

ONLINE BANKING SYSTEM

A Project Report

Submitted by

AMAN KUMAR THAKUR – AP23110011682

BIBEK KUMAR SAH – AP23110011548

SAMEER BACHHAR – AP23110011496

BIKASH KUMAR MAHATO – AP23110011681

Under the Supervision of

Dr. Rajesh Yelchuri

Assistant Professor

Department of Computer Science and Engineering

SRM University-AP

In partial fulfillment for the requirements of the Web Technology (CSE210) Project

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SRM UNIVERSITY-AP

NEERUKONDA

MANAGALAGIRI - 522503

ANDHRA PRADESH, INDIA

APRIL-2025

CERTIFICATE

This is to certify that the project work entitled “**ONLINE BANKING SYSTEM**” is a Bonafide record of project work carried out by the following students:

- **Mr. Aman Kumar Thakur** (Roll No.: AP23110011682)
- **Mr. Bibek Kumar Sah** (Roll No.: AP23110011548)
- **Mr. Sameer Bachhar** (Roll No.: AP23110011496)
- **Mr. Bikash Kumar Mahato** (Roll No.: AP23110011681)

from the **Department of Computer Science and Engineering, SRM University-AP.**

The students conducted this project work under my supervision during the period **January 2025 to April 2024**. It is further certified that, to the best of my knowledge, this project has not previously formed the basis for the award of any degree or any similar title to this or any other candidate.

This is also to certify that the project work represents the **teamwork** of the candidates.

Station: Mangalagiri
Date: 22-04-2025

Dr. Rajesh Yelchuri
Assistant Professor
Department of Computer Science & Engineering
SRM University-AP
Andhra Pradesh.

TABLE OF CONTENTS

S.NO.	TITLE	PAGENO.
1	Introduction	1
2	Problem Definition	2
3	Problem Statement	3
4	Objectives	4-5
5	Software Requirements Specifications	6-7
6	Design(DFD Diagram)	8-9
7	Implementation	10
8	Database Design	11-18
9	Output Screenshots	19-22
10.	Advantages & Limitations	23-25
11.	Future Scope	26-27
12.	Conclusion	28

1: INTRODUCTION

Online banking is a modern financial solution that allows customers to conduct banking transactions through a web-based platform. It eliminates the need for physical visits to the bank and offers 24/7 access to essential services such as balance inquiry, fund transfer, and transaction monitoring. In this project, titled “Online Banking System”, we aim to replicate the key functionalities of a banking environment using web technologies like HTML, CSS, JavaScript, PHP, and MySQL.

This system provides separate interfaces and functionalities based on the user's role Admin, Staff, or Customer. Customers can access their personal account dashboards, while staff members can approve registrations and credit user accounts. The admin has full control over all records and operations. By developing this project, we explored secure session handling, database integration, and user-friendly interface design, contributing to our skills in full-stack web development.

Bullet Points:

- Provides banking services digitally using web technologies.
- Simulates real-world banking functions such as login, account balance, and fund transfers.
- Supports three roles: Admin, Staff, and Customer.
- Uses PHP for backend logic and MySQL for data storage.
- Encourages security through authentication and session control.

2: PROBLEM DEFINITION

Traditional banking methods involve customers physically visiting bank branches for routine tasks such as checking account balances, transferring funds, updating personal details, and managing beneficiaries. This process is often time-consuming, inconvenient, and inefficient, especially for working individuals or those living in remote areas. Customers frequently experience long queues, extensive paperwork, and delays in service, leading to frustration and dissatisfaction.

Additionally, manual record-keeping and operational processes are susceptible to human errors, mismanagement, and even fraud, posing serious risks to data accuracy and financial security. Limited banking hours further restrict customer access, making it difficult for people to conduct transactions at their convenience.

In today's fast-paced digital environment, customers expect instant, secure, and user-friendly online services. The demand for an efficient, error-free, and accessible online banking solution has significantly increased, necessitating a modern system that ensures better service delivery, reduces dependency on physical branches, and enhances security and convenience.

Bullet Points:

- 1. Traditional banks require physical presence for routine transactions.**
 - Customers must visit branches for simple services like balance checks, fund transfers, and beneficiary updates.
- 2. Long queues and manual forms create user dissatisfaction.**
 - Standing in lines and filling out paper forms wastes time and leads to a poor customer experience.
- 3. Operations are prone to human error and fraud.**
 - Manual data handling increases the risk of mistakes, misplacement of records, and fraudulent activity.
- 4. Limited banking hours restrict accessibility.**
 - Services are only available during fixed hours, making it inconvenient for customers with busy schedules.
- 5. Increasing demand for secure and fast online services.**
 - There's a growing expectation for 24/7 banking solutions that are safe, reliable, and easy to use from anywhere.

3: PROBLEM STATEMENT

The purpose of this project is to develop a secure, web-based Online Banking System that enables customers to conveniently access and manage their bank accounts through an internet-based platform. The system will offer essential functionalities such as account login, beneficiary management, fund transfers, and transaction history access for customers. In addition, it must include staff and administrative functionalities, allowing authorized personnel to manage customer data, credit funds to accounts, approve new registrations, and monitor financial transactions. One of the key challenges in developing this system is ensuring the confidentiality, integrity, and availability of sensitive user data. The platform must implement secure session handling, OTP-based verification, and encrypted data transmission to protect against unauthorized access and data breaches. The system should simulate core banking processes in a user-friendly, responsive, and reliable web application environment.

Bullet Points:

- 1. Simulate key features of online banking via web application.**
 - Replicate common banking services like viewing account details, transferring funds, and managing beneficiaries through a browser-accessible interface.
- 2. Provide login and registration functionality.**
 - Implement a secure registration process for new customers and staff, along with a login system featuring role-based access control.
- 3. Implement beneficiary management and fund transfer modules.**
 - Allow customers to add, view, and delete beneficiaries as well as perform secure fund transfers to registered beneficiaries.
- 4. Allow admin/staff to manage and credit accounts.**
 - Provide authorized staff with functionalities to credit funds to customer accounts, update account details, and approve or deactivate accounts when necessary.

4: OBJECTIVES

The primary objective of this project is to design and develop a responsive, secure, and efficient online banking system. This platform is intended to serve different categories of users namely customers, bank staff, and administrators by providing them with role-specific functionalities through a clean and organized user interface. The system focuses on offering smooth interaction with the database while enforcing strict session management and ensuring data security.

A major goal is to implement role-based access control, which ensures that users can only access functionalities appropriate to their role in the system. Customers should be able to securely log in, check their balances, transfer money, and manage their beneficiaries, while staff members have privileges to manage customer accounts, credit balances, and oversee transactions. Additionally, password management is a crucial objective, allowing users to reset their credentials securely through an OTP-based verification process.

Moreover, this project emphasizes data integrity and secure communication with the server. Every transaction and data exchange should be reliably logged and protected from unauthorized access. The modular design structure of the system is intended to simplify maintenance, enabling developers to easily update individual features without affecting the overall system performance or stability.

Detailed Bullet Points:

1. Build secure login and role-based access control.

- The system will implement a login module using PHP and session management techniques.
- Different user roles such as Customer, Staff, and Admin will have restricted access based on predefined permissions.
- Unauthorized users will be prevented from accessing restricted modules or data.

2. Enable customers to view balances and transfer money.

- Customers will be able to securely log into their personal accounts and check their current account balance in real-time.
- A dedicated fund transfer module will allow customers to safely transfer money to registered beneficiaries.

3. Let staff credit user accounts and manage customer data.

- Bank staff will have access to a staff dashboard with the ability to credit funds to customer accounts.
- Staff members can manage customer records, update information, and oversee financial transactions.

4. Allow users to reset passwords using OTP.

- A password recovery module will be provided, allowing users to request a One-Time Password (OTP) via email.
- Upon receiving the OTP, users can verify their identity and reset their password securely.

5. Ensure data integrity and secure communication with the server.

- All transactions, transfers, and data modifications will be reliably saved in the MySQL database, maintaining data consistency and integrity.
- Secure protocols (such as HTTPS) and session encryption techniques will be used to protect data during transmission.

5: SOFTWARE REQUIREMENTS SPECIFICATIONS

A. Functional Requirements:

These requirements specify the core functionalities that the system must provide to its users, based on their respective roles.

1. User Registration and Login.

- The system will allow customers and staff to securely register and log in to the platform using unique credentials.

2. Fund Transfers with Beneficiary Management.

- Customers can manage a list of trusted beneficiaries and perform fund transfers only to those registered accounts.

3. View and Download Transaction History.

- Customers can view a detailed list of their past transactions and optionally download transaction reports for personal records.

4. Staff Approval for Customer Accounts.

- New customer registrations or account updates require approval from bank staff or administrators to maintain data integrity and compliance.

5. OTP-Based Password Reset.

- In case of forgotten passwords, users can request a one-time password (OTP) sent via email, which allows them to securely reset their credentials.

B. Non-Functional Requirements:

These requirements describe the system's performance, security, usability, and compatibility characteristics.

1. Cross-Browser Compatibility.

- The application will function properly on all major web browsers such as Google Chrome, Mozilla Firefox, Microsoft Edge, and Safari.

2. Responsive Design Using Bootstrap.

- The front-end interface will be responsive and mobile-friendly, ensuring usability on desktops, tablets, and smartphones.

3. Secure Communication and Session Handling.

- User sessions will be securely managed using PHP sessions and secure data exchange protocols, minimizing the risk of unauthorized access.

4. Scalable Backend Using MySQL.

- The system will use a MySQL database, designed with normalization and relationships to efficiently handle data growth and maintain performance.

C. Hardware & Software Used:

Bullet Points:

1. Frontend:

- HTML, CSS, JavaScript, Bootstrap — for designing the web interface, navigation, and responsiveness.

2. Backend:

- PHP :- for implementing server-side functionality, session management, form processing, and database connectivity.

3. Database:

- MySQL:- for storing user information, transactions, account details, and other banking records.

4. Server:

- Apache (XAMPP):- a local server environment to run and test PHP applications with integrated MySQL.

5. Editor:

- Visual Studio Code (VS Code) or Sublime Text:- code editors used for developing, editing, and debugging the project's codebase.

6: DESIGN (DFD DIAGRAM)

The system is divided into various processes based on user roles. A Data Flow Diagram (DFD) helps understand the flow of information and system operations. The Level 0 DFD shows the overall system interaction, while Level 1 goes deeper into individual processes like login, transaction, and account management.

Bullet Points:

Level 0 DFD:

- External users: Customer, Staff, Admin.
- Processes: Authentication, Account Operations, Transactions.
- Data Stores: User database, Transaction records.

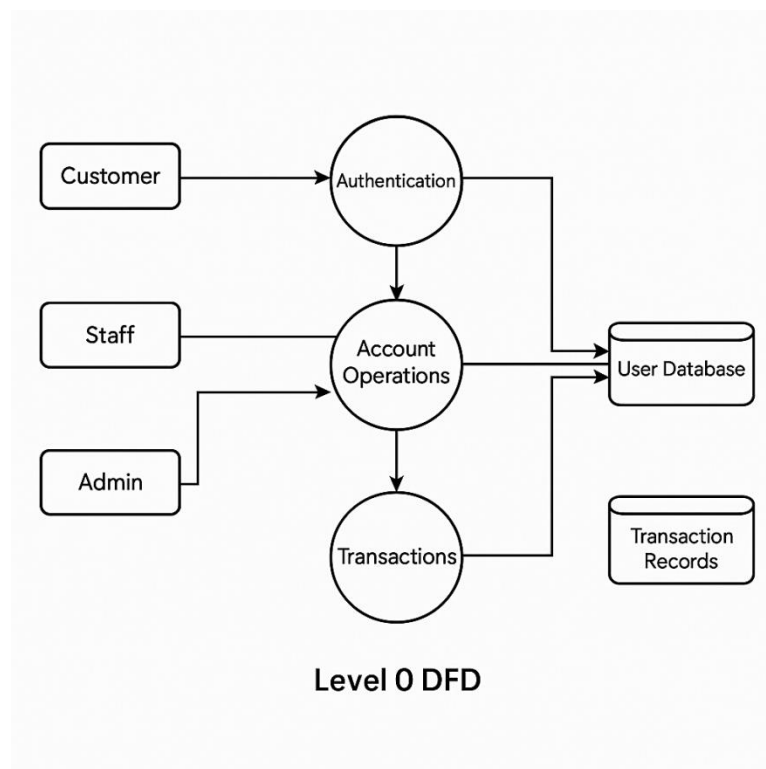


Figure 1 – Level 0 DFD: Overall Banking System.

(This diagram shows the high-level interaction between external users (Customer, Staff, Admin) and the core processes (Authentication, Account Operations, Transactions), along with the main data stores.)

Level 1 DFD:

- Login module: Validates credentials.
- Customer dashboard: Displays balance and options.
- Fund transfer: Executes and logs transactions.
- Beneficiary management: Adds/deletes beneficiaries.
- Staff dashboard: Credits account and manages records.

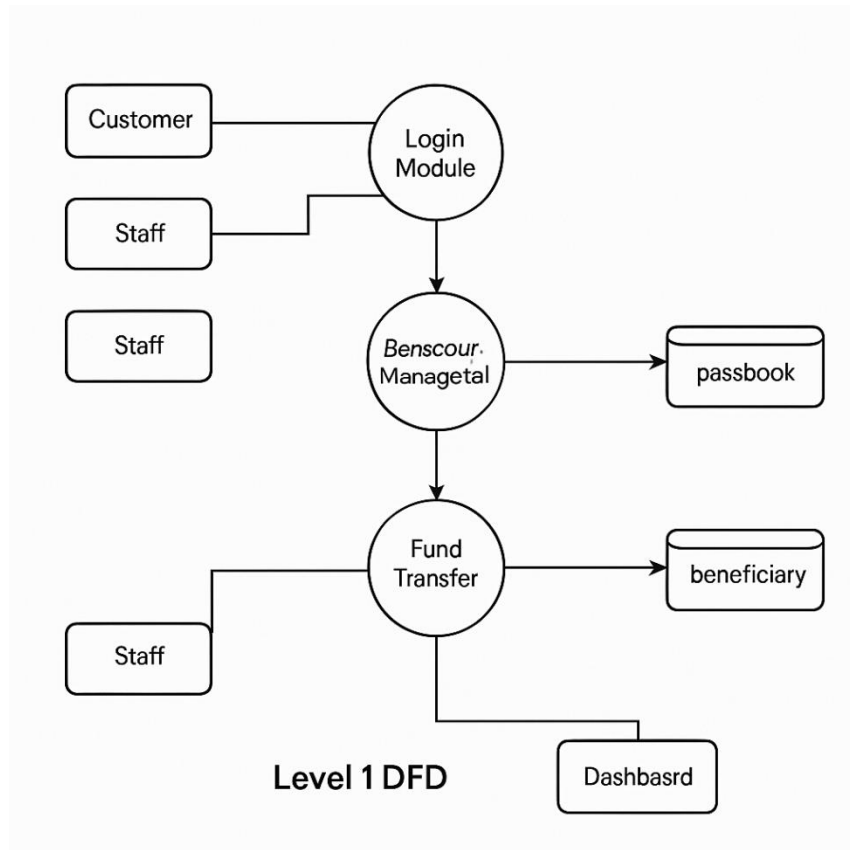


Figure 2 – Level 1 DFD: Functional Modules of the Banking System.

(This diagram breaks down the internal working of the system into detailed processes such as login, customer dashboard, fund transfer, beneficiary management, and staff dashboard with corresponding data flow and stores.)

7: IMPLEMENTATION

The system was developed using PHP and MySQL, utilizing session-based authentication for secure user access. Modular PHP files were created to handle core functionalities such as login, money transfers, beneficiary management, and staff account operations. The system interacts in real-time with the database, ensuring immediate updates and providing confirmation messages or alerts based on user actions. The codebase follows a clear separation of concerns, enhancing maintainability and scalability.

Bullet Points:

- *customer_login.php*: Validates user credentials and initiates a session upon successful login.
- *add_beneficiary.php*: Provides a form interface for customers to add or remove trusted beneficiary accounts.
- *customer_transfer_money.php*: Processes fund transfers between customer accounts, ensuring validity checks and transaction logging.
- *staff_credit_customer.php*: Allows staff members to credit customer accounts and manage transaction records securely.

Technologies Used:

- Frontend: HTML, CSS, JavaScript, Bootstrap
- Backend: PHP (Running on XAMPP Server)
- DatabaseManagementSystem: MySQL

8: DATABASE DESIGN

The database for this Online Banking System is designed using MySQL, following a normalized structure to reduce data redundancy and ensure efficient storage. The design includes several key entities such as users, accounts, beneficiaries, transactions, and staff. To maintain data integrity, relationships between these tables are carefully defined using primary keys and foreign keys. This allows for consistent and reliable connections between customers, their accounts, transaction records, and beneficiary details. By structuring the database in this way, the system ensures accurate data handling, easy retrieval, and secure storage of sensitive financial information. Bullet Points:


Main Tables:

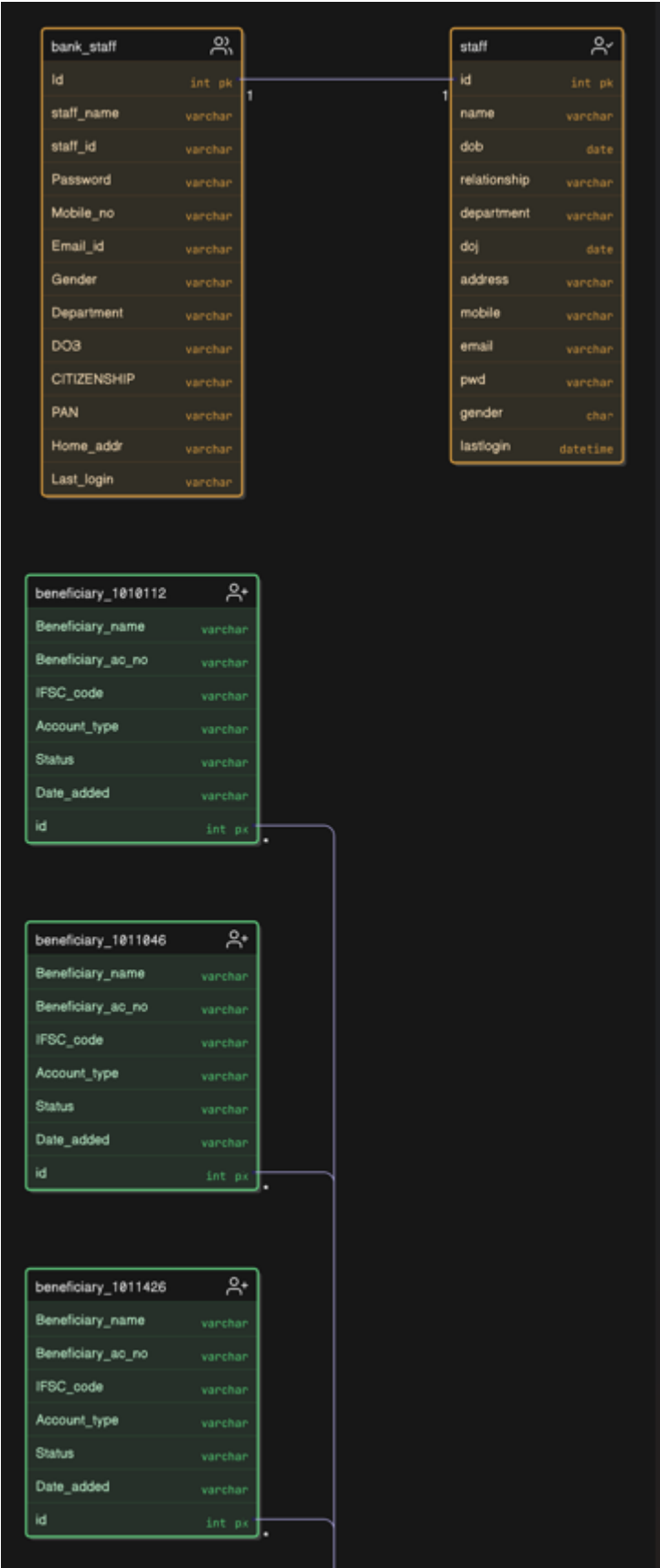
- users: Stores user data (id, name, email, role)
- accounts: Stores account number, type, and balance
- transactions: Records each debit/credit
- beneficiaries: Stores trusted external accounts
- staff: Stores staff login details


Relationships:


- Each user has multiple transactions.
- Beneficiaries are linked to customer accounts.
- Staff can access and manage customer records.


Banking System Data Model

pending_accounts 	
Application_no	varchar pk
Name	varchar
Gender	varchar
Mobile_no	varchar
Email_id	varchar
Landline_no	varchar
DOB	varchar
PAN	varchar
CITIZENSHIP	varchar
Home_Addr	varchar
Office_Addr	varchar
Country	varchar
State	varchar
City	varchar
Pin	varchar
Area_Loc	varchar
Nominee_name	varchar
Nominee_ac_no	varchar
Account_type	varchar
Application_Date	varchar

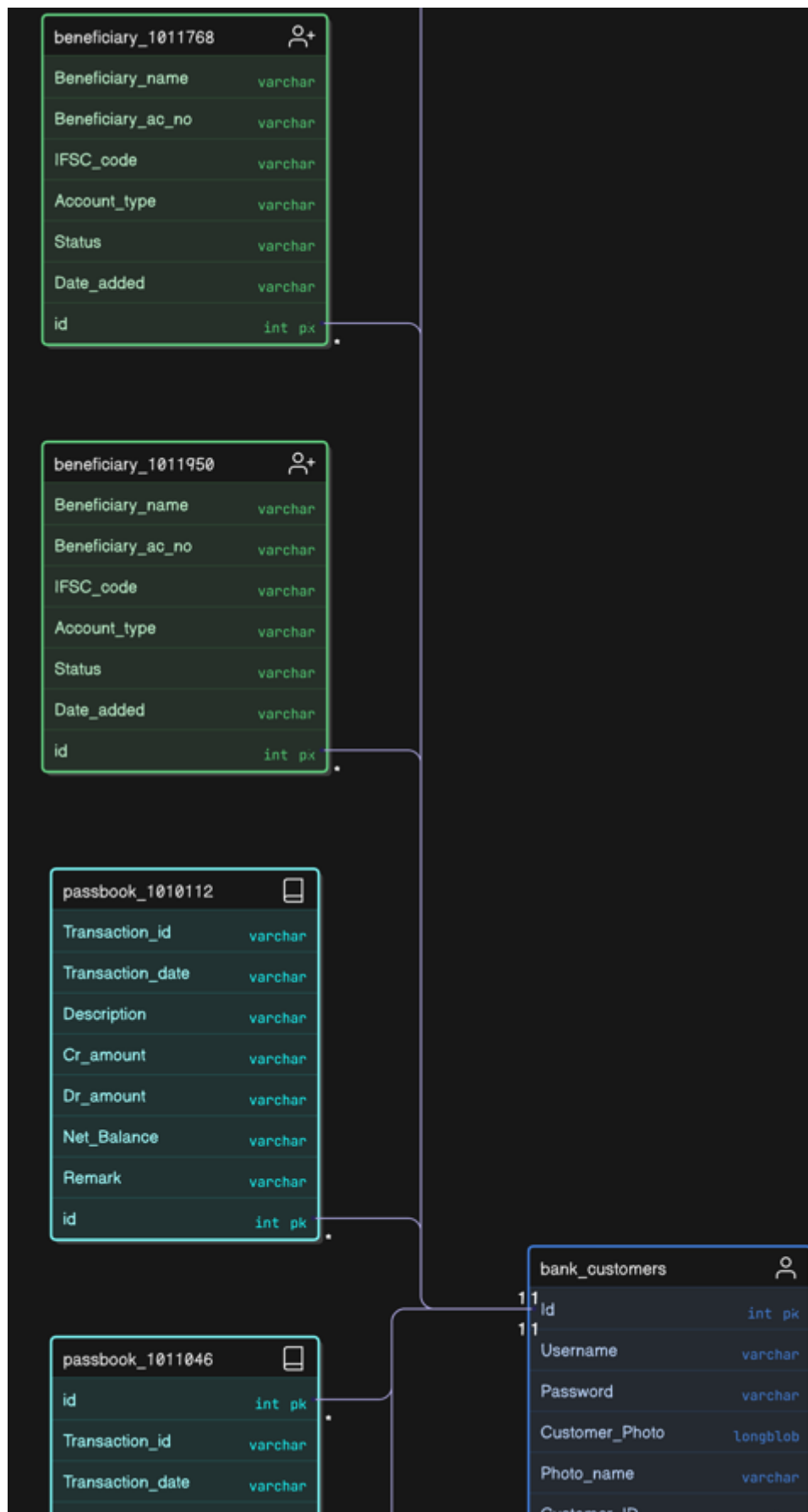


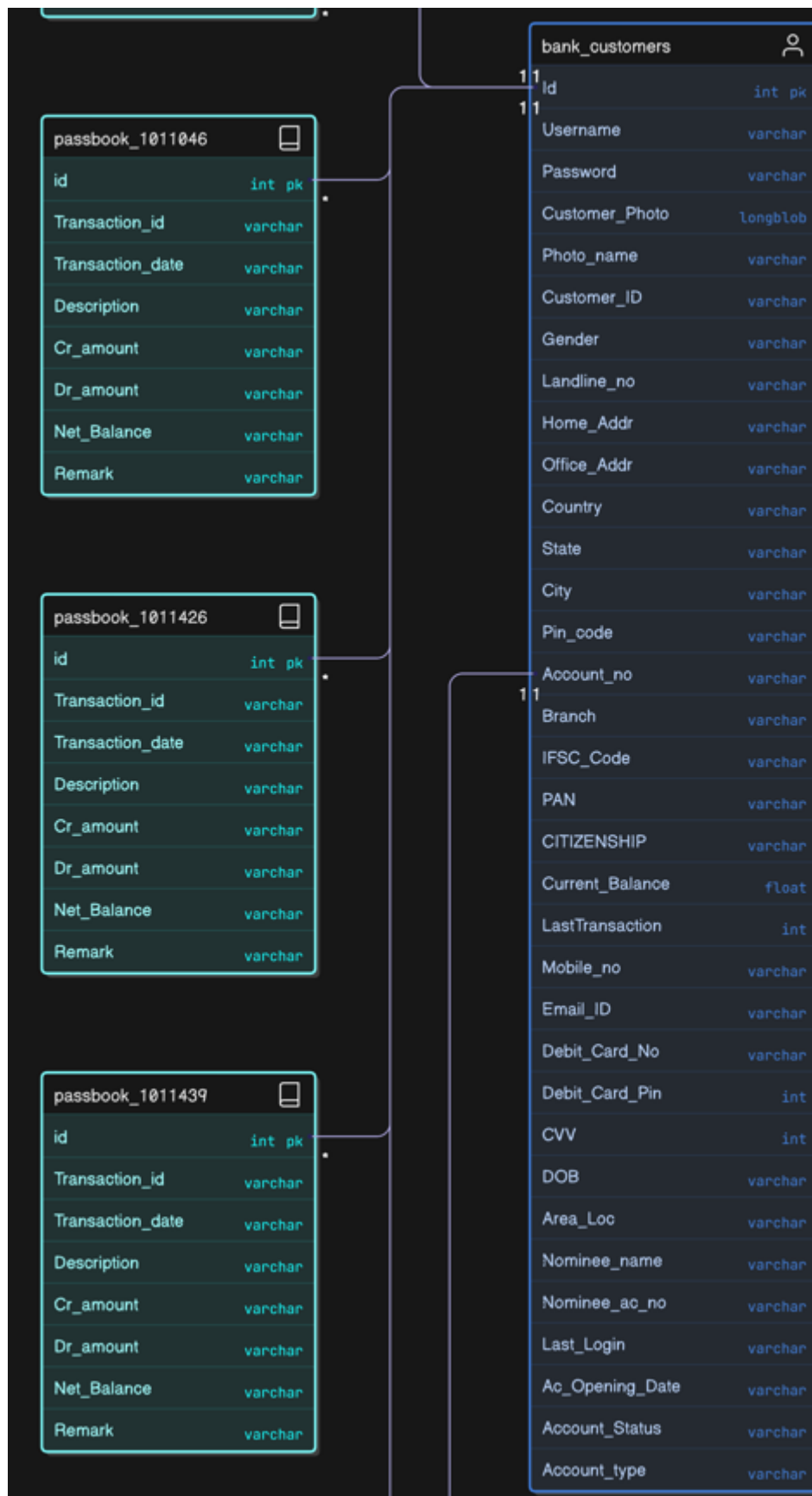
beneficiary_1011439 	
Beneficiary_name	varchar
Beneficiary_ac_no	varchar
IFSC_code	varchar
Account_type	varchar
Status	varchar
Date_added	varchar
id	int pk

beneficiary_1011722 	
Beneficiary_name	varchar
Beneficiary_ac_no	varchar
IFSC_code	varchar
Account_type	varchar
Status	varchar
Date_added	varchar
id	int pk

beneficiary_1011742 	
Beneficiary_name	varchar
Beneficiary_ac_no	varchar
IFSC_code	varchar
Account_type	varchar
Status	varchar
Date_added	varchar
id	int pk

beneficiary_1011768 	
Beneficiary_name	varchar
Beneficiary_ac_no	varchar
IFSC_code	varchar
Account_type	varchar





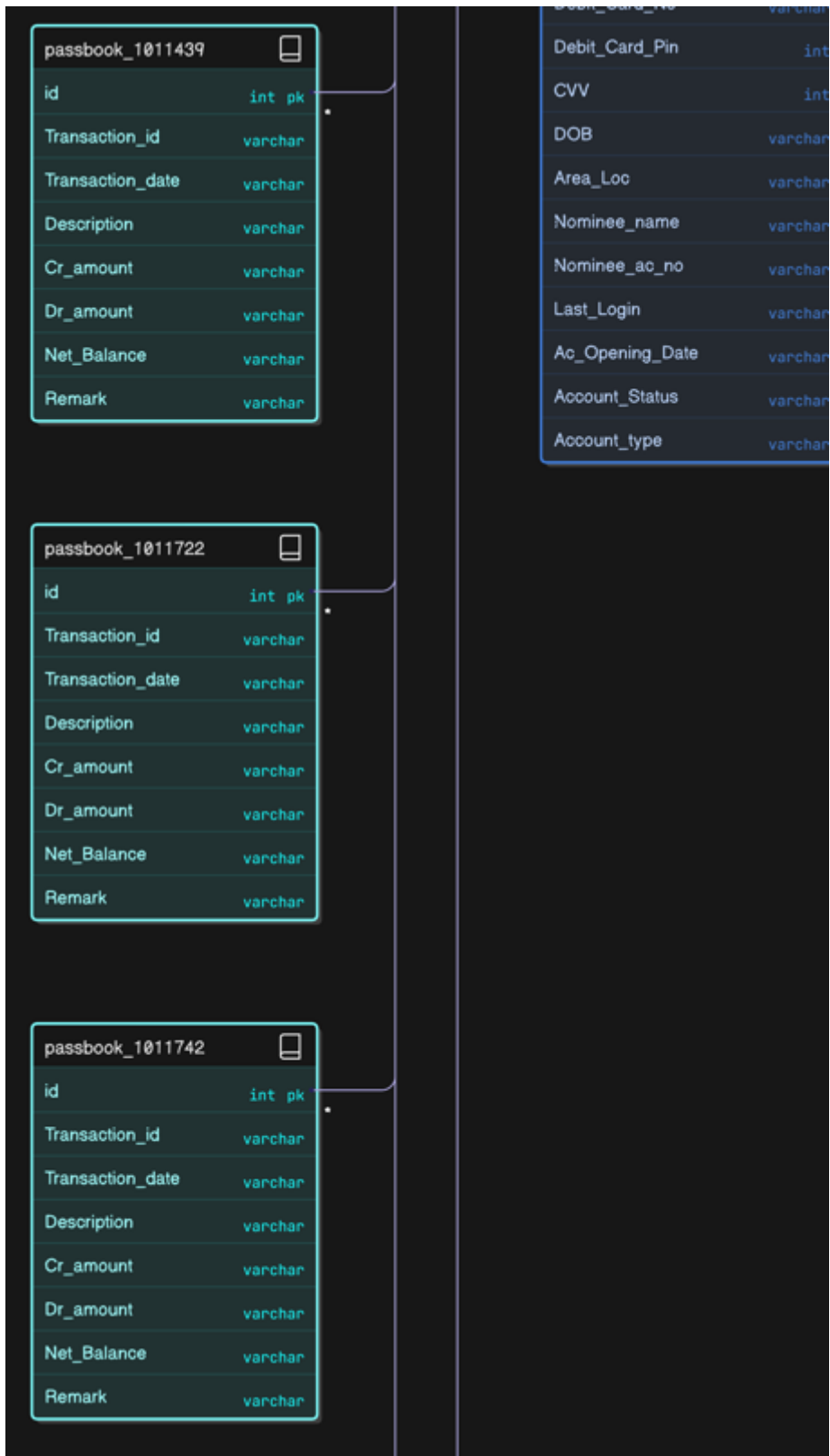
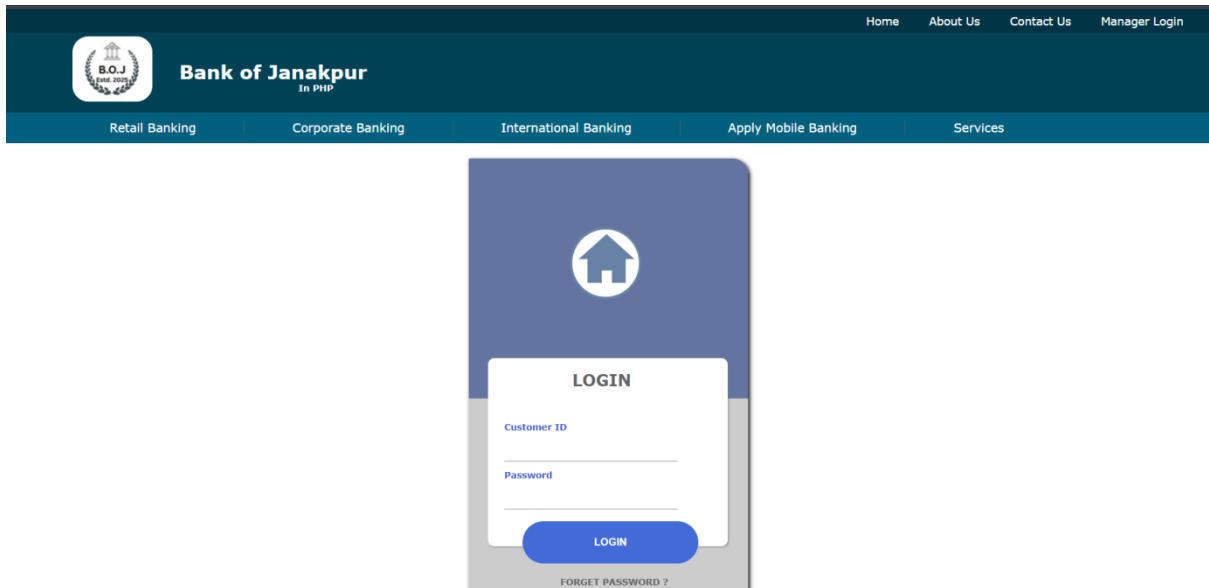




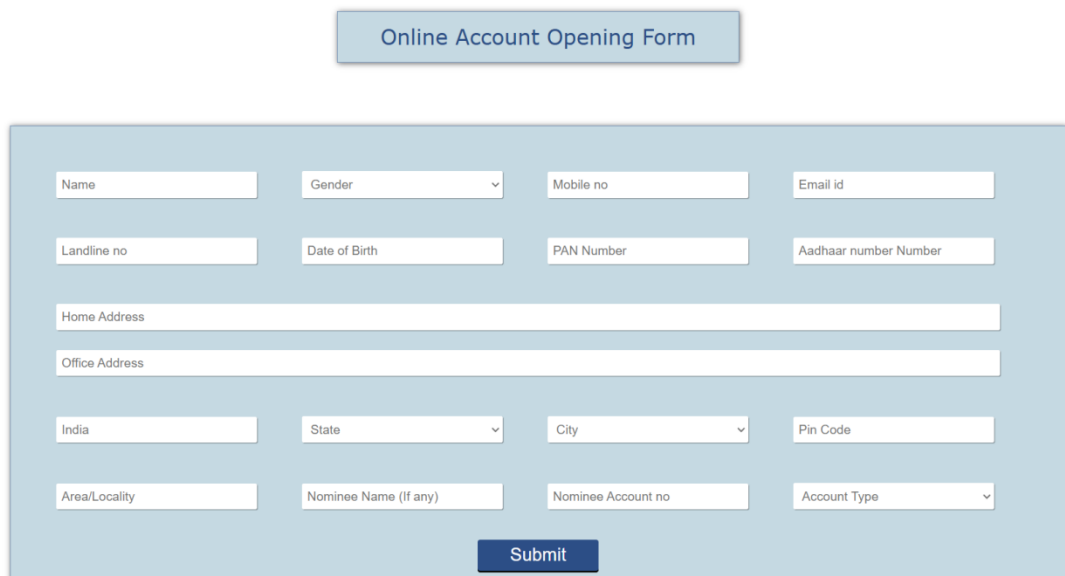
Figure: Database Design Diagram: Entity-Relationship Structure of the Banking Management System

9: OUTPUT SCREENSHOTS



The image shows the top section of the Bank of Janakpur website. The header is dark blue with the bank's logo and name 'Bank of Janakpur In PHP' on the left. Navigation links for 'Home', 'About Us', 'Contact Us', and 'Manager Login' are on the right. Below the header, a teal bar contains links for 'Retail Banking', 'Corporate Banking', 'International Banking', 'Apply Mobile Banking', and 'Services'. The main content area is a light blue box with a house icon and a 'LOGIN' form. The form has fields for 'Customer ID' and 'Password', a 'LOGIN' button, and a 'FORGET PASSWORD?' link.


1. Login Page:- Secure login for all users.



The image shows an 'Online Account Opening Form'. At the top, there is a blue button labeled 'Online Account Opening Form'. Below it is a light blue form box containing various input fields: 'Name', 'Gender' (dropdown), 'Mobile no', 'Email id', 'Landline no', 'Date of Birth', 'PAN Number', 'Aadhaar number Number', 'Home Address', 'Office Address', 'India' (dropdown), 'State' (dropdown), 'City' (dropdown), 'Pin Code', 'Area/Locality', 'Nominee Name (If any)', 'Nominee Account no', and 'Account Type' (dropdown). A blue 'Submit' button is at the bottom right of the form.

2. Add Beneficiary :- Form to register beneficiaries.

Retail Banking
Corporate Banking
International Banking
Apply Mobile Banking
Services



Welcome Sameer Bachhhar
Last login :

Home
Logout

My Account
My Profile
Change Password
Fund Transfer
Statement

Account Details

Customer Id : 1011528

Account Number : 1011851011528

Account Name : Sameer Bachhhar


Account Type : Saving

Available Balance : \$10000.00

Bank Statement

#	Date	Trans. Id	Description	Cr.	Dr.	Total
1	17/04/25 02:10:34 AM	913531067	Cash Deposit/913531067	10000	0	\$10000
2	17/04/25 02:03:52 AM	393944928	Account Opening	0	0	\$0

3. Customer Dashboard:- Displays balance and options.



Welcome Sameer Bachhhar
Last login :

Home
Logout

My Account
My Profile
Change Password
Fund Transfer
Statement

IMPS (24x7 Instant Payment)

Add beneficiary
Delete beneficiary
View beneficiary

Select Beneficiary

Amount

Remark

Send

4. Transfer Money:- Interface to send funds.

Forget Password

Customer ID

Debit Card Number

Debit Card Pin

Registered Mobile no

Submit

5. Password Reset

View Active Customer

View Customer by A/c No

Approve Pending Account

Delete Customer A/c

Credit Customer

6. Staff Panel



Welcome Aman Thakur
Last login : 19/04/25 01:39:52 AM

[Home](#) | [Logout](#)

#	Username	Customer ID	Account No.	Mobile No.	Email ID	DOB
1	Liam Moore	1010112	1011071010112	7415896650	liamoore@gmail.com	1995-02-15
2	William Richards	1011046	1011801011046	7850001250	william@gmail.com	1990-03-15
3	Christine Moore	1011426	1011411011426	7012500010	christine@gmail.com	1995-09-17
4	Romona Weiss	1011742	1011751011742	7014569690	romonaw@gmail.com	1990-10-10
5	Trevor Russo	1011439	1011951011439	7025690001	trusso@gmail.com	1992-01-01
6	Kathryn White	1011768	1011921011768	7016002001	kathrynww@gmail.com	1996-09-25

7. Admin Panel:- View and manage all users.

10: ADVANTAGES & LIMITATIONS

The Online Banking System offers several significant advantages that make it a practical, functional, and secure solution for digital financial operations. One of the most important benefits is the ability to access the banking platform online at any time, allowing users to manage their accounts and perform transactions conveniently 24/7. The project realistically simulates the structure and operations of a real-world banking system, including user registration, balance checks, fund transfers, and transaction histories.

Security features such as OTP-based verification and role-based access controls strengthen the overall reliability of the system. The modular design of the application ensures that each component (such as login, transaction, database, and admin panel) is independent, making the system easier to maintain and debug. These features are crucial for both functionality and long-term scalability.

However, the system also has a few limitations. The absence of integrated SMS or email notifications limits real-time communication with users regarding their transactions. Moreover, since the project is currently designed for desktop browsers, it lacks a dedicated mobile application, reducing accessibility for users who primarily rely on smart phones. Finally, although the system is secure on the surface, the depth of its security depends largely on the server configurations and deployment infrastructure, which may vary based on hosting environment.

Advantages:

1. 24/7 Online Access:

- Users can access the platform from anywhere at any time.
- Promotes flexibility and time-saving for both customers and admins.

2. Realistic Simulation of Banking Activities:

- Includes real-world features like transaction history, fund transfer, and balance inquiry.
- Helps users and developers understand banking workflows.

3. Role-Based Access Control:

- Admins and customers have different permissions.
- Protects sensitive operations from unauthorized users.

4. OTP Verification Mechanism:

- Adds an extra layer of authentication for important operations.
- Enhances user security and system credibility.

5. Modular Architecture:

- Each feature is developed independently.
- Makes maintenance, testing, and debugging easier and faster.
- Improves scalability for future upgrades.

6. Clean and Intuitive User Interface:

- Simplifies navigation for users of all technical levels.
- Enhances user experience with minimal training required.

7. Centralized Database Management:

- Ensures consistent data storage and retrieval.
- Facilitates backup and recovery processes.

8. Cost-Effective Development:

- Built using open-source technologies (e.g., PHP, MySQL).
- Reduces dependency on expensive software licenses.

Limitations:

1. No Integrated SMS or Email Alerts:

- Real-time communication is missing without third-party service APIs.
- Users may not be instantly notified of transactions.

2. Lack of Mobile Application:

- No native Android or iOS app.
- Reduces accessibility for mobile-first users.

3. Basic Security Relies on Deployment Setup:

- Vulnerabilities may arise without proper HTTPS, firewalls, or server-side validations.
- Needs secure hosting and SSL certificates for real-world use.

4. Limited Real-World Integration:

- No direct payment gateway or UPI integration.
- Cannot handle real currency transactions or bank APIs.

5. No Biometric Authentication:

- Does not support fingerprint or facial login for enhanced mobile security.

6. Not Scalable for Enterprise Use:

- Suitable for learning and small demos.
- Needs cloud support and load balancing for larger deployments.

11: FUTURE SCOPE

As technology advances and user expectations grow, the Online Banking System can be enhanced with several innovative features to make it more secure, user-friendly, and scalable. The future scope focuses on adding real-time alerts, multi-platform support, and advanced security measures. Additionally, the integration of Artificial Intelligence and analytics can further improve customer experience and operational efficiency. These additions will not only modernize the platform but also increase its adoption for real-world banking simulation or actual service use.

Detailed Bullet Points:

1. SMS & Email Notifications for Every Transaction:

- Automatically notify customers of debit and credit transactions.
- Improve transparency and user confidence in banking operations.
- Allow users to receive low-balance or high-value transaction alerts.
- Uses SMTP or APIs like Twilio or SendGrid for integration.

2. Two-Factor Authentication (2FA):

- Adds a second layer of security beyond username and password.
- Can be OTP-based (email/SMS) or app-based (e.g., Google Authenticator).
- Protects against unauthorized access and phishing attacks.
- Especially critical for financial operations and password resets.

3. Mobile Application Development (Android/iOS):

- Design a dedicated mobile app with React Native or Flutter.
- Provide real-time access to accounts and instant fund transfers.
- Include biometric login (fingerprint/face recognition) for ease.
- Allow customers to manage accounts anytime, anywhere.

4. Graphical Dashboard with Analytics:

- Provide visual representation of expenses, deposits, and transfers.
- Use charts and graphs (e.g., pie, bar, line) for monthly insights.
- Help users understand financial behavior and make smarter decisions.
- Useful for both customers and administrators to monitor activities.

5. AI-Based Fraud Detection System:

- Use machine learning algorithms to detect suspicious behavior.
- Flag transactions based on amount, time, frequency, or destination.
- Prevent fraudulent access or unusual withdrawals.

- Can improve customer trust and reduce financial risk.

6. Voice Assistant Integration:

- Enable voice-based transactions or queries using Google Assistant or Alexa.
- Example: “What’s my account balance?” or “Transfer ₹1,000 to Ramesh.”
- Useful for visually impaired or elderly customers.

7. Chatbot for Customer Support:

- Implement a smart chatbot using NLP tools (e.g., Dialogflow or Rasa).
- Answer common queries like balance, last transaction, or FAQs.
- Available 24/7, reducing support workload.

8. Auto-Budgeting and Expense Management:

- Automatically categorize spending (groceries, travel, shopping).
- Notify customers when they exceed predefined monthly limits.
- Suggest saving strategies or low-expense days.

9. Multi-Currency & International Transfers:

- Useful for users with international needs or overseas accounts.
- Convert currency in real time and apply exchange rates.
- Allow cross-border transfers with necessary compliance checks.

10. Blockchain-Based Ledger:

- Use blockchain to maintain a secure, tamper-proof transaction record.
- Increase transparency and reduce fraud risk.
- Beneficial in decentralized finance (DeFi) applications.

12: CONCLUSION

The Online Banking System successfully simulates the essential operations of a banking environment using web technologies. It allows multiple users to interact securely based on their role and provides a smooth interface for tasks like fund transfers and transaction tracking. This project helped us understand the development cycle from database modeling to implementation and security practices.

By building this project, we gained practical exposure to PHP-MySQL based full-stack development, system design, authentication techniques, and data management. It stands as a strong example of applying web technologies to solve real world problems in the finance sector.