Frontend for Banking Application

The frontend application for banking is designed to provide customers with a secure, user-friendly digital banking experience. While the backend manages the business logic and data processing through microservices, the frontend deals with/act as a presentation layer, enabling customers to seamlessly interact with the system.

The application provides /offers a modern, responsive web interface built with angular. It communicates with backend services via REST Apis ensuring smooth integration and real-time data flow.

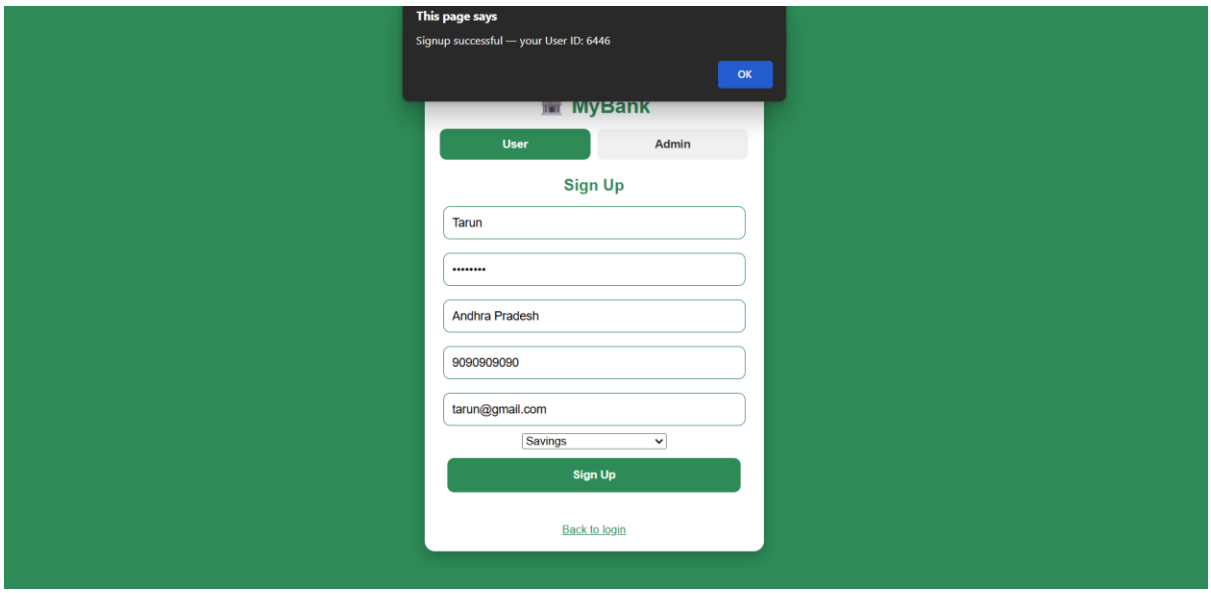
This frontend-framework is established using angular 19, and UI components used are angular material and Bootstrap, CSS….

Frontend modules :

1.User module :

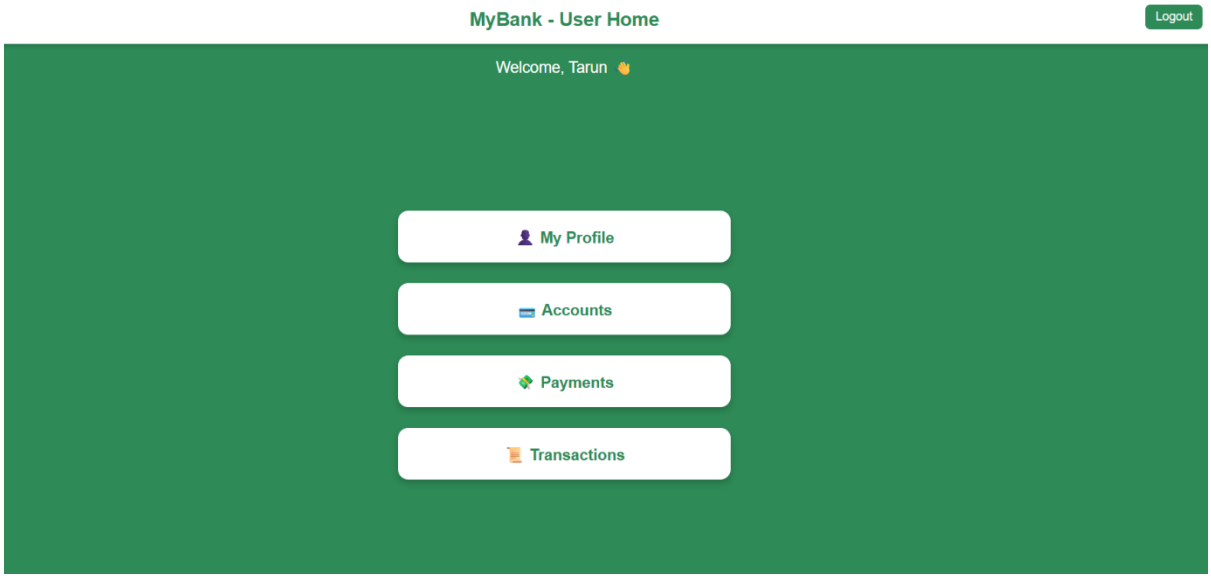
The user module is designed for bank customers to access and manage their accounts. It provides a simple, responsive interface for everyday banking needs. This module integrates with the backend services (login & signup for customers, personal information , payments , transactions) through REST Api , ensuring real time access to bank operations.

User login & Signup page :

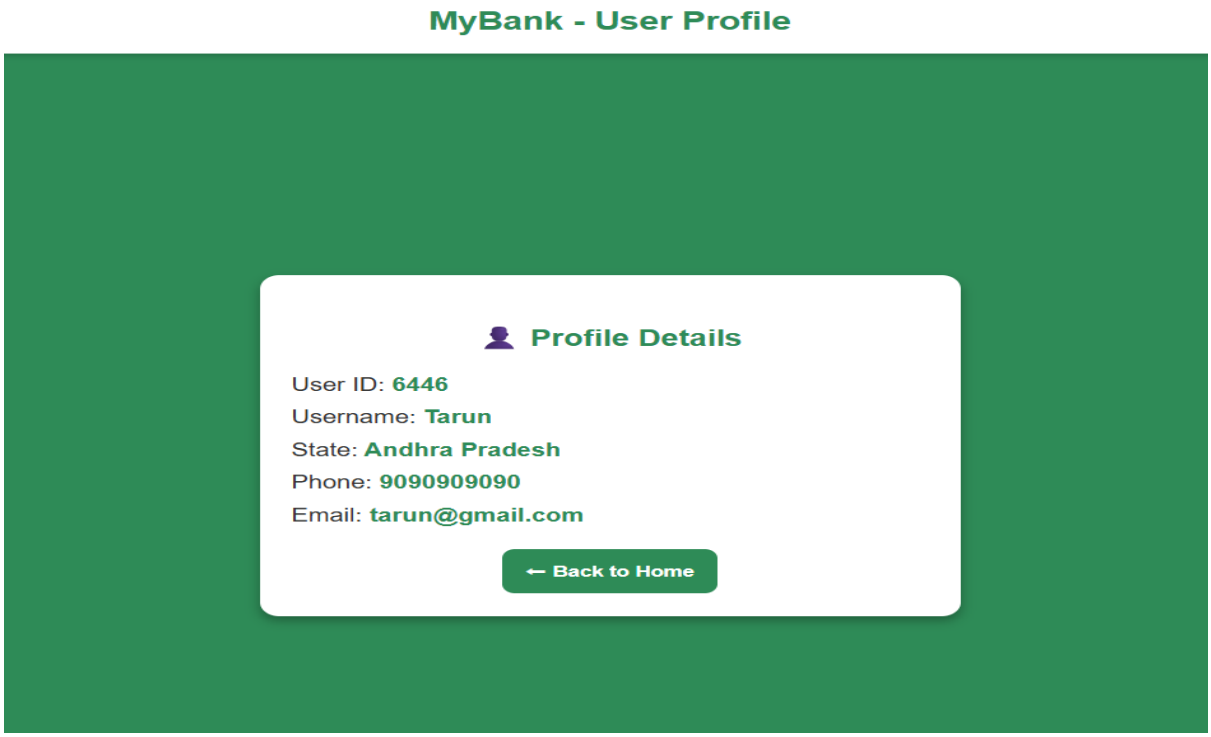


User Dashboard

It contains profile information, account details , payments , transactions.

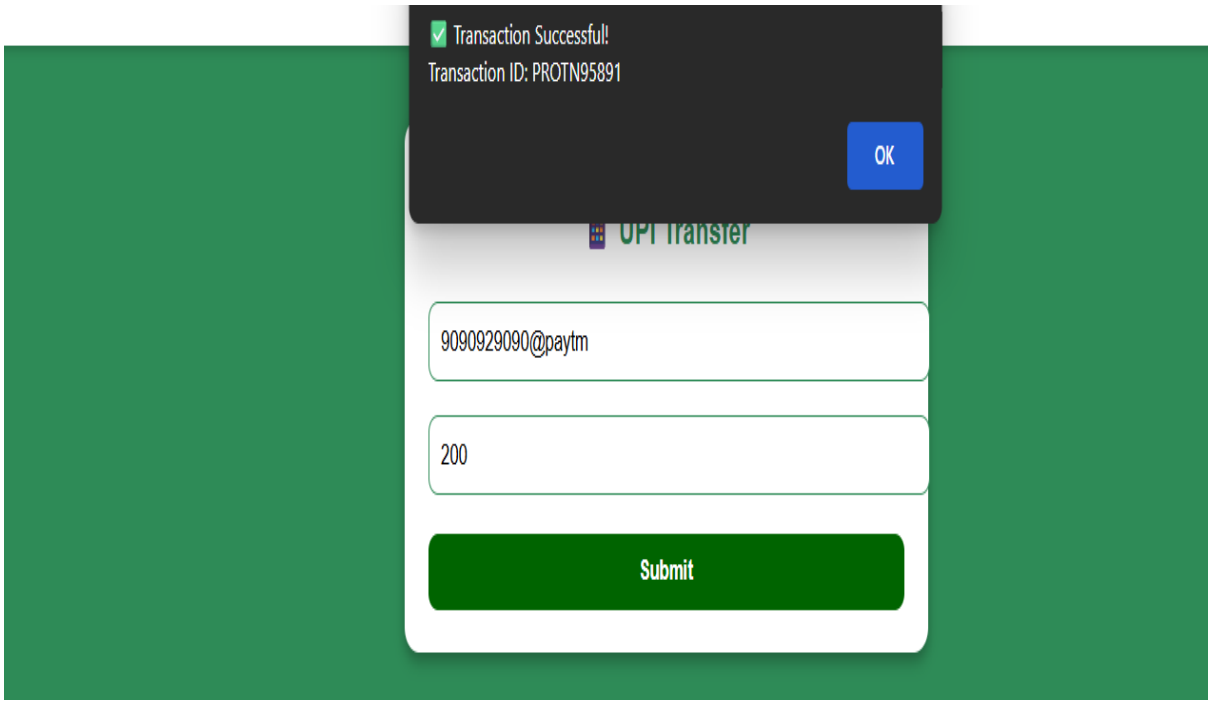


User Profile

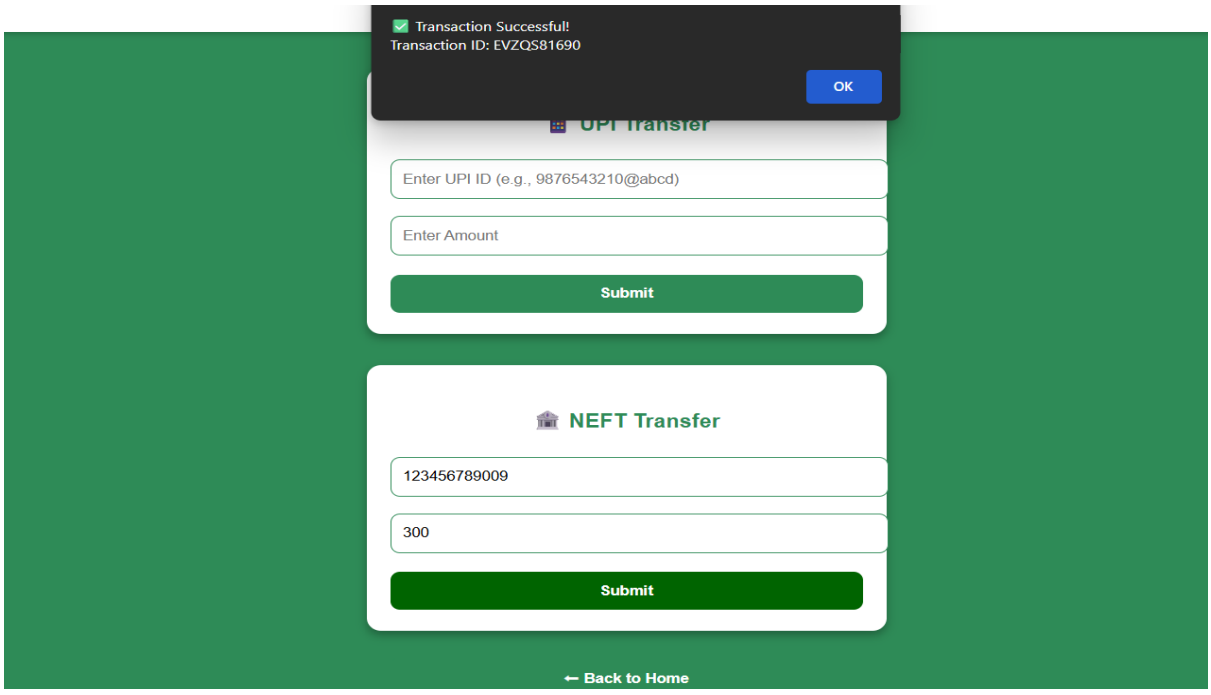


Payments Service

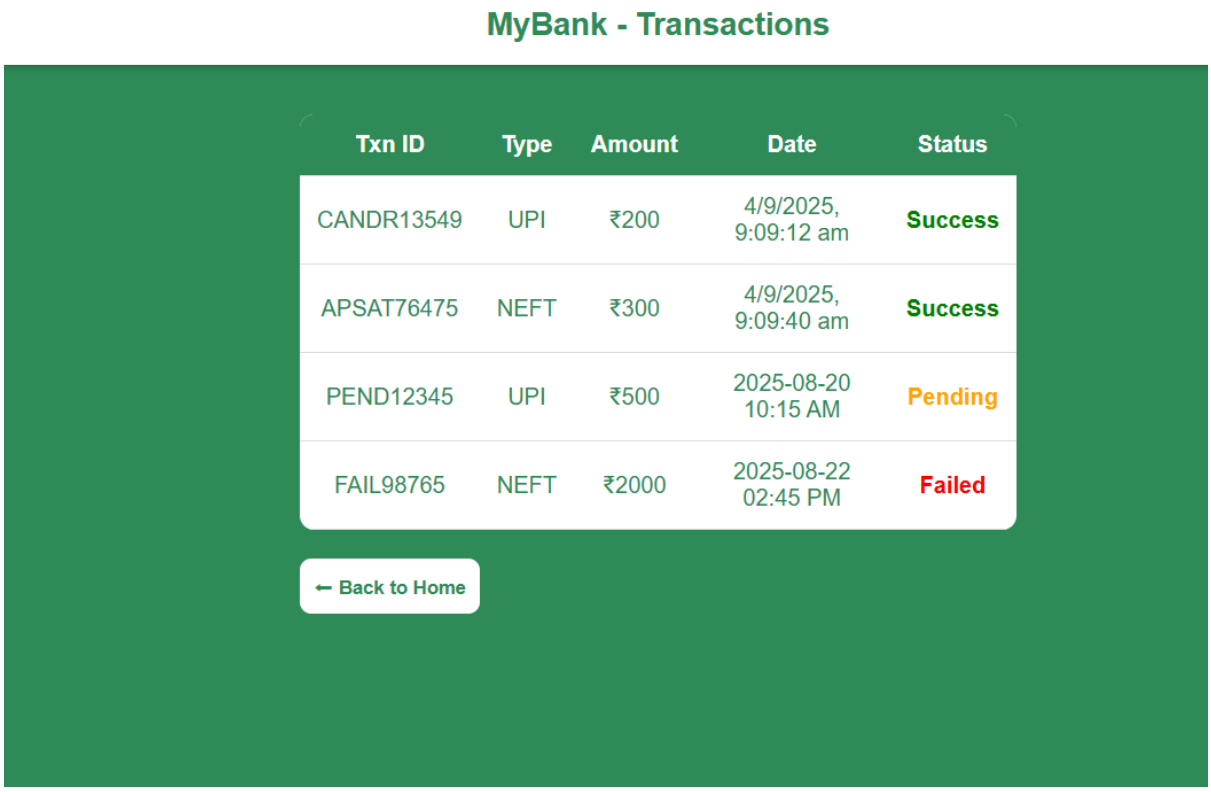
Payment using UPI Transfer



NEFT Transfer



Transactions History



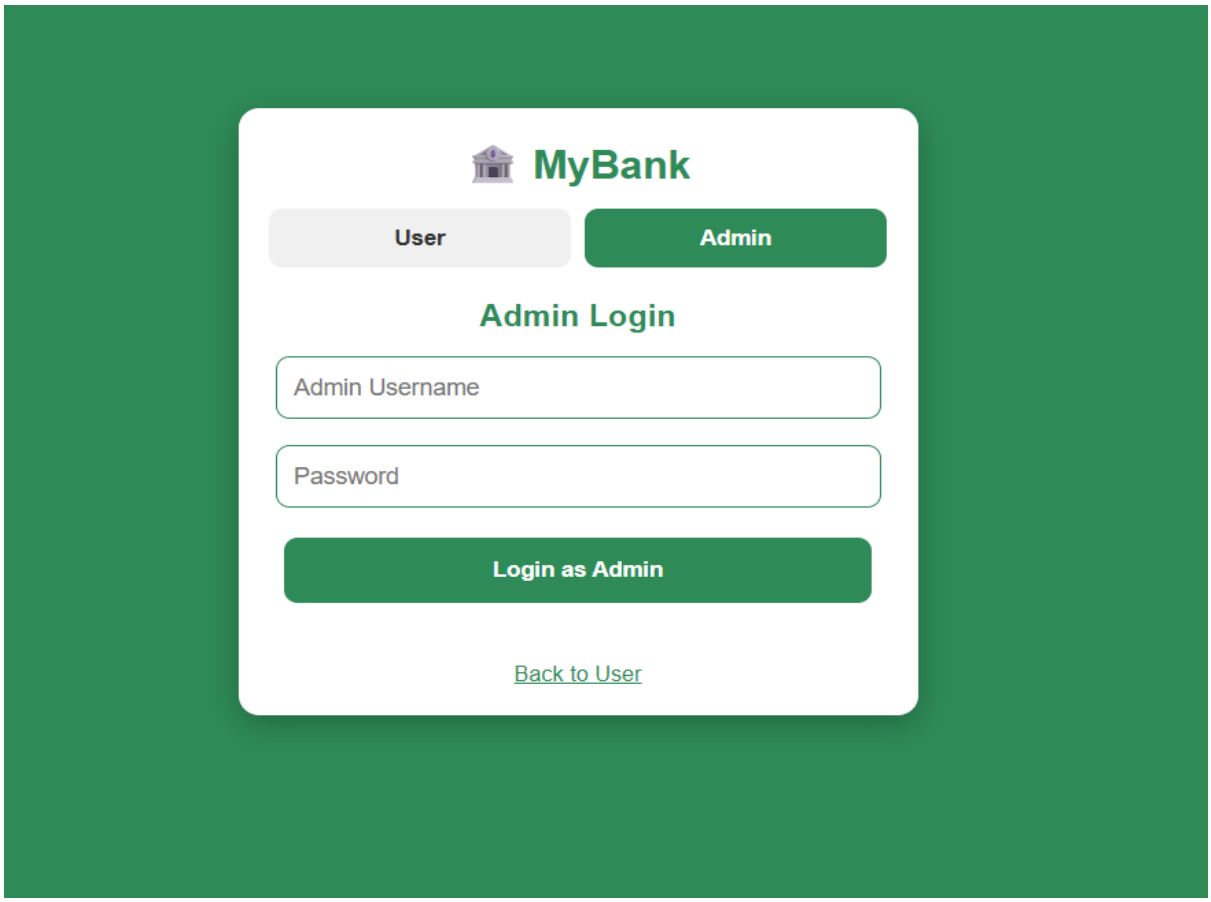
ADMIN Module

The admin module is developed for bank administrators and staff to monitor, manage and control backend operations. It provides operational control over the system .

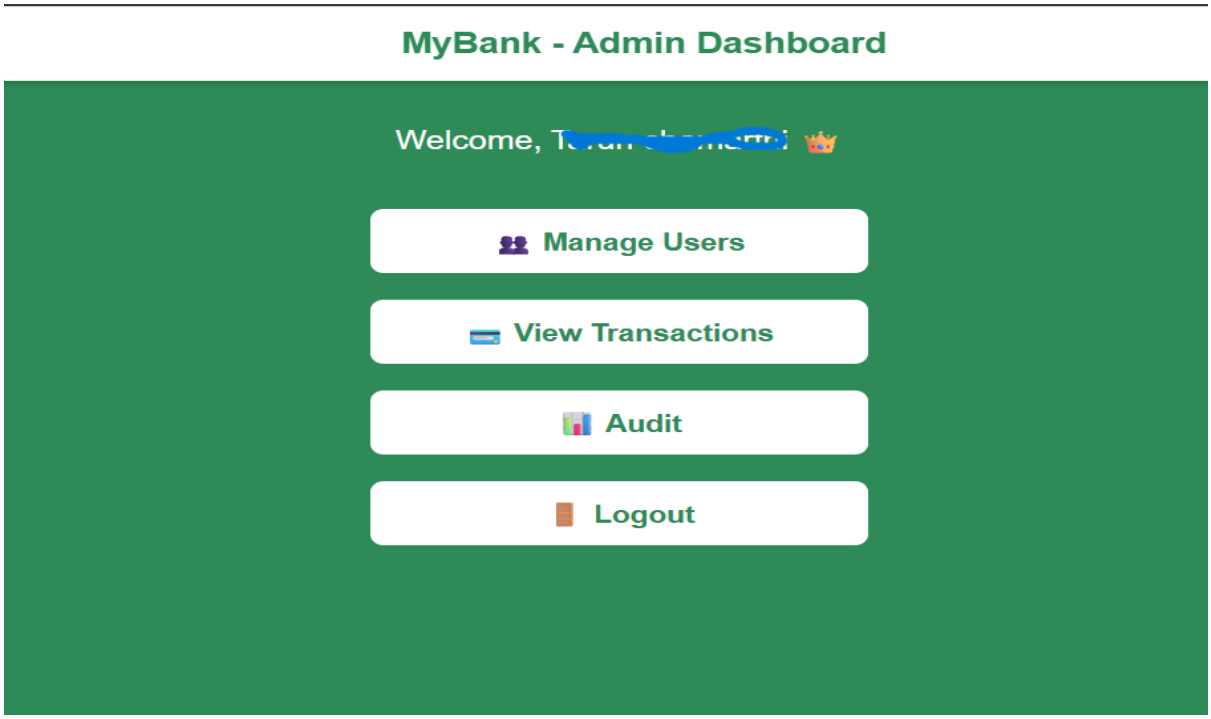
Key features:

- Manage Users – Track account activity, balances.
- View Transactions – Monitor large or suspicious transactions .
- Audit Logs – Generate reports from audit logs

Admin Login

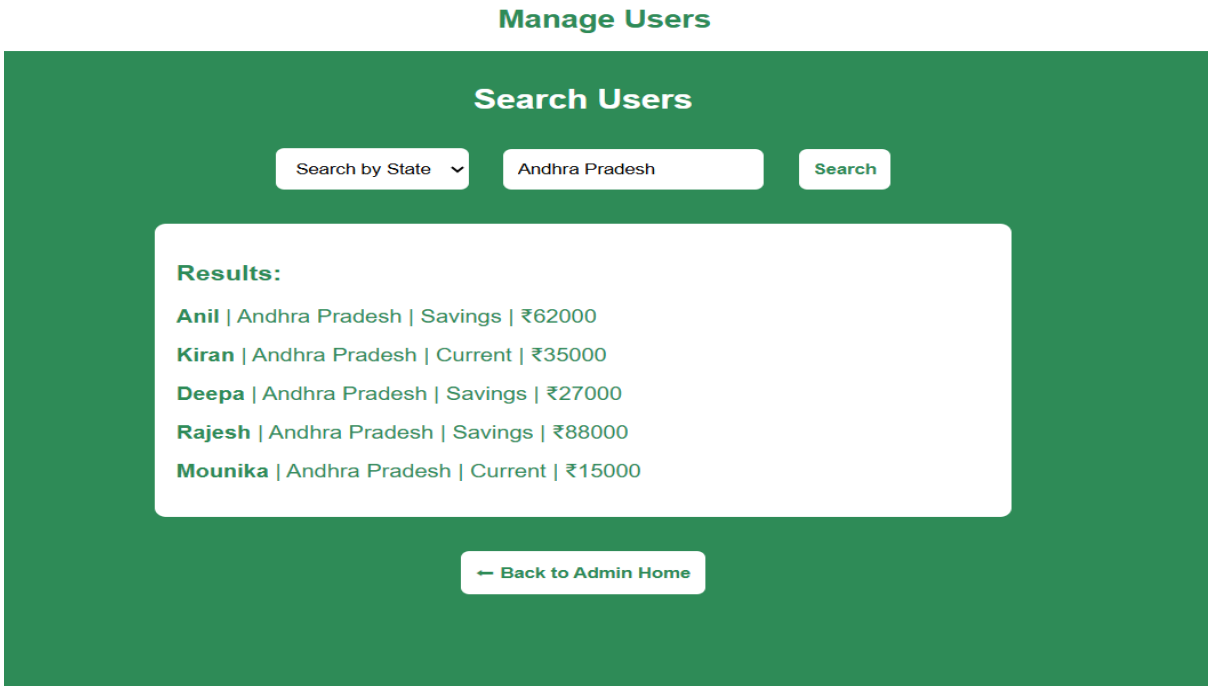


Admin Dashboard

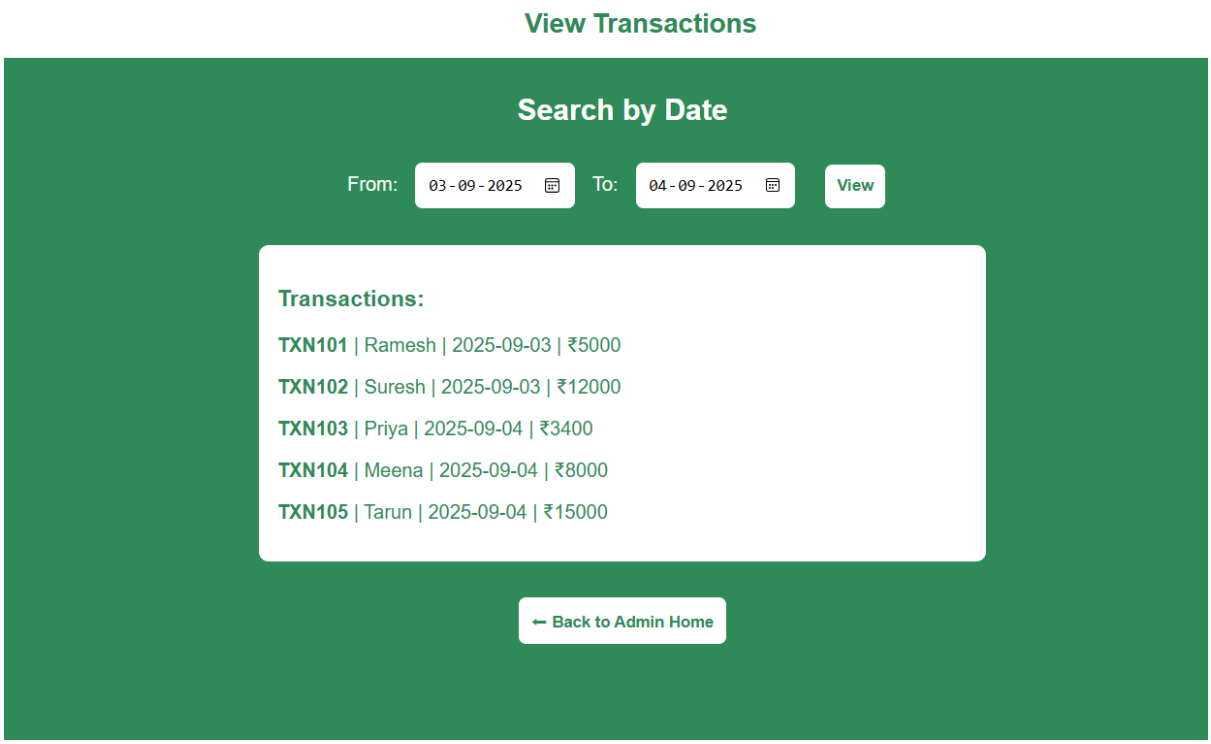


Manage Users :

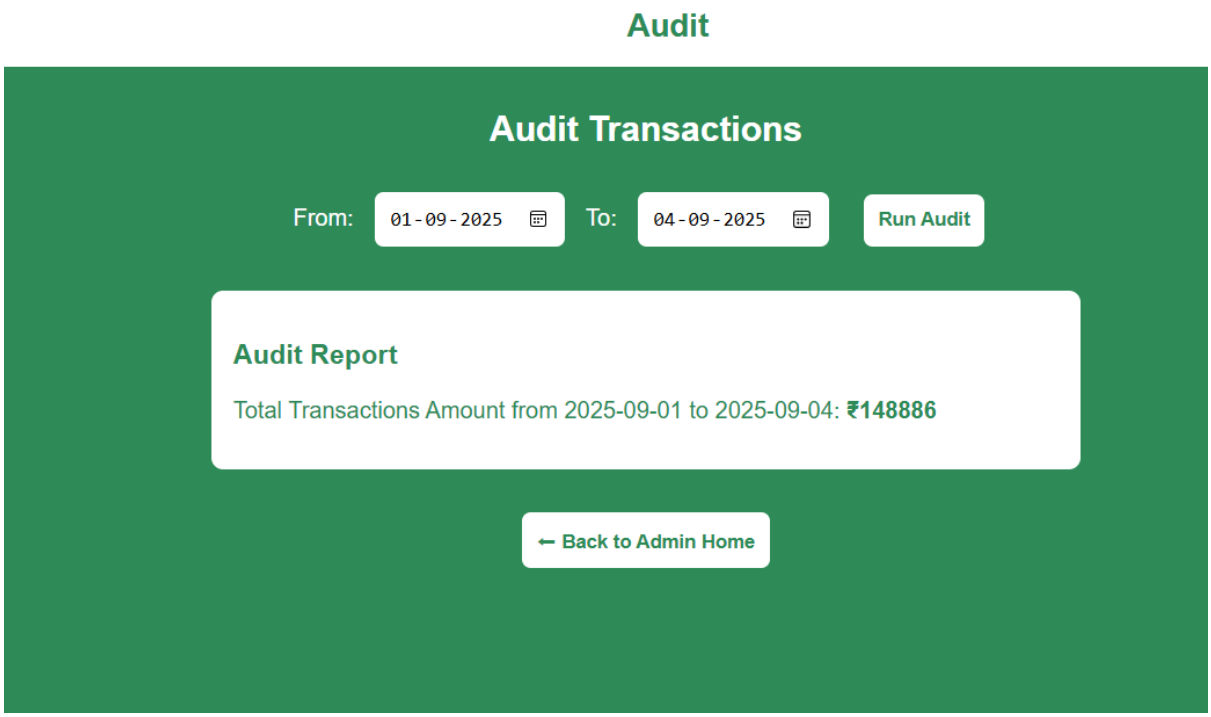
Admin can search users by state and id’s.



View-Transactions



Audit-Logs



Conclusion

The Banking Microservices Application successfully demonstrates how a modern digital banking platform can be built using a microservices architecture for the backend and a modular frontend for both customers and administrators.

On the backend, the system is divided into six independent services (Customer, Account, Transaction, Statement, Notification, and Audit), each designed for scalability, reliability, and fault isolation. Features such as authentication (JWT, Spring Security), event-driven communication (Kafka), and API documentation (Swagger UI & API Docs) ensure the platform is secure, transparent, and integration-ready.

On the frontend, the application is separated into two modules:

User Module, which empowers customers to perform everyday banking operations such as registration, login, fund transfers, viewing transaction history, downloading statements, and receiving notifications.

Admin Module, which equips administrators with tools for user management, transaction monitoring, auditing, and reporting, ensuring compliance and governance.

By combining robust backend services with a responsive, user-friendly frontend, this project provides a complete digital banking experience that is secure, scalable, and future-ready.

Looking ahead, the system can be extended with advanced features such as AI-based fraud detection, third-party payment integrations (UPI, wallets), real-time analytics, and cloud-native deployments to make it enterprise-ready.

In conclusion, the project not only fulfills the essential requirements of a banking application but also serves as a practical blueprint for building scalable, secure, and modern financial systems.