Registration(Name, DOB, emailID, Phone no., username, password, confirm-password, address, cardtype(gold, platinum), bank name, Acc no, IFSC code) -> (Register, Reset, Login)

Login(username, password)

Dashboard(

1. user card details(card type, card no, validity), credit(total, used, remaining,
2. Products purchased(product name, total, paid, balance)
3. Recent(product name, date, amount paid)

Product List(Array of Products)

Product(product name, product details, cost, EMI period(3, 6, 9, 12 months), pay now)

Admin( activate/deactivate users, show all users )

Change password(username, old password, new password)

Forgot password(phone no, get otp, enter otp, verify otp) // JFrame

1. Product List <Product: Read All Products>
   1. Product 1
      * 1. product name
        2. product details
        3. cost
        4. Buy now
           1. EMI period(3, 6, 9, 12 months)
   2. …
   3. …
   4. …
2. Register <User: Create new user>
   1. Fill the registration form:
      1. Name
      2. DOB
      3. Email ID
      4. Phone no.
      5. Username
      6. Password
      7. confirm-password
      8. address
      9. card type(gold, platinum)
      10. bank name
      11. Account no
      12. IFSC code
      13. Reset (fill again)
      14. Register
   2. Already a User ? Login
3. Login
   1. Login <User: Verify username, password>
      1. username
      2. password
   2. Forgot password <User: Verify username, phone no.>
      1. Phone no
      2. Enter OTP <Jframe>
   3. Change Password <User: Update password>
      1. Username
      2. Old password
      3. New password
      4. Confirm password
   4. New user ? Register
4. AdminLogin <Admin: Verify username, password>
   1. Username
   2. Password
5. Dashboard (if user is logged in)
   1. User card details <User: Read Card Details>
      1. Card no.
      2. Cardtype
      3. Validity --- ??????????? ---
      4. Credit (total, used , remaining)
   2. Products purchased <Purchase: Read all products with given user id>

<User: total, used credit>

* + 1. Product List ---- Installments ?? ---
    2. Total
    3. Paid
    4. Balance
  1. Recent
     1. Product name
     2. date
     3. amount paid

1. Admin page(if admin is logged in)
   1. Manage activation requests
      1. Name
      2. phone number
      3. card type
      4. activate
   2. Show all users
      1. Name
      2. phone number
      3. card type
   3. deactivate account using username
      1. uid ??

CREATE DATABASE finance\_app;

USE finance\_app;

CREATE TABLE users (

user\_id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(100) NOT NULL,

dob DATE NOT NULL,

email VARCHAR(100) NOT NULL UNIQUE,

phone\_no VARCHAR(15) NOT NULL UNIQUE,

username VARCHAR(50) NOT NULL UNIQUE,

password VARCHAR(255) NOT NULL,

address TEXT,

card\_type VARCHAR(10) NOT NULL CHECK (card\_type IN ('gold', 'platinum')),

bank\_name VARCHAR(100) NOT NULL,

account\_no VARCHAR(30) NOT NULL UNIQUE,

ifsc\_code VARCHAR(15) NOT NULL,

total\_credit DECIMAL(15, 2) DEFAULT 0.00,

used\_credit DECIMAL(15, 2) DEFAULT 0.00,

is\_active BOOLEAN DEFAULT FALSE,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

card\_validity TIMESTAMP DEFAULT (DATE\_ADD(CURRENT\_TIMESTAMP, INTERVAL 3 YEAR))

);

DELIMITER $$

CREATE TRIGGER set\_credit\_before\_insert

BEFORE INSERT ON users

FOR EACH ROW

BEGIN

IF NEW.card\_type = 'gold' THEN

SET NEW.total\_credit = 100000.00;

SET NEW.used\_credit = 0.00;

ELSEIF NEW.card\_type = 'platinum' THEN

SET NEW.total\_credit = 300000.00;

SET NEW.used\_credit = 0.00;

END IF;

END$$

DELIMITER ;

DELIMITER $$

CREATE TRIGGER update\_total\_credit\_before\_update

BEFORE UPDATE ON users

FOR EACH ROW

BEGIN

IF OLD.card\_type <> NEW.card\_type THEN

IF NEW.card\_type = 'gold' THEN

SET NEW.total\_credit = 100000.00; -- You can also use := for assignment

ELSEIF NEW.card\_type = 'platinum' THEN

SET NEW.total\_credit = 300000.00; -- You can also use := for assignment

ELSE

SET NEW.total\_credit = 0.00; -- You can also use := for assignment

END IF;

END IF;

END$$

DELIMITER ;

CREATE TABLE products (

product\_id INT AUTO\_INCREMENT PRIMARY KEY,

product\_name VARCHAR(100) NOT NULL,

product\_details TEXT NOT NULL,

cost DECIMAL(10, 2) NOT NULL

);

CREATE TABLE purchases (

purchase\_id INT AUTO\_INCREMENT PRIMARY KEY,

user\_id INT,

product\_id INT,

emi\_period VARCHAR(10) NOT NULL CHECK (emi\_period IN ('3 months', '6 months', '9 months', '12 months')),

purchase\_date TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

total\_amount DECIMAL(10, 2) NOT NULL,

amount\_paid DECIMAL(10, 2) DEFAULT 0.00,

installment\_count INT DEFAULT 0,

installment\_amount DECIMAL(10, 2) DEFAULT 3,

payment\_status ENUM('completed', 'pending', 'failed') DEFAULT 'pending',

FOREIGN KEY (user\_id) REFERENCES users(user\_id),

FOREIGN KEY (product\_id) REFERENCES products(product\_id)

);

CREATE TABLE installments (

installment\_id INT AUTO\_INCREMENT PRIMARY KEY,

purchase\_id INT,

installment\_due\_date TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

amount DECIMAL(10, 2) NOT NULL,

payment\_status ENUM('paid', 'pending') DEFAULT 'pending',

FOREIGN KEY (purchase\_id) REFERENCES purchases(purchase\_id)

);

DELIMITER $$

CREATE TRIGGER insert\_installments\_after\_purchase

AFTER INSERT ON purchases

FOR EACH ROW

BEGIN

DECLARE i INT DEFAULT 0;

DECLARE installment\_amount DECIMAL(10, 2);

DECLARE due\_date TIMESTAMP;

SET installment\_amount = NEW.total\_amount / NEW.installment\_count;

SET due\_date = CURRENT\_TIMESTAMP;

WHILE i < NEW.installment\_count DO

INSERT INTO installments (purchase\_id, installment\_due\_date, amount, payment\_status)

VALUES (NEW.purchase\_id, due\_date, installment\_amount, 'pending');

SET i = i + 1;

SET due\_date = DATE\_ADD(due\_date, INTERVAL 1 MONTH);

END WHILE;

END$$

DELIMITER ;

CREATE TABLE admin (

admin\_id INT AUTO\_INCREMENT PRIMARY KEY,

username VARCHAR(50) NOT NULL UNIQUE,

password VARCHAR(255) NOT NULL,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

INSERT INTO users (name, dob, email, phone\_no, username, password, address, card\_type, bank\_name, account\_no, ifsc\_code, is\_active)

VALUES

('Amit Sharma', '1990-05-14', 'amit.sharma@example.com', '9876543210', 'amit90', 'password123', '123 MG Road, Mumbai, Maharashtra', 'gold', 'HDFC Bank', '123456789012', 'HDFC0001234', TRUE),

('Priya Gupta', '1995-09-23', 'priya.gupta@example.com', '8765432109', 'priya95', 'password456', '456 MG Road, Bangalore, Karnataka', 'platinum', 'SBI', '234567890123', 'SBIN0001234', TRUE),

('Rahul Verma', '1988-12-11', 'rahul.verma@example.com', '7654321098', 'rahul88', 'password789', '789 MG Road, Delhi', 'gold', 'ICICI Bank', '345678901234', 'ICIC0001234', FALSE),

('Sita Rani', '1992-03-04', 'sita.rani@example.com', '6543210987', 'sita92', 'password101', '321 MG Road, Kolkata, West Bengal', 'platinum', 'Axis Bank', '456789012345', 'UTIB0001234', TRUE),

('Vikram Singh', '1985-07-30', 'vikram.singh@example.com', '5432109876', 'vikram85', 'password202', '987 MG Road, Chennai, Tamil Nadu', 'gold', 'Punjab National Bank', '567890123456', 'PUNB0001234', TRUE);

INSERT INTO products (product\_name, product\_details, cost)

VALUES

('Smartphone', 'Latest Android smartphone with 128GB storage.', 30000.00),

('Laptop', '15-inch laptop with Intel i5 processor and 8GB RAM.', 60000.00),

('Smartwatch', 'Water-resistant smartwatch with fitness tracking.', 10000.00),

('Wireless Headphones', 'Noise-cancelling wireless headphones.', 5000.00),

('Tablet', '10-inch tablet with WiFi connectivity and 64GB storage.', 20000.00);

INSERT INTO purchases (user\_id, product\_id, emi\_period, total\_amount, installment\_count)

VALUES

(1, 1, '12 months', 30000.00, 12),

(2, 2, '6 months', 60000.00, 6),

(3, 3, '3 months', 10000.00, 3),

(4, 4, '9 months', 5000.00, 9),

(5, 5, '12 months', 20000.00, 12);

INSERT INTO admin (username, password) VALUES ('admin', 'adminpass123');