Internship Report

Web Development (MERN Stack)

DLithe Consultancy Services Pvt. Ltd.

Internship Report

Trainee/Intern Name: Sindhoora O

Reg. no: U72900KA2019PTC121035

Period: 4 Months

Job Assignment: Banking application for Loan recommendation based on user Transactions

Organization: DLithe Consultancy Services Pvt. Ltd.

Supervisor's Name: Mr. Purushottam Pattanashetty

Observations:

- The report introduces the banking application model that typically involves analyzing various financial activities of users for recommending loans.
- The various factors that are taken into consideration in loan recommendation models are deposits, withdrawals, account balance patterns, and transaction frequency.
- The various advancements to be made while using AIML in banking applications.

Submitted to

Signature of Training Supervisor

Date: 05/05/2025

Signature of Co-ordinator

Date:05/05/2025

DLithe™
Be Agile..Be in Demand

Letter of Transmittal

To,

Program Co-ordinator
DLithe Consultancy services
Bengaluru

Dear Sir,

I am writing to submit my report on Web development(MERN Stack). The training program was an invaluable learning experience, and I am grateful for the opportunity to participate.

The training program covered various aspects of full-stack development using MongoDB, Express.js, React.js, and Node.js. I gained hands-on experience in designing and developing a functional banking application with both user and admin modules. The user side enables users to perform transactions such as deposits and withdrawals, and based on their financial activity, loan recommendations are made. The admin side allows for viewing user details, transaction history, loan application management, and handling user messages and settings.

The report includes a detailed overview of the training program, including the topics covered, the learning objectives, and the outcomes achieved. It also provides observations and insights into the practical implementation of a full-stack application and how transaction data can be utilized for intelligent decision-making like loan recommendations.

I believe that the knowledge and skills that I acquired during the training program will be valuable to our organization. As digital banking continues to evolve, the ability to build intelligent web applications using MERN Stack and data-driven logic will play a vital role in delivering modern financial services.

I hope that the report provides useful insights into the benefits of on-job training and the potential of intelligent web applications.

Sincerely,

Name: Sindhoora O

Table of Contents

1. INTRODUCTION

Banking plays a critical role in the financial stability of individuals and the economy. With the growing demand for digital financial services, understanding user transactions has become essential in offering efficient and personalized banking solutions. Loan recommendation is a key functionality in banking systems that helps users access financial support based on their account activities. Accurate evaluation of deposits, withdrawals, and transaction history is crucial for determining loan eligibility and maintaining financial transparency.

Loan recommendation is one of the essential but challenging tasks in banking applications. Since it depends on multiple financial factors such as transaction patterns, balance consistency, and user behavior, implementing an effective recommendation system requires handling various types of data. This makes loan recommendation a structured yet detailed process that must ensure accuracy, fairness, and responsiveness. While traditional systems relied on manual processing, modern web-based platforms now allow for real-time processing and tracking of financial data to support better decision-making.

The application was developed using the MERN Stack—MongoDB for database management, Express.js and Node.js for server-side functionality, and React.js for building a dynamic user interface. The system consists of user and admin modules, where users can perform transactions and view loan recommendations, while the admin can manage user data, review transactions, and handle loan applications efficiently.

2. LITERATURE SURVEY

All the published papers and research articles cover the following points. These survey aim to identify the current state of the field, key methodologies, challenges, and potential areas for improvement. They typically includes:

1. Introduction to Web Applications in Banking: Exploring literature on the evolution of web-based banking applications, focusing on the transition from traditional to digital banking. It discusses how user transactions are handled in online systems .

2. Existing Loan Recommendation Systems: Review of existing loan recommendation systems focusing on rule-based decisions and user financial profiling based on transaction behavior.

3. Transaction-Based User Analysis: Studies on how user transactions such as deposits and withdrawals are used for creditworthiness evaluation and financial pattern analysis.

4. Web Technologies for Banking Applications: Overview of MERN stack technologies enabling frontend-backend integration, real-time data handling, and secure database management.

5. Role-Based Access Systems: Research on separating admin and user functionalities to enhance data privacy, control, and user interface simplicity.

6. Security and Data Handling: Discussion of challenges in secure user data handling, including encryption, transaction security, and prevention of unauthorized access.

7. Challenges in Traditional Banking Systems: Limitations of manual loan evaluation and inefficient customer service, emphasizing the need for automated digital solutions.

8. Future Directions: Future trends in banking applications include smarter loan recommendations, better user interfaces, enhanced admin control, and integration with external services.

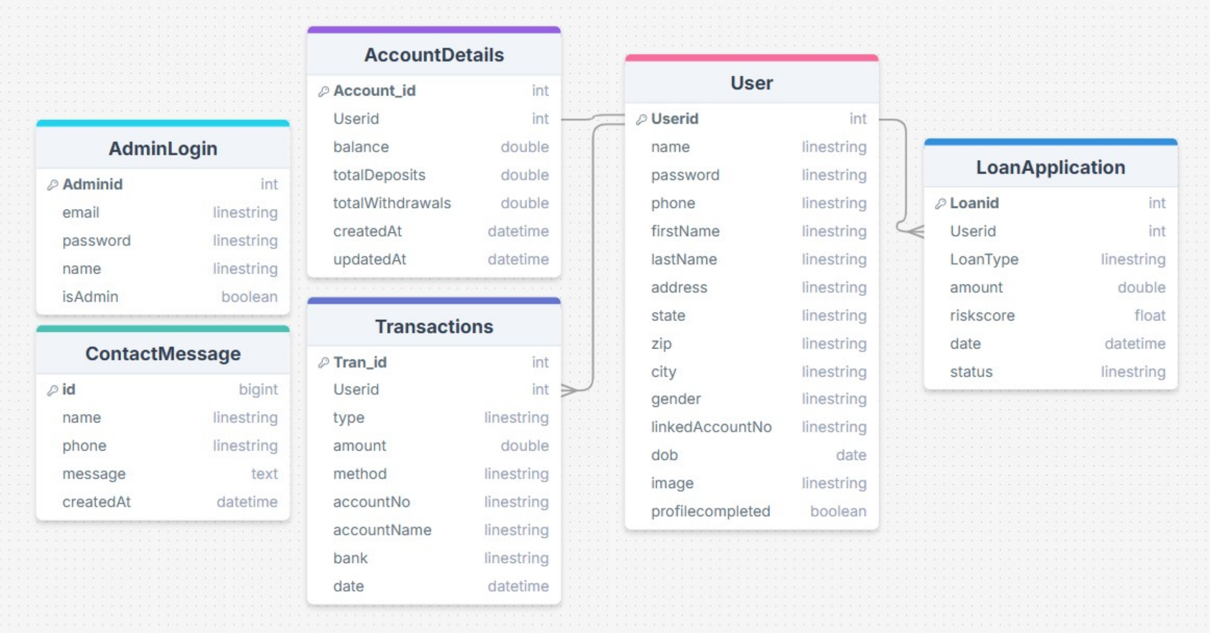
3. PROPOSED WORK

One of the key challenges in the Banking Application project is designing an efficient system that manages user transactions and provides accurate loan recommendations based on transaction history. The project includes creating a database schema, defining workflows, and designing the user interface to ensure smooth operation and user experience.

The system consists of two main modules: the User side and the Admin side. Users can perform deposits and withdrawals, and based on their transaction history, the system recommends suitable loan options. The Admin has access to all user details, transaction records, and loan applications, which can be approved or rejected.

To illustrate the overall design and flow of the application, we have created several diagrams and models:

- **Database Design:** Structured collections in MongoDB to store user profiles, transactions, loan applications, and admin data efficiently.



AdminLogin	
Adminid	int
email	linestring
password	linestring
name	linestring
isAdmin	boolean

ContactMessage	
id	bigint
name	linestring
phone	linestring
message	text
createdAt	datetime

AccountDetails	
Account_id	int
Userid	int
balance	double
totalDeposits	double
totalWithdrawals	double
createdAt	datetime
updatedAt	datetime

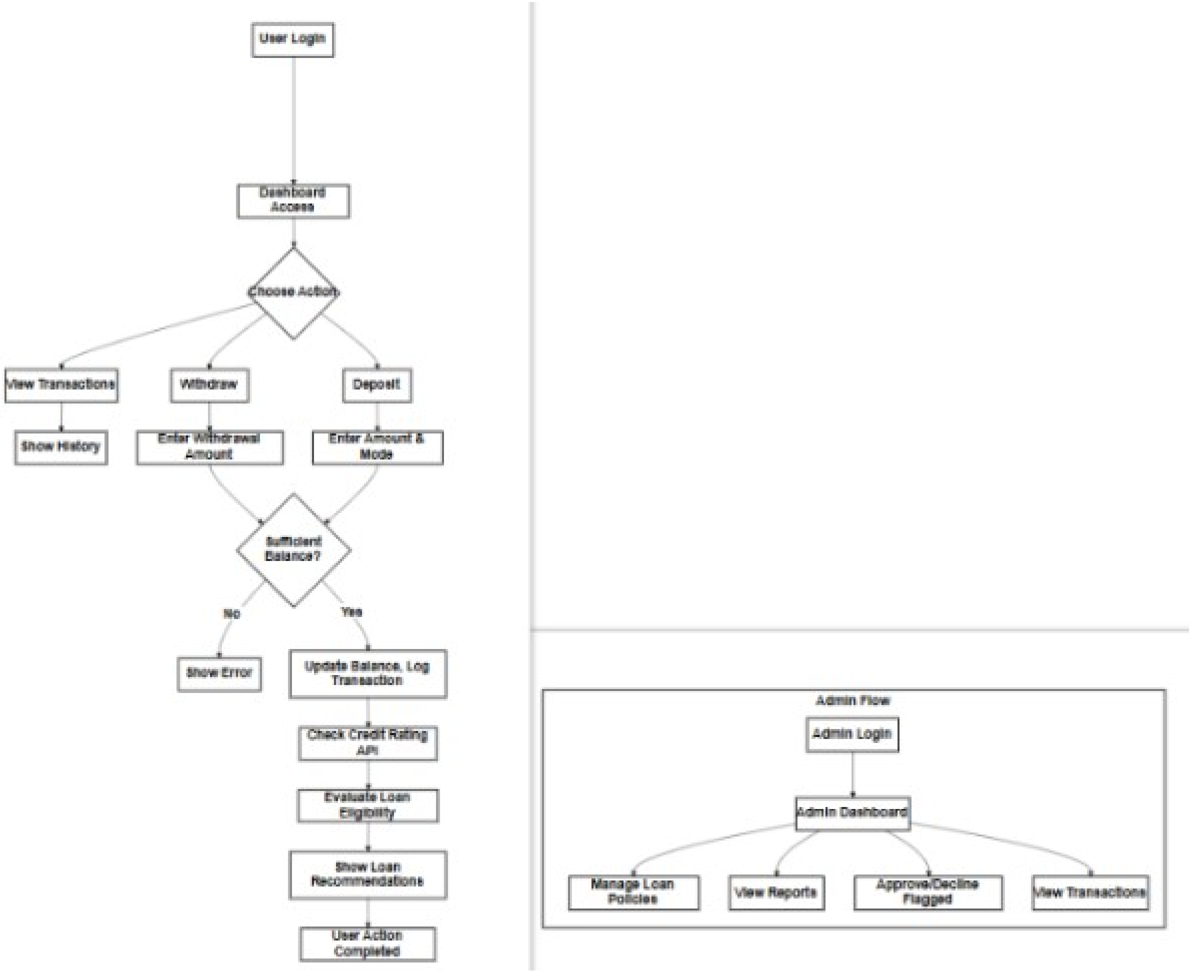
Transactions	
Tran_id	int
Userid	int
type	linestring
amount	double
method	linestring
accountNo	linestring
accountName	linestring
bank	linestring
date	datetime

User	
Userid	int
name	linestring
password	linestring
phone	linestring
firstName	linestring
lastName	linestring
address	linestring
state	linestring
zip	linestring
city	linestring
gender	linestring
linkedAccountNo	linestring
dob	date
image	linestring
profilecompleted	boolean

LoanApplication	
Loanid	int
Userid	int
LoanType	linestring
amount	double
riskscore	float
date	datetime
status	linestring

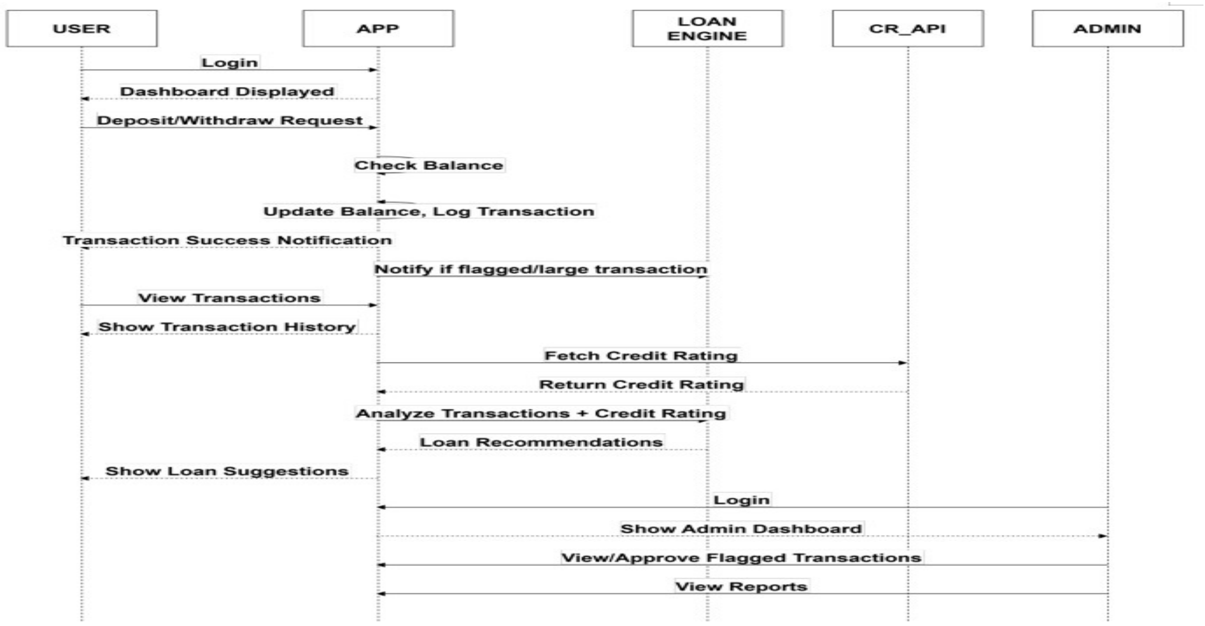
Database Design

- **Workflow Diagram:** Visual representation of user actions, transaction processing, and loan recommendation logic.



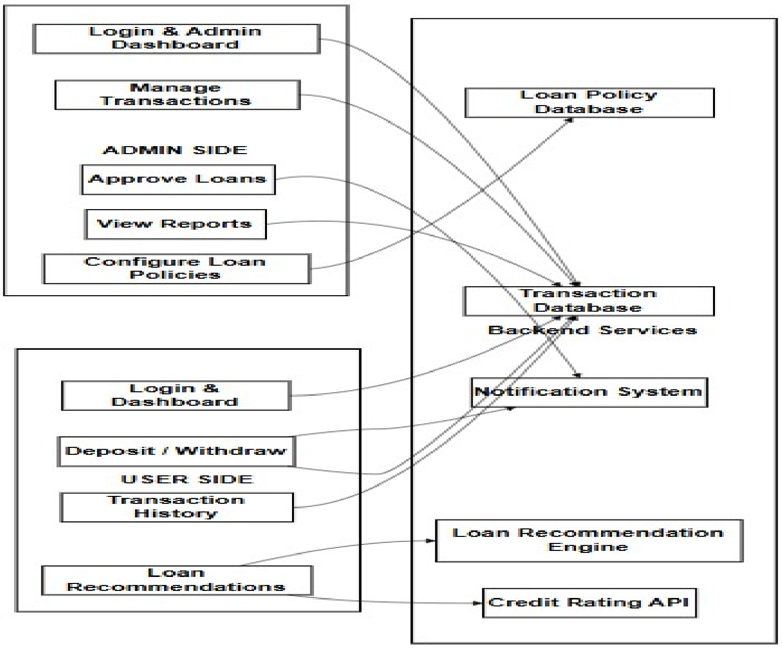
Workflow Diagram

- **Sequence Diagram:** Depicts the interaction between users, admin, and the backend system during key operations like loan application and approval.



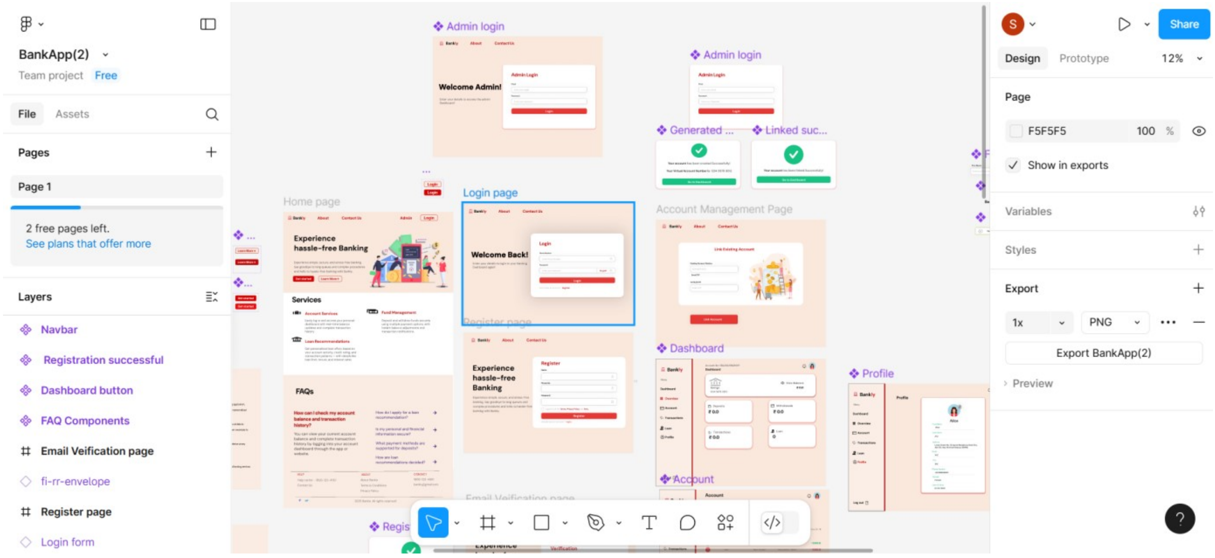
Sequence Diagram

- **Functional Architecture:** Shows the components of the MERN stack and how they integrate to support the application's functionality.



Functional Achitecture Diagram

- **Figma Design:** UI/UX prototype demonstrating the layout and navigation for both user and admin interfaces.



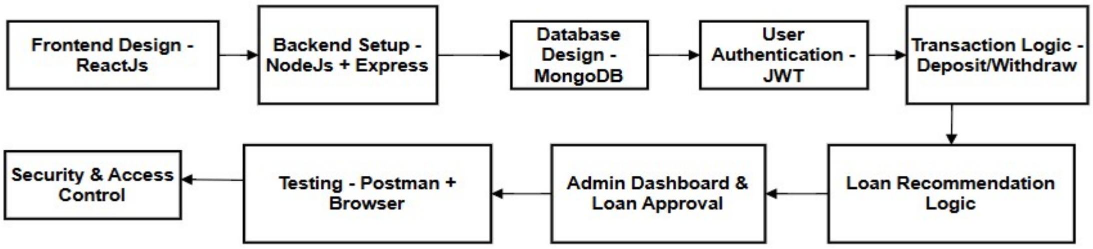
Figma Design

The loan recommendation logic is based on predefined criteria analyzing the user’s transaction patterns, including frequency and volume of deposits and withdrawals. This helps the system determine eligibility and suggest loan amounts accordingly.

Here is an overview of the proposed process:

- Step 1: Users perform transactions such as deposits and withdrawals.
- Step 2: Transaction data is stored and updated in the MongoDB database.
- Step 3: The system analyzes transaction history to generate loan recommendations dynamically.
- Step 4: Users can apply for loans based on these recommendations.
- Step 5: Admin reviews loan applications, verifies user data, and approves or rejects the requests.

4. IMPLEMENTATION

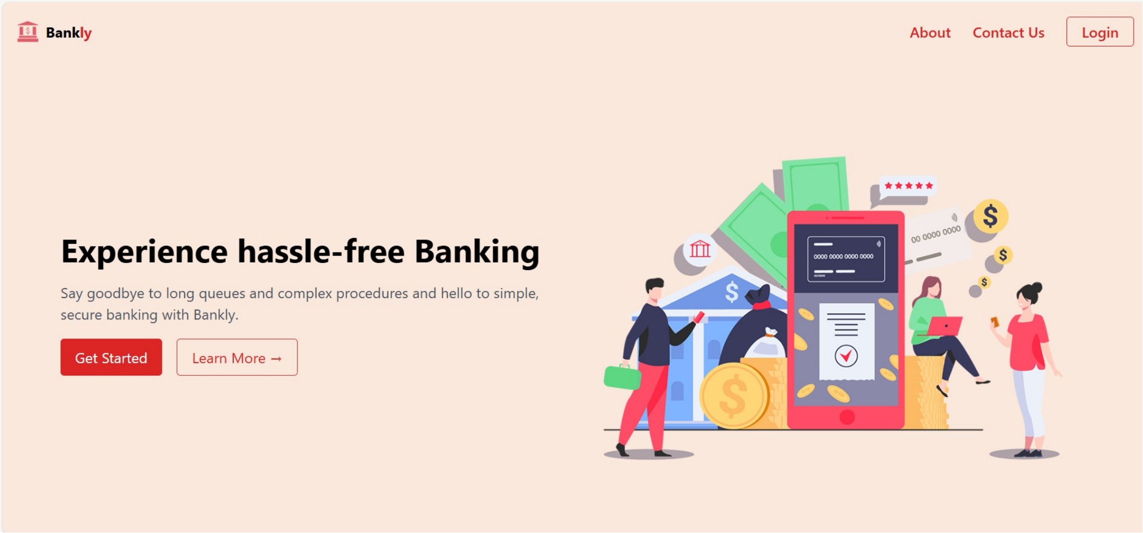

Frontend Design - ReactJs
Backend Setup - NodeJs + Express
Database Design - MongoDB
User Authentication - JWT
Transaction Logic - Deposit/Withdraw
Loan Recommendation Logic
Admin Dashboard & Loan Approval
Testing - Postman + Browser
Security & Access Control

1. Database Design: A MongoDB database was used for storing user data, transaction records, and loan application details. Collections were structured for Users, Transactions, Admin, and Loans, ensuring normalized and scalable data management.
2. Frontend and UI Design: The interface was designed using Figma, then implemented using React.js. User-friendly forms and dashboards were created for both the user and admin, allowing for smooth navigation and functionality like login, deposits, withdrawals, and loan applications.
3. Backend Development: Node.js with Express.js was used to handle API requests, perform server-side operations, and manage authentication and data routing. Backend endpoints were created for handling user registration, login, transaction operations, and loan management.
4. Loan Recommendation Logic: Loan recommendations are based on transaction frequency and volume. If a user’s deposit/withdrawal history meets predefined criteria, the system recommends suitable loan options. This logic was implemented using conditional checks and rules in the backend.
5. Workflow and Sequence Implementation: The full workflow was built with the following process: user actions (deposit, withdrawal) → transaction logging → data analysis → loan suggestion → admin approval system. This process flow was implemented based on the workflow and sequence diagrams created during design.
6. Admin Dashboard Functionality: The admin panel allows viewing of all users, their transaction summaries, and pending loan requests. Admins can approve or reject loans, and send messages or status updates to users, all through secure endpoints.
7. Testing and Validation: The application was tested using Postman and browser-based

manual testing. Transactions, logins, and loan logic were validated to ensure proper working conditions across different users.

8. Security and Access Control: Authentication was handled using JWT (JSON Web Tokens) to secure user sessions. Role-based access ensured that only admins could manage users and loan approvals.

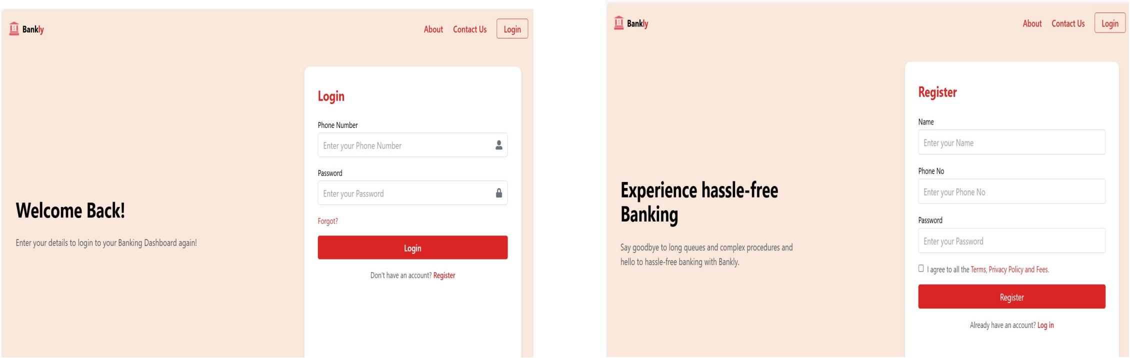
5. RESULTS



Experience hassle-free Banking

Say goodbye to long queues and complex procedures and hello to simple, secure banking with Bankly.

Figure 1: Landing page of Bankly



Welcome Back!

Enter your details to login to your Banking Dashboard again!

Login

Phone Number: Enter your Phone Number

Password: Enter your Password

Don't have an account? Register

Experience hassle-free Banking

Say goodbye to long queues and complex procedures and hello to hassle-free banking with Bankly.

Register

Name: Enter your Name

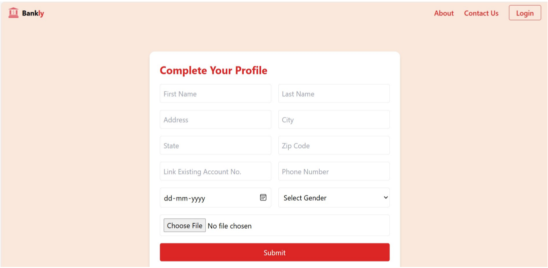
Phone No: Enter your Phone No

Password: Enter your Password

□ I agree to all the Terms, Privacy Policy and Fees.

Already have an account? Log in

Figure 2: Register and Login page for a user of Bankly



Complete Your Profile

First Name | Last Name

Address | City

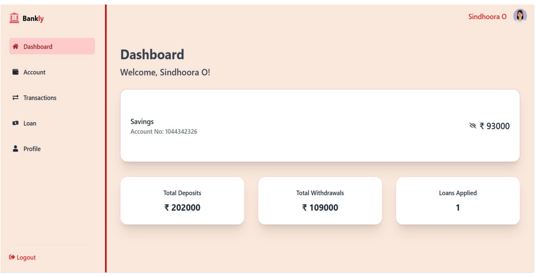
State | Zip Code

Link Existing Account No. | Phone Number

dd-mm-yyyy | Select Gender

No file chosen

Figure 3: Profile Setup for a User



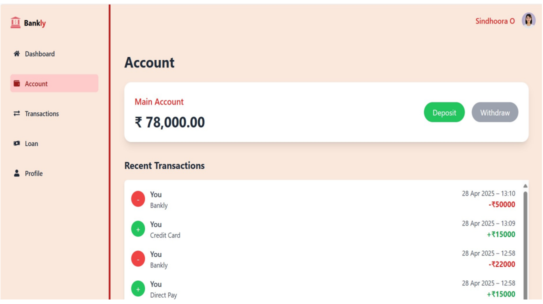
Dashboard

Welcome, Sindhoora O!

Savings
Account No: 1044342326
₹ 93000

Total Deposits	Total Withdrawals	Loans Applied
₹ 202000	₹ 109000	1

Figure 4: User Dashboard



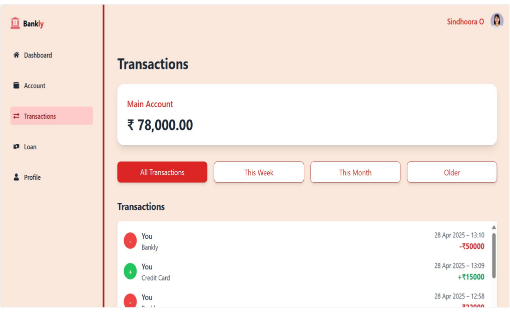
Account

Main Account
₹ 78,000.00

Recent Transactions

-	You Bankly	28 Apr 2025 – 13:10 -₹50000
+	You Credit Card	28 Apr 2025 – 13:09 +₹15000
-	You Bankly	28 Apr 2025 – 12:58 -₹22000
+	You Direct Pay	28 Apr 2025 – 12:58 +₹15000

Figure 5: User Account section



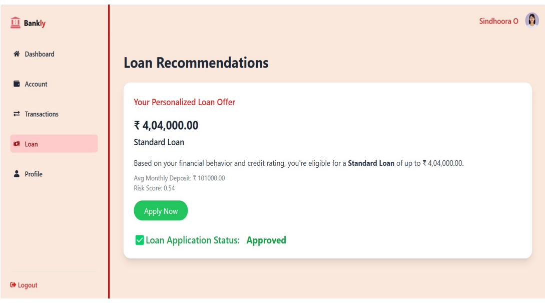
Transactions

Main Account
₹ 78,000.00

Transactions

-	You Bankly	28 Apr 2025 – 13:10 -₹50000
+	You Credit Card	28 Apr 2025 – 13:09 +₹15000
-	You	28 Apr 2025 – 12:58

Figure 6: User Transactions section



Loan Recommendations

Your Personalized Loan Offer

₹ 4,04,000.00

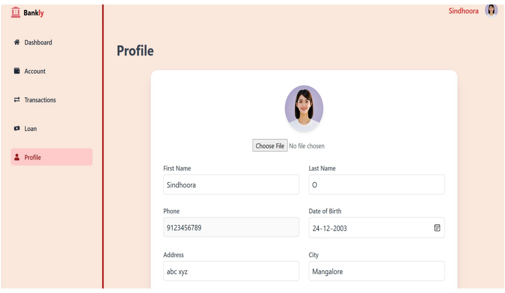
Standard Loan

Based on your financial behavior and credit rating, you're eligible for a **Standard Loan** of up to ₹ 4,04,000.00.

Avg Monthly Deposit: ₹ 101000.00
Risk Score: 0.54

☑ Loan Application Status: **Approved**

Figure 7: User Loan Recommendations

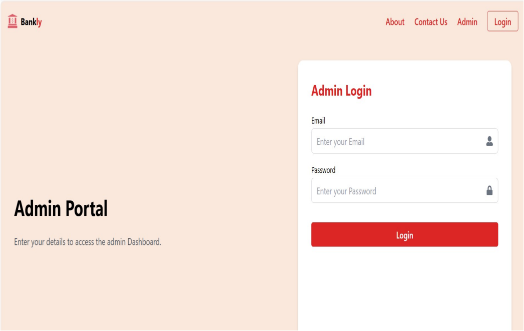


Profile

No file chosen

First Name Sindhoora	Last Name O
Phone 9123456789	Date of Birth 24-12-2003
Address abc xyz	City Mangalore

Figure 8: User profile



Admin Portal

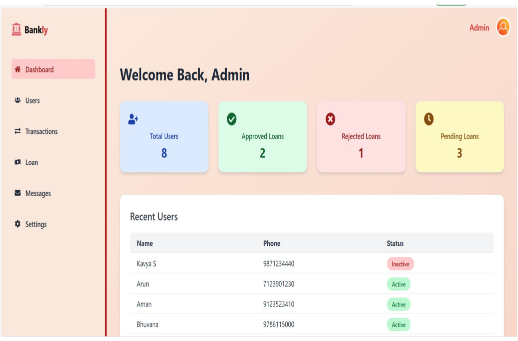
Enter your details to access the admin Dashboard.

Admin Login

Email
Enter your Email

Password
Enter your Password

Figure 9: Admin Login



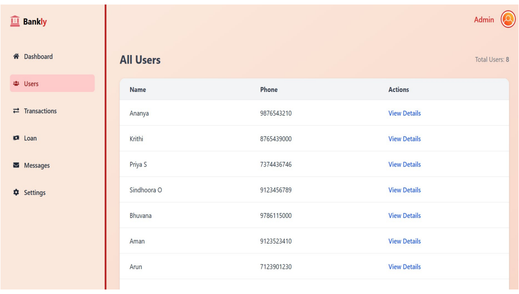
Welcome Back, Admin

Total Users	Approved Loans	Rejected Loans	Pending Loans
8	2	1	3

Recent Users

Name	Phone	Status
Kavya S	9871234440	Inactive
Arun	7123901230	Active
Aman	9123523410	Active
Bhuvana	9786115000	Active

Figure 10: Admin Dashboard

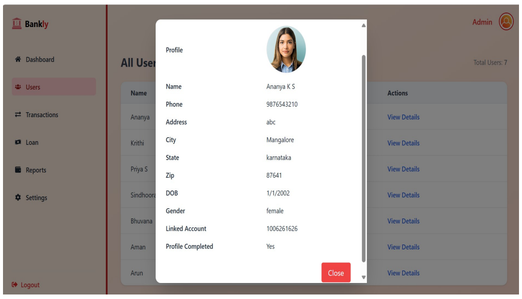


All Users

Total Users: 8

Name	Phone	Actions
Ananya	9876543210	View Details
Krithi	8765439000	View Details
Priya S	7374436746	View Details
Sindhoora O	9123456789	View Details
Bhuvana	9786115000	View Details
Aman	9123523410	View Details
Arun	7123901230	View Details

Figure 11: All Users - Admin



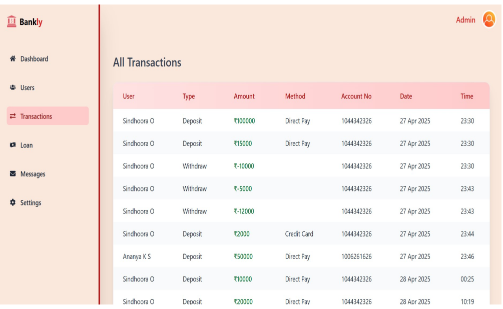
All User
Total Users: 7

Name	Actions
Ananya	View Details
Krithi	View Details
Priya S	View Details
Sindhoora	View Details
Bhuvana	View Details
Aman	View Details
Arun	View Details

Profile

Name	Ananya K S
Phone	9876543210
Address	abc
City	Mangalore
State	karnataka
Zip	87641
DOB	1/1/2002
Gender	female
Linked Account	1006261626
Profile Completed	Yes

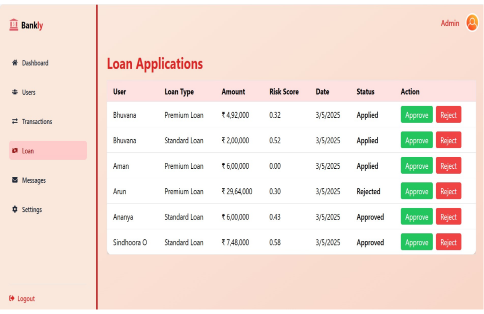
Figure 12: User details



All Transactions

User	Type	Amount	Method	Account No	Date	Time
Sindhoora O	Deposit	₹100000	Direct Pay	1044342326	27 Apr 2025	23:30
Sindhoora O	Deposit	₹15000	Direct Pay	1044342326	27 Apr 2025	23:30
Sindhoora O	Withdraw	₹-10000		1044342326	27 Apr 2025	23:30
Sindhoora O	Withdraw	₹-5000		1044342326	27 Apr 2025	23:43
Sindhoora O	Withdraw	₹-12000		1044342326	27 Apr 2025	23:43
Sindhoora O	Deposit	₹2000	Credit Card	1044342326	27 Apr 2025	23:44
Ananya K S	Deposit	₹50000	Direct Pay	1006261626	27 Apr 2025	23:46
Sindhoora O	Deposit	₹10000	Direct Pay	1044342326	28 Apr 2025	00:25
Sindhoora O	Deposit	₹20000	Direct Pay	1044342326	28 Apr 2025	10:19

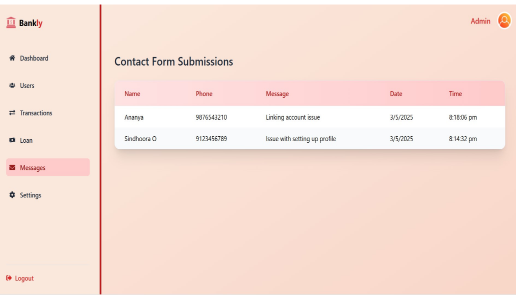
Figure 13: All transactions - Admin



Loan Applications

User	Loan Type	Amount	Risk Score	Date	Status	Action
Bhuvana	Premium Loan	₹ 4,92,000	0.32	3/5/2025	Applied	Approve Reject
Bhuvana	Standard Loan	₹ 2,00,000	0.52	3/5/2025	Applied	Approve Reject
Aman	Premium Loan	₹ 6,00,000	0.00	3/5/2025	Applied	Approve Reject
Arun	Premium Loan	₹ 29,64,000	0.30	3/5/2025	Rejected	Approve Reject
Ananya	Standard Loan	₹ 6,00,000	0.43	3/5/2025	Approved	Approve Reject
Sindhoora O	Standard Loan	₹ 7,48,000	0.58	3/5/2025	Approved	Approve Reject

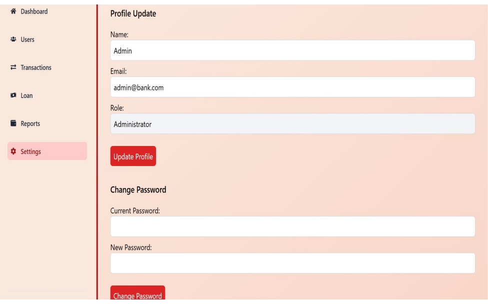
Figure 14: Applied Loan details



Contact Form Submissions

Name	Phone	Message	Date	Time
Ananya	9876543210	Linking account issue	3/5/2025	8:18:06 pm
Sindhoora O	9123456789	Issue with setting up profile	3/5/2025	8:14:32 pm

Figure 15: Messages/Queries



Profile Update

Name: Admin

Email: admin@bank.com

Role: Administrator

Change Password

Current Password:

New Password:

Figure 16: Admin Settings

6. CONCLUSION

The development of this banking application involved designing an efficient and scalable system using the MERN stack. Starting from database modeling and frontend prototyping to implementing secure backend logic, the project went through structured stages of development.

The final system successfully allows users to perform basic banking operations and receive automated loan recommendations based on their transaction behavior. Admins can manage users and oversee the loan application process. With structured workflow, real-time processing, and secure data handling, the system meets its goal of enhancing loan recommendation accuracy and improving decision-making.

In future versions, features such as real-time analytics, notification systems, credit score integration, and AI-based recommendation systems can further enhance the application.

7. REFERENCES

- Roadmap.sh. (n.d.). *Developer Roadmaps*. https://roadmap.sh
- Mongoose. (n.d.). *Mongoose ODM v7.6.1 Documentation*. https://mongoosejs.com/docs/

- npm Documentation. (n.d.). *npm Docs*. http://docs.npmjs.com
- Express.js Official Docs. *Express Web Framework*. https://expressjs.com/