SRS

OF

FinTech Bridge: Empowering Banking Through Digital Innovation

Group No: 04

Submitted By: Piyush Raj (12405997)

Ashutosh Kumar (12403211)

Ashwini Kumar (12403215)

Nitish Kumar Pathak (12409335)

Submitted To: Dr. Ajay Rastogi

Table of Content

S No:	Topic	Page No
01.	Introduction	03
02.	Scope	03
03.	Definitions & Abbreviations	03
04.	Product Functions	03
05.	User Classes and Characteristics	04
06.	User Interfaces	04
07.	Software Interfaces	04
08.	Functional Requirements	04
09.	Non-Functional Requirements	05
10.	Performance Requirements	05
11.	Context Level Diagram	06
12.	DFD Level 0	07
13.	DFD Level 1	07
14.	USE Case Diagram	08
15.	ER Diagram	08
16.	Sample Test Cases	09
17.	Conclusion	09
18.	Limitations	09
19.	References	09

1. INTRODUCTION

The "FinTech Bridge" project is a digital banking system designed to modernize banking operations by providing a fast, secure, and easy-to-use desktop platform for users. This system supports account creation, transaction handling, password recovery, and admin management, ensuring a complete banking experience.

2. SCOPE

The system will allow:

- User registration, login, and account management
- Deposit, withdrawal, mini statement, and balance checks
- Password recovery for forgotten passwords
- Admin management of users and transactions
- Chatbot support for customer queries
- MySQL database integration for data storage

The software will serve banks, credit unions, and fintech companies aiming to digitalize their services.

3. DEFINITIONS & ABBREVIATIONS

Term	Meaning
UI	User Interface
DBMS	Database Management System
OTP	One time password
DAO	Data Access Object Layer
JDBC	Java Database Connectivity
JavaFX	Java GUI Framework

4. PRODUCT FUNCTIONS

- Registration and Authentication: In the FinTech Bridge system, new users can securely register by providing their personal details, while existing users can log in using their credentials. Authentication ensures that only valid users can access banking services, maintaining security and data privacy.
- Account Creation (Savings, Current): After successful registration, customers can create either a savings account for personal savings or a current account for business and frequent transactions. Each account is linked uniquely to the customer's profile and stored securely in the database.
- Deposit and Withdrawal of Money: Customers can deposit funds into or withdraw money from their accounts through a simple user interface. Every transaction updates the account balance in real-time and logs the operation into the transaction history for accountability.

- View Balance and Mini Statement: Customers can easily check their account balance and generate a mini statement to view recent transactions. The mini statement provides a quick summary of financial activities like deposits and withdrawals within the account.
- Password Recovery System: The system includes a password recovery feature that allows users to reset or recover their forgotten passwords after validating their identity. This ensures customers can regain account access without compromising security.
- Admin User Management: An admin panel is available for administrators to manage customer accounts, monitor all transactions, and oversee system operations. Admins can view, update, or deactivate user profiles, ensuring the system remains organized and secure.
- Chatbot Help Desk: A simple chatbot is integrated into the system to assist customers with common banking queries such as how to deposit money, create accounts, or recover passwords. This enhances the user experience by offering instant support without human intervention.
- Secure Database Transactions: All database interactions in the system are handled securely through JDBC with measures like prepared statements to prevent SQL injection. Sensitive data like passwords are securely managed, and proper session handling ensures data integrity and user confidentiality.

5. USER CLASSES AND CHARACTERISTICS

- Guest: A guest is an unregistered user who can sign up by providing their details. Once registered, they can log in to access basic functionalities of the system.
- Customer: A customer is a registered user who has access to full banking services. They can perform operations like viewing account details, transferring funds, and managing transaction.
- Admin: An admin is a user with special privileges who can manage customer accounts. They can also oversee and monitor all transactions to ensure smooth and secure operations.

6. USER INTERFACES

- Registration Panel: This panel allows new users to sign up by entering their personal and account-related details. It ensures that only registered users can access banking services.
- Login Panel: Existing customers use this panel to securely log in using their username and password. It verifies their identity before granting access to the dashboard.
- Forgot Password Panel: This panel helps users recover or reset their password if they forget it. It usually involves email or mobile verification for security.
- Dashboard Panel: The dashboard acts as the central hub where users can view their account summary, manage transactions, and access other features.

- Account Creation Panel: This panel allows users to create different types of bank accounts (like savings, current, etc.). It collects necessary information and processes new account requests.
- Deposit Panel: Users can deposit money into their accounts using this panel. It may offer multiple methods like cash deposit, cheque deposit, or online transfer.
- Withdrawal Panel: This panel enables users to withdraw funds from their account. It ensures that sufficient balance checks and authentication are performed.
- Statement Panel: Users can view a mini statement of recent transactions through this panel. It provides quick access to transaction history and balance details.
- Chatbot Dialog: A chatbot integrated here answers banking-related FAQs and helps users navigate through services without needing human assistance.
- Main Frame: The Main Frame is the core structure that contains and manages all other panels. It ensures a smooth and organized user interface flow.

7. SOFTWARE INTERFACES

- **Backend Database:** SQL SERVER MANAGEMENT STUDIO
- **Java Frontend:** Swing Components (Panels, Frames)
- **JDBC Connector:** mysql-connector-java.jar
- **Development Tools:** IntelliJ IDE

8. FUNCTIONAL REQUIREMENTS

ID	Functional Requirement
FR 1	Users must register before accessing banking services.
FR 2	Users must authenticate with credentials (username, password).
FR 3	Customers can create Savings or Current accounts.
FR 4	Users can deposit/withdraw funds from their account.
FR 5	Customers can generate mini statements.
FR 6	Admins can manage users and view all accounts/transactions.
FR 7	Chatbot will assist users with common banking queries.
FR 8	Users can recover forgotten passwords through secure methods.

9. NON-FUNCTIONAL REQUIREMENTS

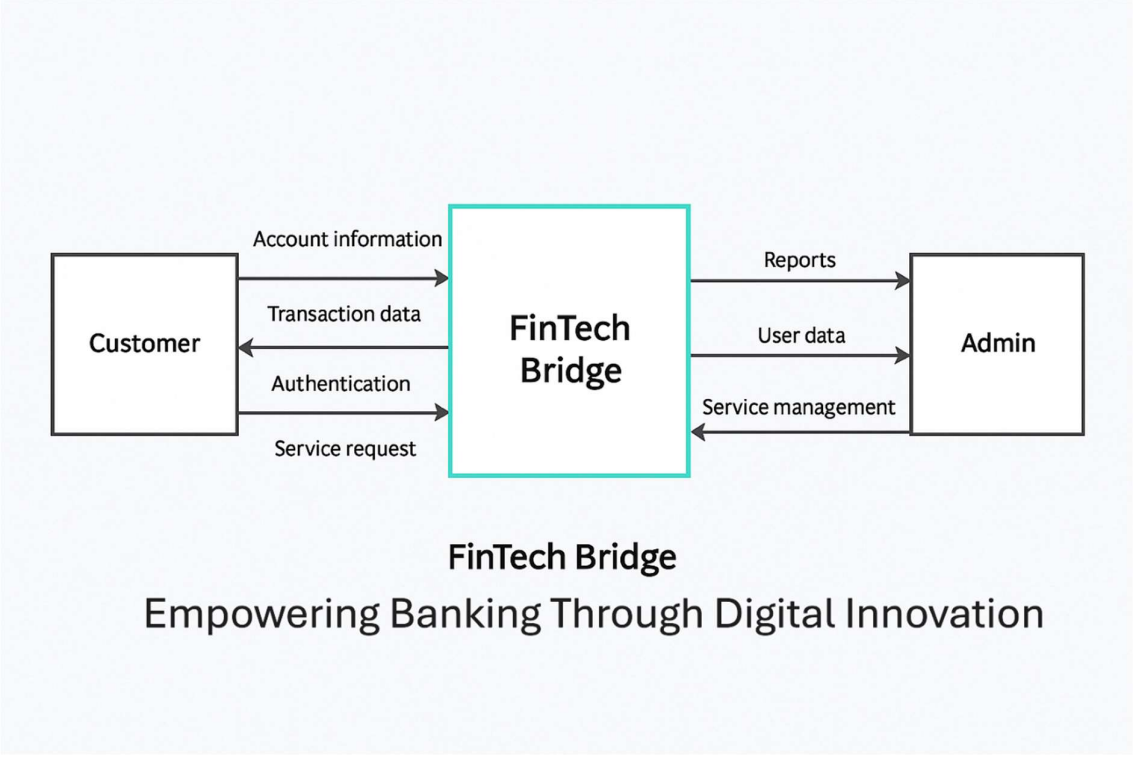
- Performance: Response time under 2 seconds.
- Scalability: Easily extendable to add new banking services.
- Security: All passwords must be encrypted (hashing suggested).
- Reliability: Should handle 1000+ concurrent users.

- Usability: Easy navigation using Swing-based UI.
- Maintainability: Clean modular code using MVC pattern.

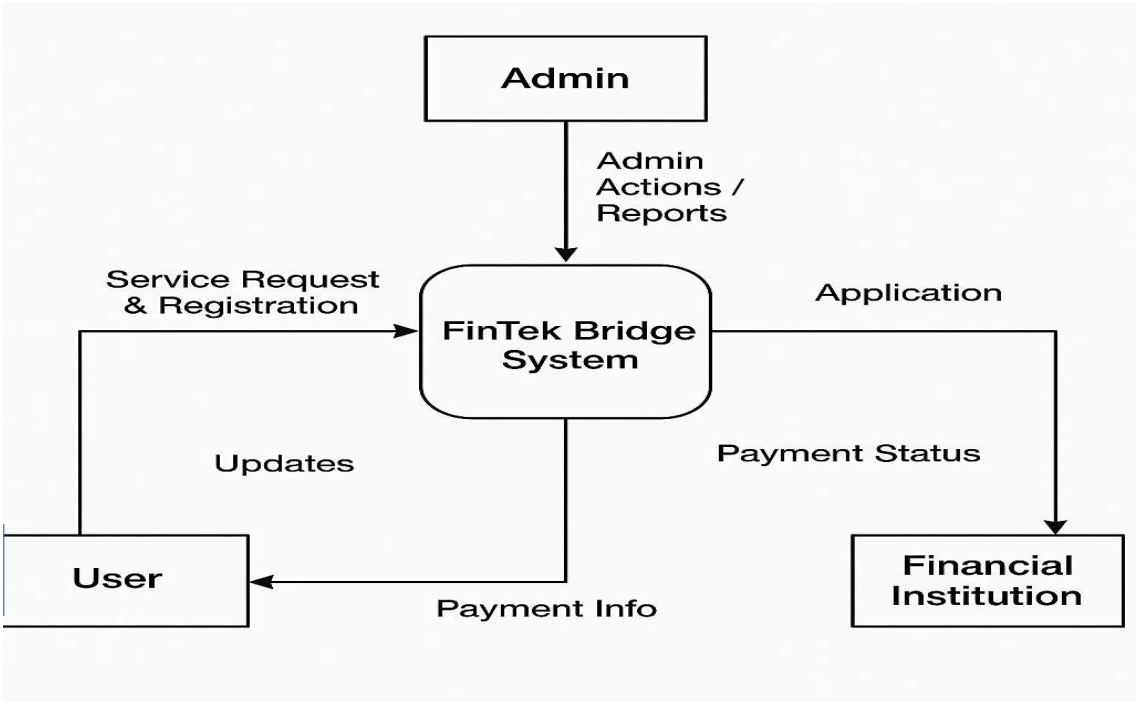
10. Performance Requirements

- Response Time: All user actions (e.g., login, transfer funds, apply loan) should get a response within 2–3 seconds under normal load.
- Concurrent Users: The system must support at least 50 simultaneous users performing transactions without crashing or freezing.
- Database Performance: SQL queries (especially for balance checking or transaction history) should execute in less than 1 second under normal load.
- Scalability: The system should allow easy future upgrades like adding new features (e.g., credit cards, bill payments) or support for 100+ users.
- System Availability: The system should have 99% uptime in deployed environments (demo or production servers) during working hours.
- Error Rate: System failures (e.g., transaction errors, app crashes) should be less than 2% during normal operations.

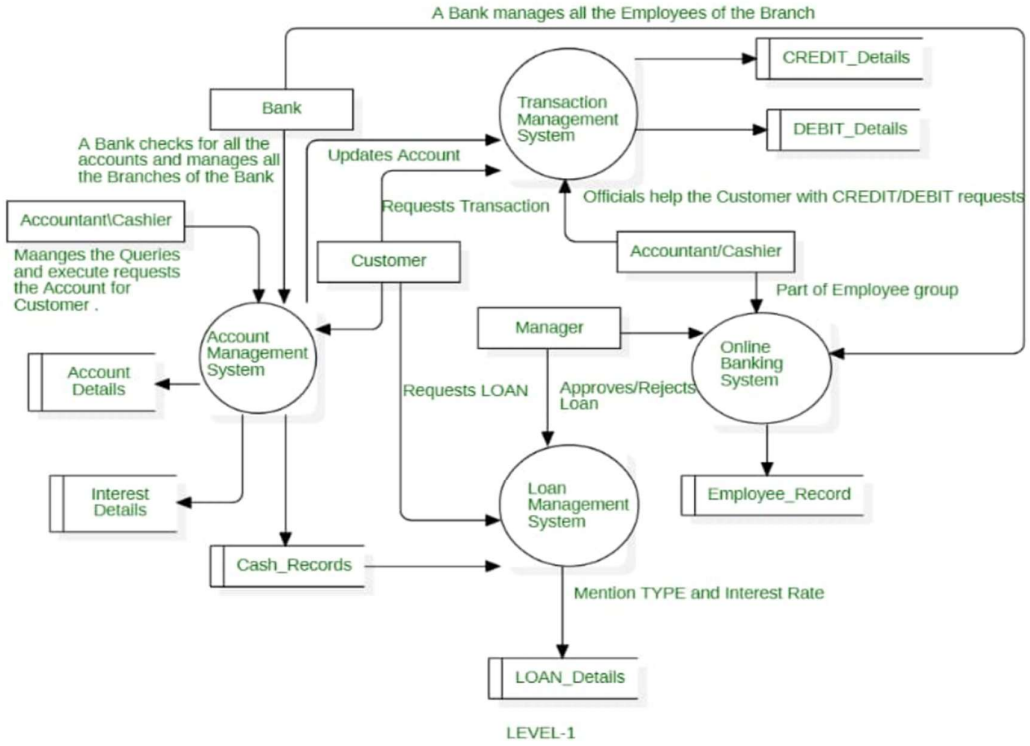
11.Context Level Diagram



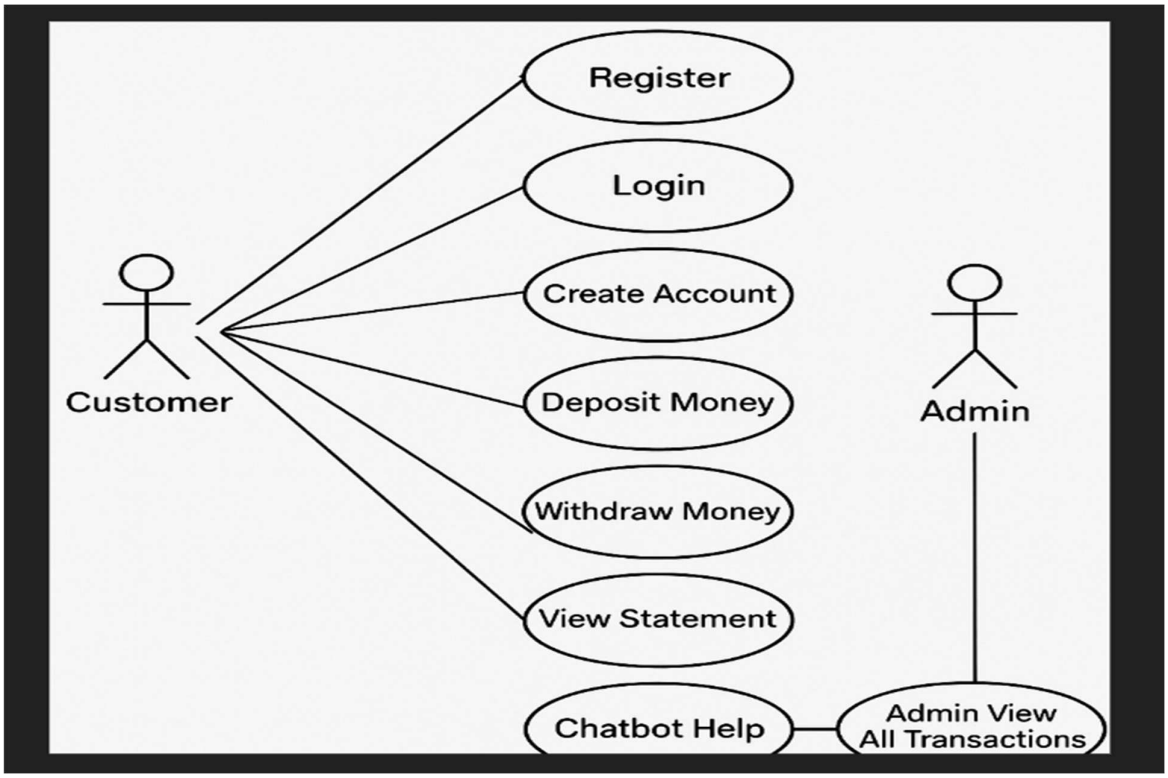
12. DFD LEVEL 0



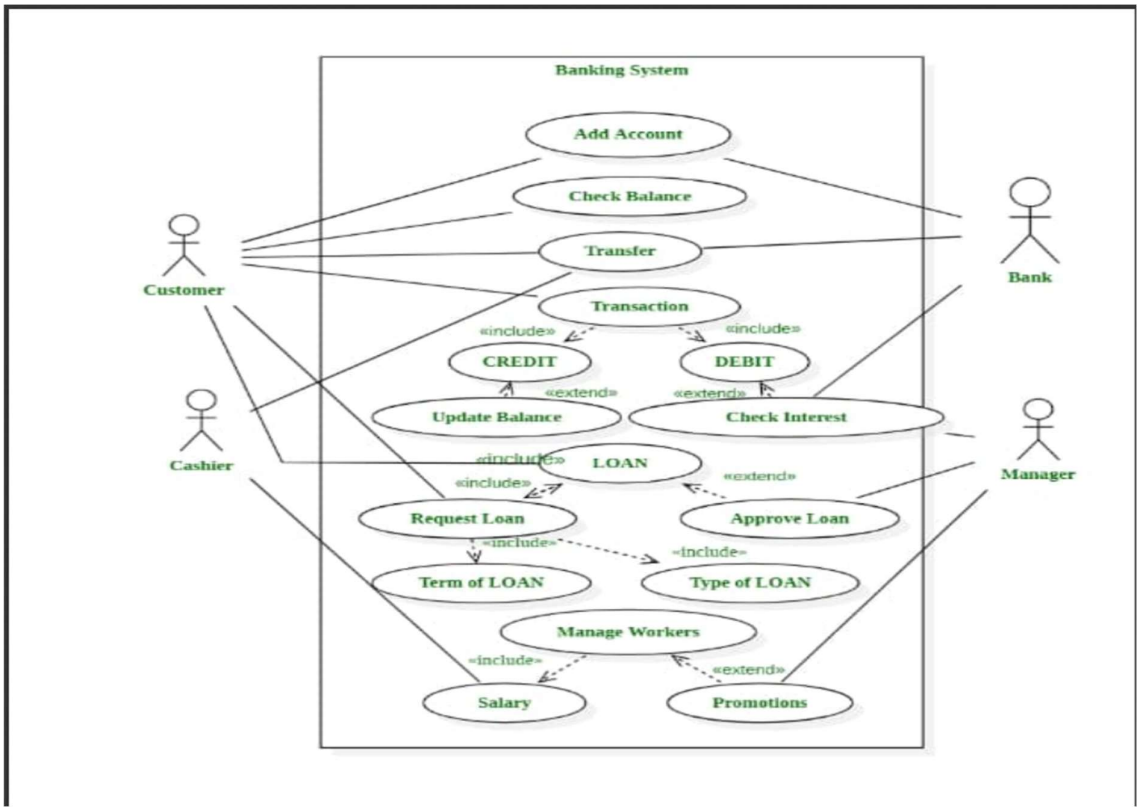
13. DFD Level 1



14. USE CASE DIAGRAM



15. ER DIAGRAM



16. SAMPLE TEST CASES

Test ID	Test Description	Expected Result
TC1	Valid login	Dashboard is shown
TC2	Invalid Login	Error message shown
TC3	Deposit Money	Amount added and receipt generated
TC4	Withdraw Money	Amount deducted if balance sufficient
TC5	Password Recovery	Email sent for reset
TC6	Mini Statement View	List last 5 transactions

17. CONCLUSION

The FinTech Bridge system will bridge the gap between traditional and digital banking services, providing a robust, secure, and easy-to-use platform for customers and administrators. With proper database management, secured transaction handling, and responsive UI, the system aims to offer a complete digital banking experience.

18. LIMITATIONS

- Application limited to desktop (no web/mobile version)
- Dependent on MySQL server availability
- Currently supports only one bank account per customer (can be enhanced later)

19. REFERENCES

- Java Official Documentation — Oracle Docs
- MySQL Documentation — MySQL Docs
- GeeksforGeeks.org
- StackOverflow.com
- YouTube (Java Swing Tutorials)