction records.

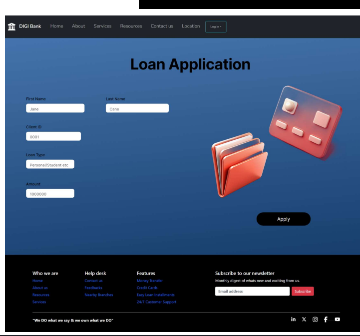


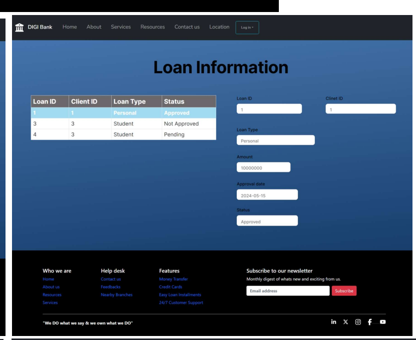


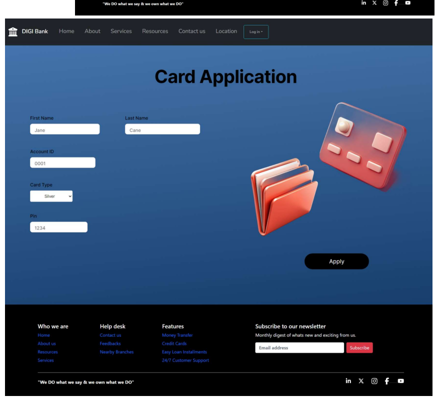


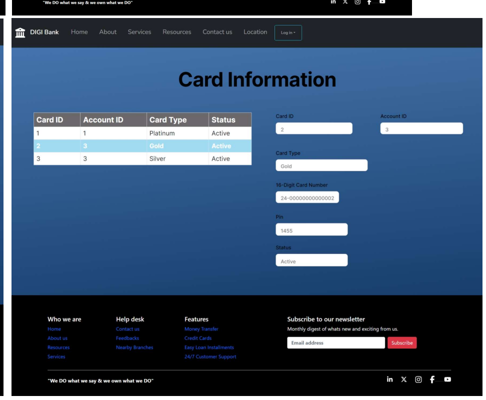
iv. Financial Products Coordinator: Review, process, manage, and track client loans and client credit card requests.



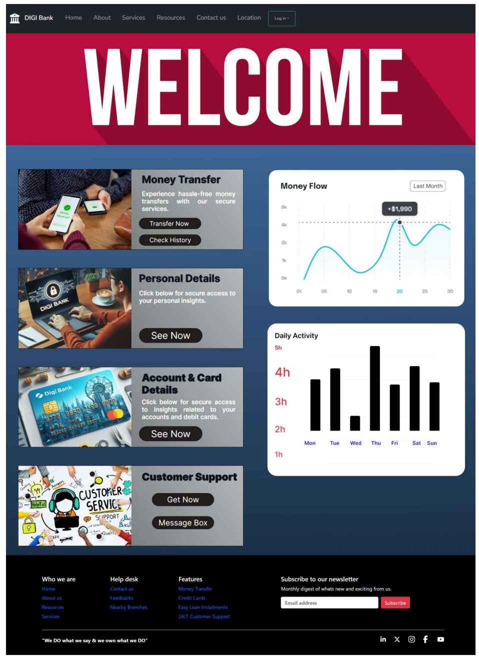




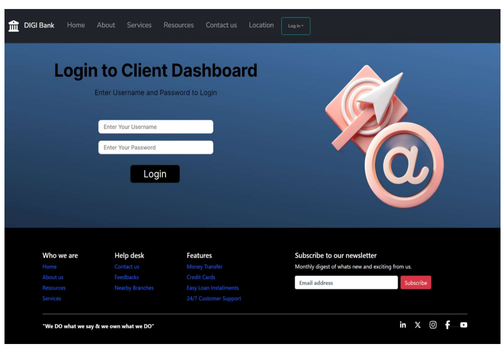


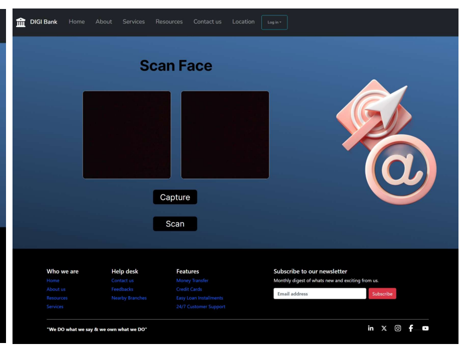


➢ Client Interface:

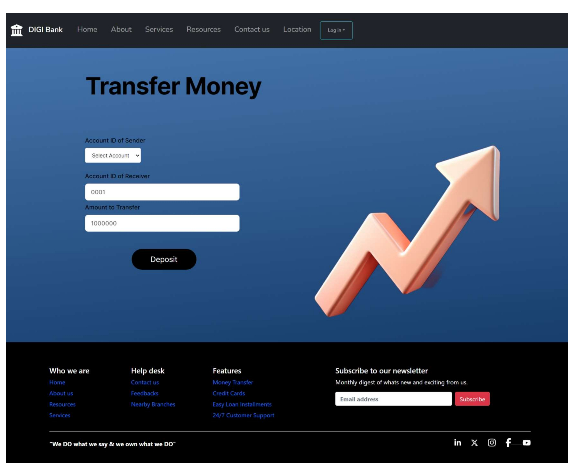


i. Facial Recognition Login: Secure login using facial recognition combined with a password.

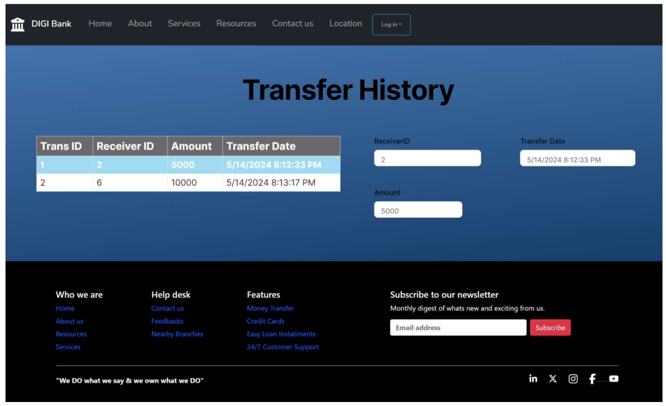




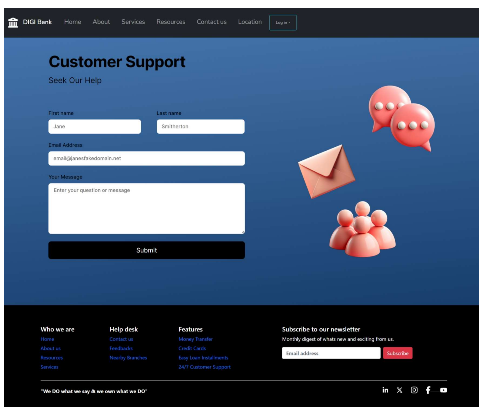
ii. Money Transfer: Perform money transfers.

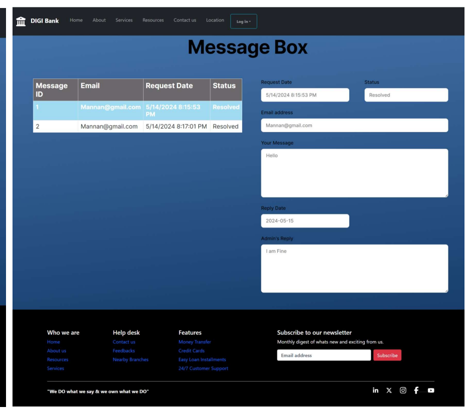


iii. Transfer History: Access detailed transfer history.

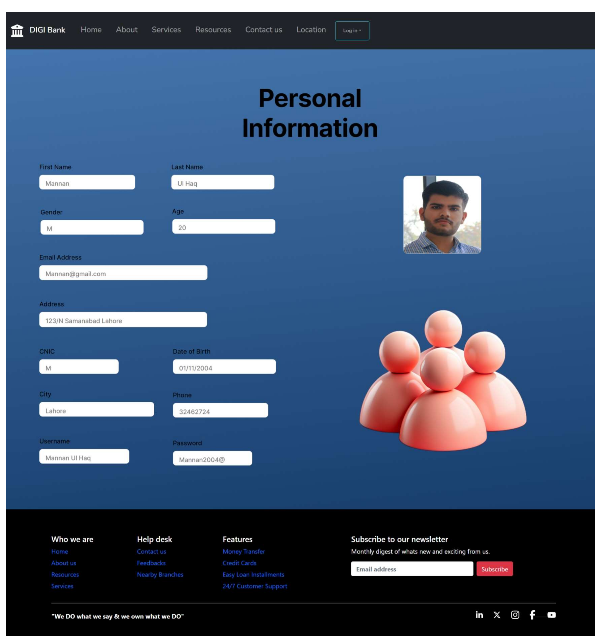


iv. Customer Support: Contact bank representatives through the integrated message box.

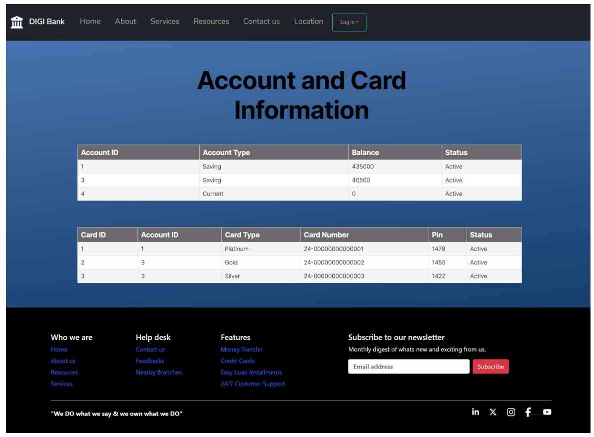




v. Personal Details: Check personal details.



vi. Account and Card Details: View account and card details.



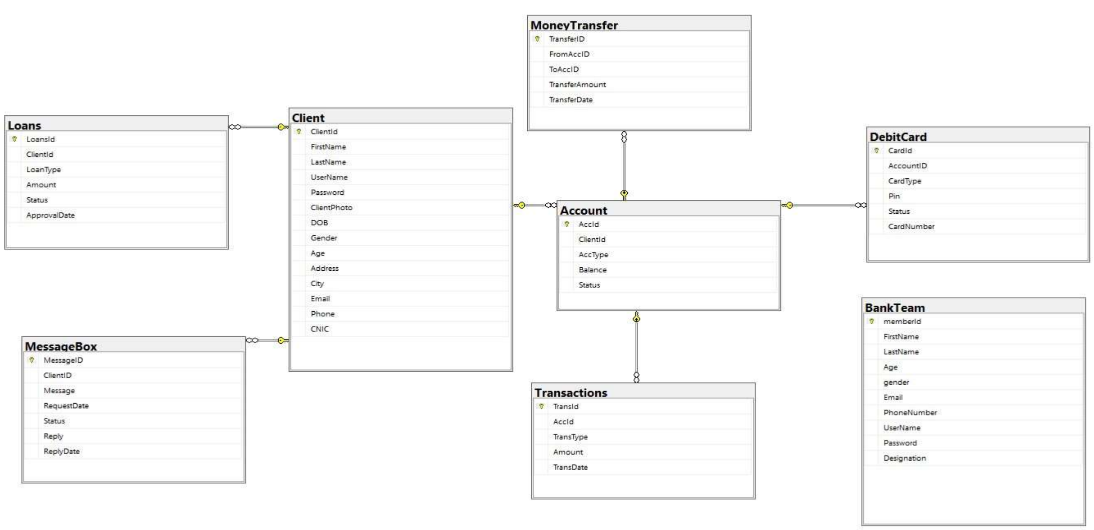
➢ Azure Database Integration:

i. Utilizes Azure database services for efficient management of data.
ii. Provides secure storage and reliable access to the database from anywhere via the internet.
iii. Enables scalability and high availability of data through Azure server infrastructure.

➢ Facial Recognition:

i. Implements advanced facial recognition technology using the Google Facenet framework to authenticate clients securely.
ii. Ensures the integrity of the application by verifying the identity of clients before granting them access to their dashboard.

3. Database Schema:



Here's a schema design for our database of Bank Management System (DigiBank):

Tables:

1. **Bank Team:**
    - MemberID (Primary Key, Identity(Auto Increment))
    - FirstName (Not Null)
    - LastName (Not Null)
    - Age (Check(Age >= 18), Not Null)
    - Gender (Check(Male or Female), Not Null)
    - Email (Unique)
    - Phone (Unique)
    - Username (Unique, Not Null)
    - Password (Not Null)
    - Designation (Not Null) - Admin, Account Manager, Transaction Manager, Financial Products Coordinator roles.
2. **Clients:**
    - ClientID (Primary Key, Identity(Auto Increment))
    - FirstName (Not Null)
    - LastName (Not Null)
    - Username (Unique, Not Null)
    - Password (Not Null)
    - Client_Photo (Not Null)
    - DateOfBirth (Not Null)
    - Gender (Check(Male or Female), Not Null)
    - Age (Check(Age >= 18), Not Null)
    - Address (Not Null)
    - City (Not Null)
    - Email (Unique)
    - Phone (Unique)
    - CNIC (Unique, Not Null)
3. **Accounts:**

- AccountID (Primary Key, Identity(Auto Increment))
- ClientID (Foreign Key referencing Clients table, Not Null)
- AccountType (Check(AccountType = Saving Account OR AccountType = Current Account), Not Null)
- Balance (Default 0, Not Null)
- Status (Default 'Active', Not Null)

4. **Debit Card:**
   - CardID (Primary Key, Identity(Auto Increment))
   - AccountID (Foreign Key referencing Accounts table, Not Null)
   - CardType (Check(CardType = Silver OR CardType = Gold OR CardType = Platinum), Not Null)
   - Card Number (16-digit number, Not Null, Unique, Default (CONCAT(RIGHT(YEAR(GETDATE()), 2), '-', LPAD(CAST(CardID AS varchar(14)), 14, '0'))))
   - Pin (4-digit password, Not Null)
   - Status (Default 'Active', Not Null)
5. **Transactions:**
   - TransactionID (Primary Key, Identity(Auto Increment))
   - AccountID (Foreign Key referencing Accounts table, Not Null)
   - TransactionType (Check (TransactionType = Deposit OR TransactionType = Withdraw), Not Null)
   - Amount (Not Null)
   - TransactionDate (Not Null, Default GetDate())
6. **Money Transfer:**
   - TransferID (Primary Key, Identity(Auto Increment))
   - FromAccountID (Foreign Key referencing Accounts table, Not Null)
   - ToAccountID (Foreign Key referencing Accounts table, Not Null)
   - TransferAmount (Not Null)
   - TransferDate (Not Null, Default GetDate())
7. **Loans:**
   - LoanID (Primary Key, Identity(Auto Increment))

- ClientID (Foreign Key referencing Clients table, Not Null)
- LoanType (Not Null)
- Amount (Not Null)
- Status () (Check (Status = Pending OR Status = Approved OR Status = Not Approved), Default 'Pending', Not Null)
- ApprovalDate (Default 'Null')

8. **MessageBox:**
    - MessageID (Primary Key, Identity(Auto Increment))
    - ClientID (Foreign Key referencing Clients table, Not Null)
    - Message (Not Null)
    - RequestDate (Not Null, Default GetDate())
    - Status (Check (Status = In Progress OR Status = Resolved), Not Null, Default ‘In Progress’)
    - Reply (Default ‘Waiting for Reply’)
    - ReplyDate (Not Null, Default GetDate())