DBMS PROJECT MID EVALUATION REPORT GROUP 48

Group Members:-

Rahul Kumar 2020110

Harshit Saini 2019243

Surya 2020138

Vishal 2020154

Problem Statement:

A small bank named “Gramin Bank” uses the traditional file-processing system where various records are stored in the files. They had been facing many problems like data redundancy, data inconsistency, unshareable data, etc.. They are planning to automate the whole process for which they need a perfectly well-designed database and portal. The requirements are given below:-

- The customer will have a unique id by which he can be uniquely identified, First name, last name, address DOB, and the credentials and multiple contact numbers.
- Customers can have a savings account and a current account. And can transfer money to the account having in our bank.
- Customers can request multiple services: checkbook request ,atm card request, recharge, passbook request etc.. and can complain about anything.
- A savings account may have an atm card for withdrawing cash or depositing cash through atm card. Any customer having account requests for cash withdrawal or deposit cash through atm. The database should validate ATM credentials and then the transaction can take place.
- Branch or through online loan can be issued to customer irrespective of whether he has an account in the bank or not. All customers are loyal to us. They will pay the EMI timely. So customers can pay the EMI by itself on the portal or by visiting the branch.
- EMI calculator will be also there. Before applying for the loan he can check the emi’s.
- There is a condition that customers will be issued a loan if they have successfully paid the previous loan.
- The branch will have a unique id, address, and other contact details.
- Employee will be a customer in our bank. So we can process the salary to the account. He/she can also be identified by

employee_ID. they will have designation,Date of Joining ,credentials etc… There will be a manager in a branch which will manage and supervise all the employees working in that branch.

- Provide the searching facilities based on various factors such as account,Customer, transaction, Cash withdrawal or deposit through atm.
- Employee will have the access to all the complaint's or request for service
- Manage the details of transactions.
- Editing and updating of records is available

Scope of Project:

- Customers can perform financial transactions like transfer funds online, pay bills, apply for loans and open a savings account among various other debit card transactions.
- Users cannot change the date of transactions.
- Basic functionalities of the system are:-
    - ❖ Create account
    - ❖ Login
    - ❖ Update profile
    - ❖ Password Recovery
    - ❖ View Profile
    - ❖ Complain
    - ❖ Request Service
    - ❖ ATM and Branch Finder
    - ❖ Statement
    - ❖ EMI and Loan Calculator
    - ❖ Apply for loan
    - ❖ ATM Cash Withdrawal/Deposit
    - ❖ Logout

Stakeholder Identified: Stakeholders in our Online Banking System are primiarily two kinds of people, the Customers and the Employees (including Managers)

Entities defined with Primary Key : Customer (Customer_ID), Account (Account_No), Branch (Branch_ID), Loan (Loan ID), Payment (Payment_ID, weak entity), Employee (Emp_id, has role manager), Transaction, Login (ID), Complaint (Complaint_ID), Employee Login (ID), ATM Card (ATM No), ATM (ATM ID), Service (ID).

Relationships: Represented in the ER Diagram with proper representations of participation types (double line for total participation, and single line for partial participation) and Cardinalities using notations along the relationship connector (1:N, N:1, M:N, 1:1) and also defined roles where applicable. Same for all the Attributes of Entities and the relationships.

Identification of Weak Entity: Payment is made to be a weak entity as it is the payment for a Loan, which is it's strong entity as payment cannot individually exist without Loan. So it is depended on the entity loan. Payment has payment no as discriminator/partial key, to correctly idenityfy it we would need primary key of Loan which is Load ID as well.

Ternary Relationship: No ternary relationship exists in the model of our ER Diagram. All the relationships were able to be broken into binary relationships.

Relational Schema:

Customer (CustomerID , First_Name, Last_Name, Street, City, State, Pincode, DOB, User_ID, Password)

Customer_Contact (CustomerID, PhoneNo)

Service (ID, CustomerID, Service_Type, DORQ, DORS, Service_Status, Service_Description)

Complain (Complaint_ID , CustomerID, Complaint_ID, Complain_Status, Complain_Description)

Branch(Branch_ID , City, State, Pincode)

Branch_Contact (Branch_ID , Telephone)

Branch_Email (Branch_ID , Email)

Employee (Emp_ID , User_ID, Password, Emp_Name, Branch, DOJ, Designation)

mgr_emp(Emp_ID int, Mgr_ID) [Candidate Key: Emp_ID, Mgr_ID]

Cust_Emp(Emp_ID , CustomerID)
Loan (CustomerID, Branch_ID, Loan_ID , Loan_Type, Duration, Amount, Rate, Loan_status, Amount_Paid)

Payment (Loan_ID, Payment_Id, Date, Amount int) [Weak Entity so need primary key of Related Strong Entity]

Account (CustomerID, Account_No , Balance, Date_of_opening]

Saving_account (Account_No, Interest_Rate)

Current_account (Account_No, Overdraft)

Transaction (ID , Account_No , Date_of_Transaction, Type_of_Transaction)

ATM_CARD(Account_No, Card_No , PIN, CVV)

ATM (ATM_ID , City, State, Pincode, Cash)

Withdraw_cash (ATM_ID , Card_No , Amount) [Candidate Key: ATM_ID, Card_No]

Population of Tables:

All tables populated with moderate amount of data, done by importing .csv files with rows and columns filled with random data created using scripts and Online Data Generator. Unique attributes like Customer ID, account numbers have been kept the same across all tables. Tables are filled with test data keeping in mind the constraints like participation constraints and cardinialities in relationships in such a way that it is able to test most of possible functionaltiies that are implemented or will be implemented in future. Edge cases also considered and filling the tables with data.

Sample query for insertion of data in customer table:

```
INSERT INTO `customer` VALUES
(43008270,'Nicola','Lowten','Dayton','Sanok','Puducherry',106928,'2
006-10-02','nlowten15','raVj4RRerA6'),
(50394832,'Martainn','Zolini','Jenifer','Napanee
Downtown','Arunachal
Pradesh',302298,'1951-09-05','mzolinis','RRvVpHcP7bIz');
```

SQL QUERIES-

1. select branch.Branch_ID, email,city,state,pincode from branch inner join branch_email where branch.Branch_ID=branch_email.Branch_ID;

2. create view complain_list_notresolved as select complain.Complaint_ID,Customer.customerid,customer.first_name,customer.last_name,customer.state,customer_contact.PhoneNo ,complain.Complain_Status from ((customer inner join complain on customer.CustomerID=complain.CustomerID and complain.Complain_Status="not resolved") inner join customer_contact on customer_contact.CustomerID=customer.CustomerID) order by complain.Complaint_ID asc;

3. SELECT First_name,last_name,Street,City,State,Pincode,DOB FROM Customer WHERE City LIKE 'B%' or City LIKE 'S%';

4. CREATE USER 'employee'@'localhost' IDENTIFIED BY 'password';
   GRANT select,delete ON bank.customer TO 'employee'@'localhost';
   FLUSH PRIVILEGES;

5. SELECT CustomerID FROM account_ WHERE EXISTS (Select Account_No from transaction where account_.Account_No=transaction.Account_No and Type_of_Transaction="credit");

6. select customerid,first_name,last_name from customer where customerID in (SELECT CustomerID FROM account_ WHERE EXISTS (Select Account_No from transaction where

account_.Account_No=transaction.Account_No and Type_of_Transaction="debit"));

7. SELECT loan.branch_id,count(loan.Loan_ID) as "No. of loans issued" FROM loan GROUP BY Branch_ID order by count(loan.Loan_ID) ;

8. SELECT (state), COUNT(state) as "No. of customers" FROM customer GROUP BY state having count(state)>=2

9. UPDATE customer SET First_name = "Nichloas" WHERE First_Name = 'Nicola';

10. ALTER TABLE branch ADD IFSC varchar(20);

ER DIAGRAM:-

Click here