Loan Management System Mini-Project

Objective: Develop a Loan Management System with MERN stack that allows for the management of loans, borrowers, repayments, interest rates, transactions, and contracts.

Functional Requirements:

1. Borrower Management:

○ Implement a CRUD (Create, Read, Update, Delete) system for borrowers.

○ Each borrower must have details such as Full Name, Contact Info (Phone, Email), Address, and Identification Number (e.g., NRC).

○

2. Loan Management:

○ Allow the creation of loans with the following attributes:

■ Loan ID (Auto-generated)

■ Borrower ID (Linked to a borrower)

■ Loan Amount

■ Loan Type (e.g., Personal, Mortgage)

■ Loan Start Date and End Date

■ Interest Rate

○ Ensure loans are associated with borrowers and have appropriate relationships defined.

3. Repayment Tracking:

○ Implement a system that records repayments:

■ Payment ID

■ Loan ID

■ Payment Date

■ Amount Paid

■ Remaining Balance

■ Payment Term (e.g., 12 months)

○ Ensure that repayment schedules are correctly handled based on the loan terms and updates are made to the loan balance accordingly.

4. Interest Rate Management:

○ Create a module to manage interest rates (5%, 10%, 15%, or 20%) that can be applied to each loan.

○ Each loan should automatically calculate and apply interest based on the assigned rate.

5. Transaction Management:

○ Record every financial transaction on the loan (e.g., repayment, late fees, penalties):

■ Transaction ID

■ Loan ID

■ Transaction Date

■ Transaction Type (e.g., Repayment, Late Fee, Penalty)

■ Amount

■ Transaction Description (optional)

○ Display transaction history for each loan.

6. Contract Management:

○ Implement a file upload system to store loan contracts:

■ Contract ID

■ Loan ID

■ Contract Document (PDF or other formats)

■ Signing Date

○ Ensure contracts can be viewed and downloaded from the system.

Technical Requirements:

● MERN stack should be used to build the application.

● Design views using Tailwind css and ensure the user interface is simple yet functional.

● Utilize dependency injection for services and repository pattern for data access logic.

● Implement proper validation to ensure that the data entered into the system is correct and complete.

● Implement at least one unit test and one integration test.

Optional Features (For Bonus Points):

● Implement a loan calculator feature that calculates monthly repayments, including interest.

● Implement authentication and authorization for different user roles (e.g., Admin, Loan Officer).

Please make sure to follow the requirements and complete the project within one week. If you need any clarifications during the process, feel free to reach out to us. Once the project is completed, you can host it on GitHub and share the repository link with us.