Loan Approval System

Name: Abhideep Jain

UID: 24BCS10694

Section: 24KGR_711-B

Subject: Full Stack Development-I

Subject Code: 24CSP-210

Faculty: Er. Juned Alam

1. Introduction

In today's digital banking ecosystem, financial institutions are rapidly adopting automated systems to streamline loan processing, reduce manual errors, and improve decision-making efficiency. Traditional loan approval methods involve extensive paperwork, manual verification, and time-consuming approval cycles, which can lead to delays and inconsistencies. To address these challenges, the **Loan Approval System** project is designed as a modern, web-based application that automates the end-to-end loan approval workflow.

The Loan Approval System enables users to apply for loans online, checks their eligibility dynamically based on predefined criteria, and allows administrators to review and approve or reject loan applications efficiently. By integrating a robust backend with an intuitive frontend, the system ensures transparency, scalability, and accuracy in loan decision-making.

This project is developed using **Spring Boot** for backend services, **React** for frontend user interfaces, and **MySQL** for persistent data storage. The system emphasizes modularity, security, and real-time eligibility validation, making it suitable for real-world banking and financial applications.

2. Objectives of the Project

The primary objectives of the Loan Approval System are:

- To automate the loan application and approval process
- To reduce manual intervention and paperwork
- To provide a transparent and user-friendly loan application platform
- To implement a dynamic eligibility checker based on credit score and financial parameters
- To enable administrators to efficiently manage loan requests
- To store and retrieve loan-related data securely using a relational database
- To ensure scalability and maintainability through a modular architecture

3. Scope of the Project

The scope of this project includes the design, development, and testing of a web-based loan approval platform with distinct user roles. The system focuses on **personal loan processing** but can be extended to other loan types such as home loans, education loans, and vehicle loans.

4. Technology Stack

The Loan Approval System is built using the following technologies:

Frontend

- **React.js**
  - Component-based architecture
  - Responsive user interface
  - Efficient state management
  - REST API consumption

Backend

- **Spring Boot**
  - RESTful web services
  - Business logic implementation
  - Authentication and authorization
  - Dependency injection and modular structure

Database

- **MySQL**
  - Structured data storage
  - Relational schema design
  - ACID compliance
  - Efficient querying using SQL

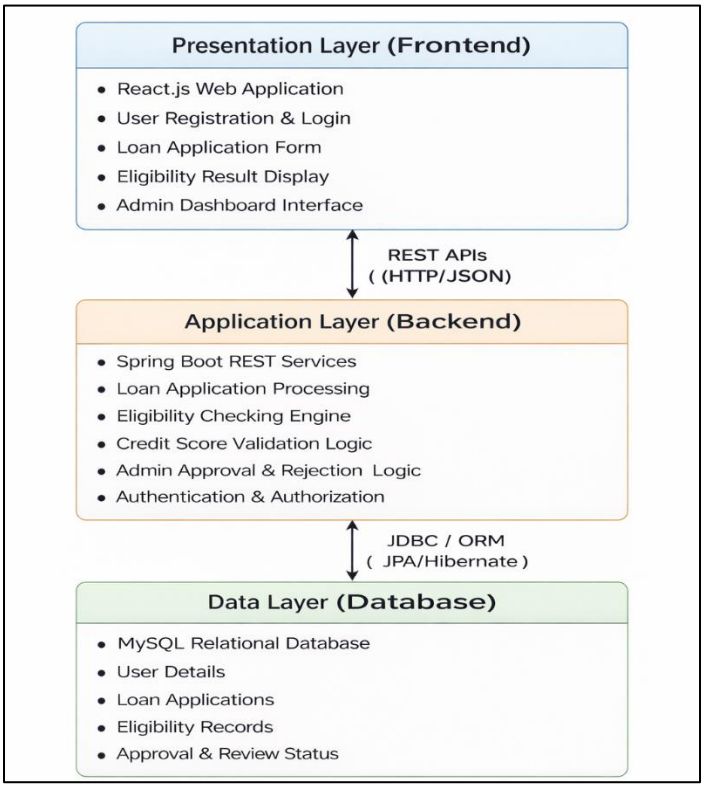
5. System Architecture

The system follows a **three-tier architecture**:

1. **Presentation Layer (Frontend)**
   - Developed using React
   - Handles user interaction and data visualization
   - Communicates with backend via REST APIs
2. **Application Layer (Backend)**

- Developed using Spring Boot
- Contains business logic and validation rules
- Manages authentication and authorization
- Interacts with the database layer

3. **Data Layer (Database)**
    - MySQL database
    - Stores user details, loan applications, eligibility results, and approval status



This architecture ensures separation of concerns, scalability, and ease of maintenance.

6. High-Level Design (HLD)

The system follows a **three-tier architecture**:

Presentation Layer

- Built using React
- Handles user interaction and data presentation
- Communicates with backend via REST APIs

Application Layer

- Built using Spring Boot
- Contains business logic, eligibility rules, and validation
- Manages authentication and authorization

Data Layer

- MySQL relational database
- Stores users, loan applications, eligibility results, and admin decisions

Data Flow Overview:

1. User submits loan application via frontend
2. Backend processes and evaluates eligibility
3. Admin reviews application
4. Decision stored in database
5. Status returned to user interface

7. Low-Level Design (LLD)

7.1 Backend Components

- **Controller Layer:** Handles HTTP requests and responses
- **Service Layer:** Contains business logic and eligibility calculations
- **Repository Layer:** Manages database operations using JPA/Hibernate
- **Security Layer:** Handles authentication, authorization, and token validation

7.2 Database Design

Entities:

- User
- LoanApplication

- EligibilityResult
- AdminReview

Each entity maps to a relational table with proper foreign key relationships to ensure data integrity.

8. Functional Modules

8.1 Loan Application Module

This module allows users to apply for loans by submitting personal, financial, and employment details.

Key Features:

- User registration and login
- Loan application form
- Input validation
- Loan amount and tenure selection
- Application submission and tracking

Inputs:

- Personal information (name, age, contact details)
- Employment details
- Monthly income
- Loan amount requested
- Loan tenure

Outputs:

- Application status (submitted, under review, approved, rejected)

8.2 Eligibility Checker Module

The Eligibility Checker is a core component of the system. It dynamically evaluates whether a user qualifies for a loan based on predefined business rules.

Eligibility Criteria (Sample):

- Minimum credit score threshold
- Minimum monthly income

- Debt-to-income ratio
- Employment stability

Key Features:

- Automated eligibility calculation
- Real-time feedback to users
- Configurable rules for future enhancements

Benefits:

- Faster decision-making
- Reduced human bias
- Consistent evaluation logic

8.3 Admin Approval Module

This module is accessible only to administrators and financial officers.

Key Features:

- Admin login and authentication
- View all loan applications
- Filter applications by status
- Review applicant details
- Approve or reject loan requests
- Add remarks or feedback

Admin Actions:

- Approve loan
- Reject loan
- Request additional information

9. Module Breakdown (8 Modules)

The project is divided into **8 independent modules** for better maintainability and scalability.

Frontend Modules (3 Modules)

Module 1: User Authentication Module (Frontend)

Description:
Handles user registration and login functionality.

Responsibilities:

- User signup and login forms
- Client-side validation
- Token storage and session handling

Module 2: Loan Application Module (Frontend)

Description:
Allows users to apply for loans through an interactive form.

Responsibilities:

- Loan application form
- Input validation
- Submission of application data to backend
- Display of eligibility feedback

Module 3: Dashboard & Status Tracking Module (Frontend)

Description:
Provides users and admins a dashboard to view application status.

Responsibilities:

- Display loan status
- View application history
- Admin application listing interface

Backend Modules (5 Modules)

Module 4: User Management Module (Backend)

Description:
Manages user accounts and authentication.

Responsibilities:

- User registration and login
- Password encryption
- Role assignment (User/Admin)

Module 5: Loan Application Management Module (Backend)

Description:
Handles creation and management of loan applications.

Responsibilities:

- Store loan application data
- Update application status
- Fetch applications for admin review

Module 6: Eligibility Evaluation Module (Backend)

Description:
Evaluates loan eligibility using dynamic business rules.

Responsibilities:

- Credit score validation
- Income-to-loan ratio calculation
- Employment duration validation
- Loan tenure feasibility check

Module 7: Admin Approval Module (Backend)

Description:
Allows administrators to review and decide on loan applications.

Responsibilities:

- View application details
- Approve or reject applications
- Add remarks and feedback

Module 8: Security & Authorization Module (Backend)

Description:
Ensures secure access and data protection.

Responsibilities:

- Role-based access control
- JWT/session token validation
- Secure API endpoint access
- Input validation and request filtering

10. Use Case Description

Primary Use Case: Loan Request and Approval

1. User registers and logs into the system
2. User submits a loan application
3. System evaluates eligibility automatically
4. Application is forwarded to admin
5. Admin reviews application details
6. Admin approves or rejects the loan
7. User is notified of the decision

This use case ensures a smooth, transparent, and efficient loan approval workflow.

11. Advantages of the System

This section highlights the key benefits provided by the Loan Approval System.

- Faster loan processing significantly reduces approval time and improves user satisfaction.
- Reduced paperwork minimizes manual effort and streamlines operational workflows.
- Improved accuracy ensures fair, consistent, and unbiased eligibility evaluation.
- User-friendly interface simplifies application submission and loan status tracking.
- Scalable architecture supports future expansion and increased user load.
- Centralized data management enables efficient storage, retrieval, and monitoring of records.

These advantages make the system suitable for modern, technology-driven financial institutions.

12. Limitations

This section outlines the current constraints affecting system functionality.

- No real-time credit bureau integration limits external credit verification capabilities.
- System depends heavily on user-provided data, which may reduce evaluation accuracy.
- Manual admin approval is still required for final loan decision decisions.
- Limited to predefined eligibility rules, reducing adaptability to complex scenarios.

These limitations identify areas requiring improvement in future system versions.

13. Future Enhancements

This section describes planned upgrades to enhance intelligence, automation, and usability.

- Integration with credit bureaus like CIBIL and Experian for real-time credit assessment.
- AI and machine learning based risk assessment for smarter loan approval decisions.
- Mobile application support to improve accessibility and user engagement.
- Email and SMS notification system for real-time loan status updates.
- Multi-loan type support including education, home, and vehicle loans.
- Admin dashboard analytics to monitor application trends and system performance.

These enhancements will significantly increase system efficiency, intelligence, and real-world applicability.

14. Conclusion

The Loan Approval System is a robust, scalable, and efficient web-based solution that modernizes the loan approval workflow. By leveraging Spring Boot, React, and MySQL, the system ensures reliability, security, and performance. The modular design allows easy expansion and integration with advanced technologies such as artificial intelligence and real-time financial verification systems.

This project demonstrates practical implementation of full-stack development concepts and serves as a strong foundation for real-world financial software applications.