

Online Banking System

Akshat Tripathi

Arjun Jayakumar

Ritik Naik

Srijarko Roy

Case Study Summary

The Online Banking System application has been designed for two kinds of users - Customers and Admin. The application aims at simulating a real-time banking experience for the users by providing the most common banking services.

Admin:

- Search Users
- Enable or Disable users
- Open Bank Account for users
- Check User Transaction History

User:

- Balance Check
- Open Bank Account
- Check Transaction History
- Fund Transfer
- Withdrawal
- Transaction Password Change

Objectives

- Build a seamless and secured Online Banking Application
- Provide all basic services to users which are available in a banking system
- Provide admin controls
- Develop reusable and modular components
- Provide appropriate validations at Entity level as well as at UI
- Run proper tests at backend to ensure proper working of the application

Tech stack

Frontend: React, HTML, CSS

Backend: Java, Springboot

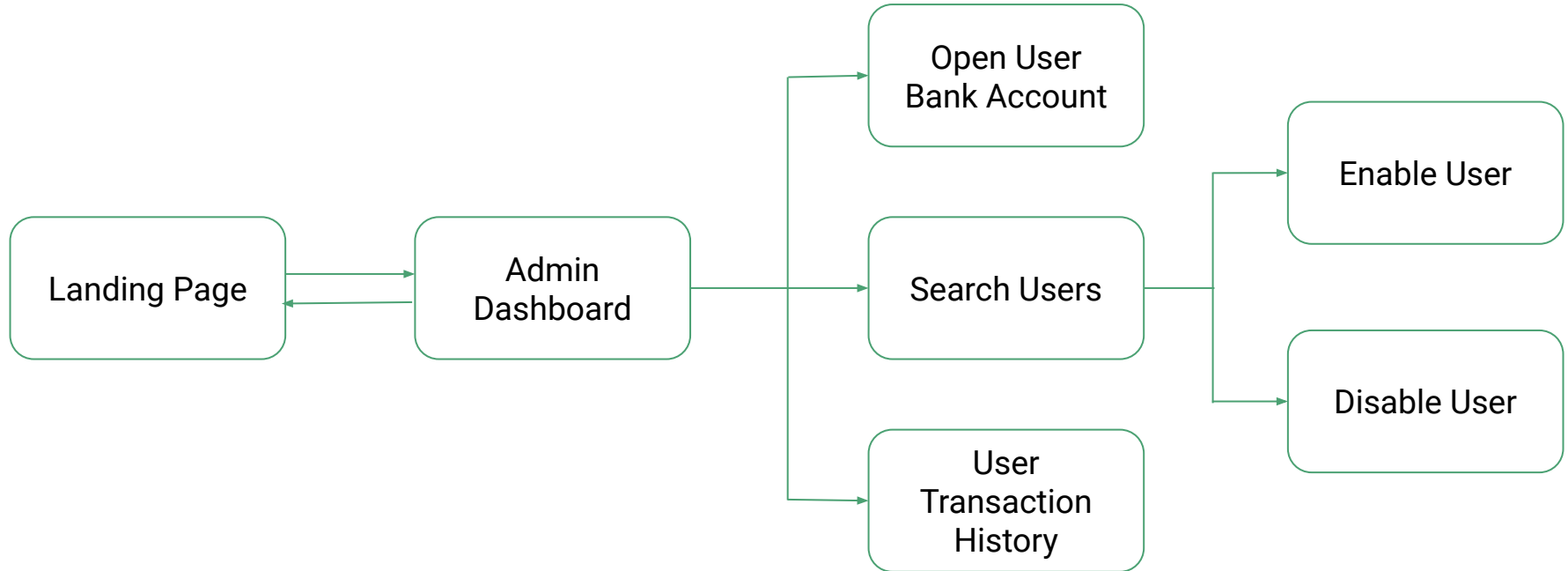
Database: MariaDB

Software Requirements: IntelliJ, Visual Studio Code, HeidiSQL, Postman, Git

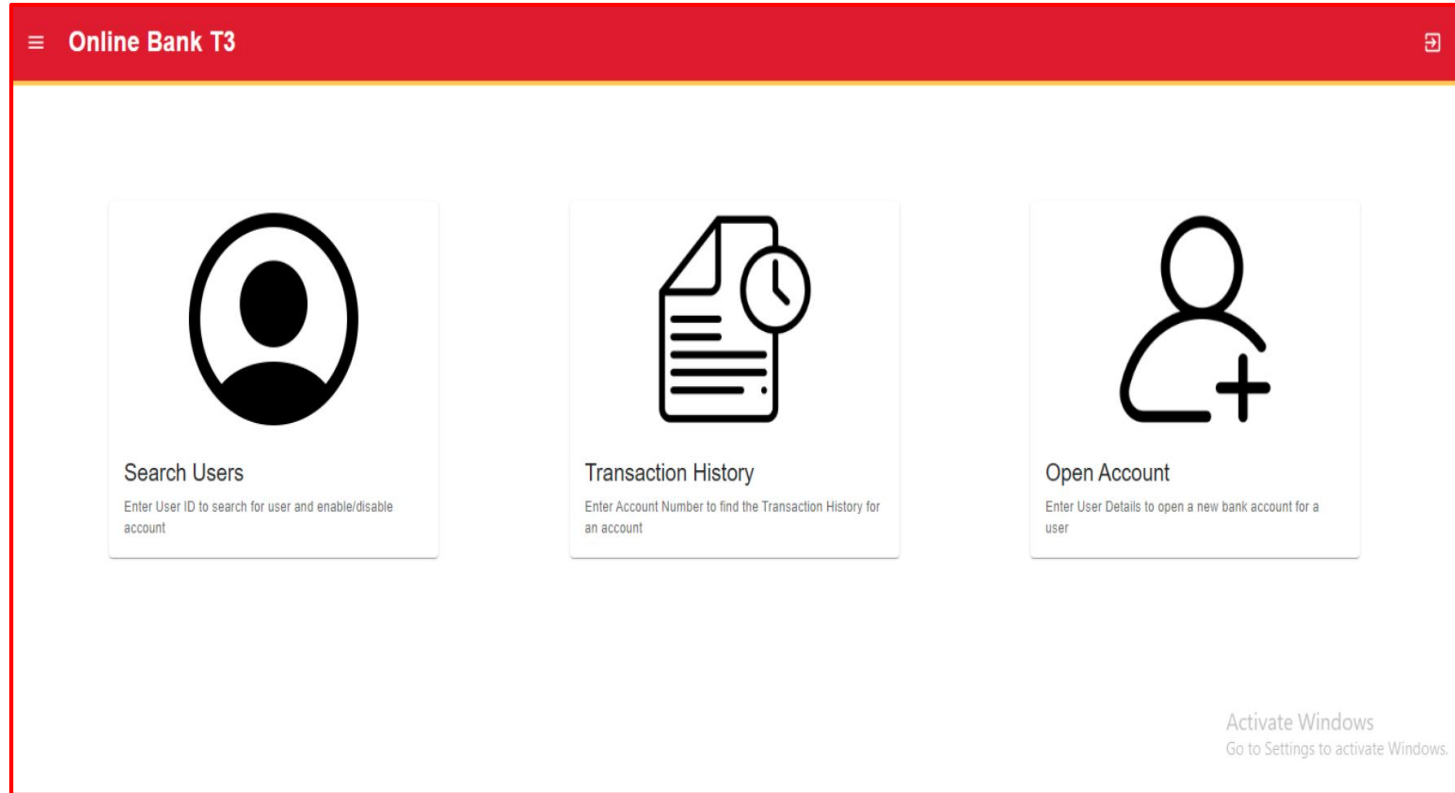
Testing: JUnit, Mockito

Build: Maven

Application Architecture - Admin



User Interface - Admin



User Interface - Admin

≡ Online Bank T3

Search Users

Username *

11006

SEARCH USER

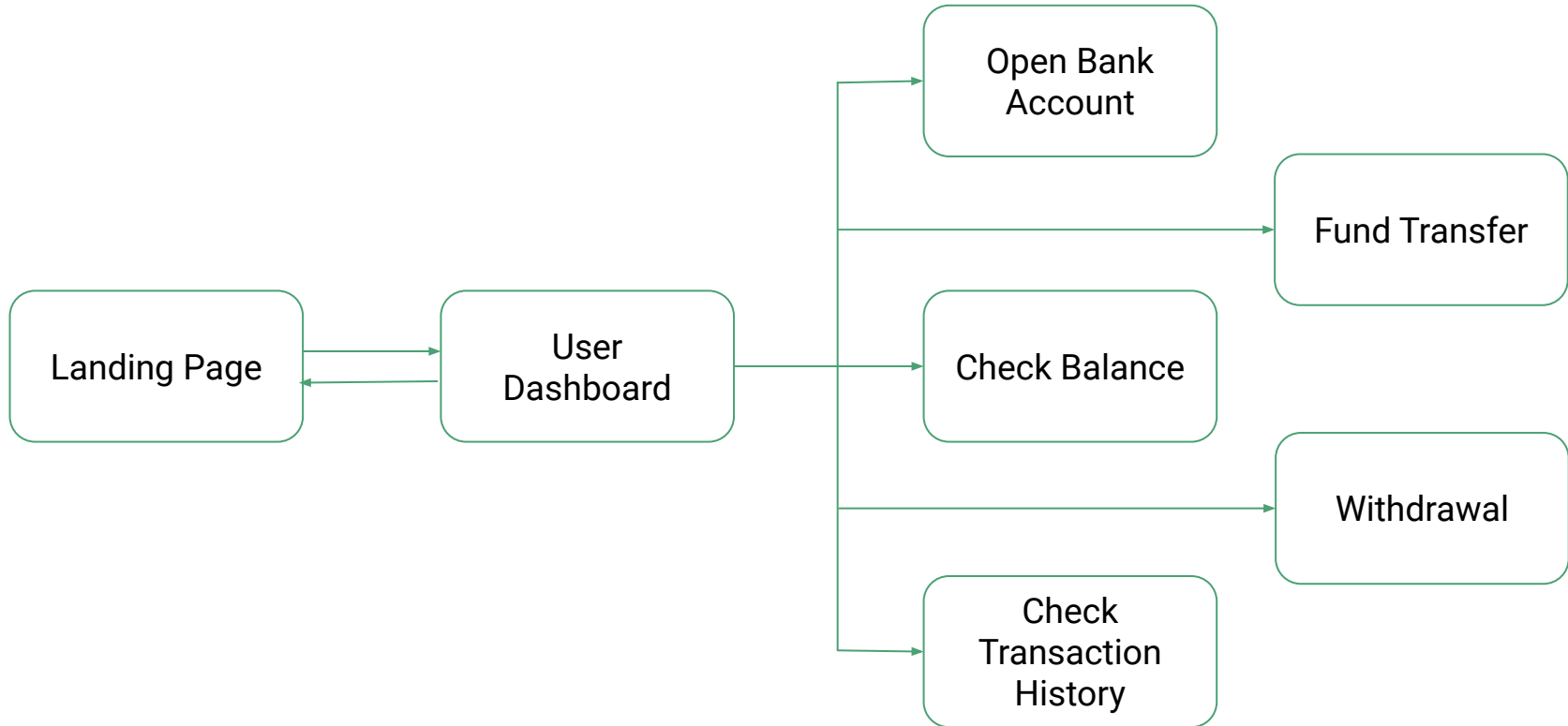
User ID	First Name	Last Name	E-Mail	Phone	Roles	Enable
11006	abcdw	abcdw	abcd@abcd.com	1234567890	USER	<input checked="" type="checkbox"/>

SAVE CHANGES

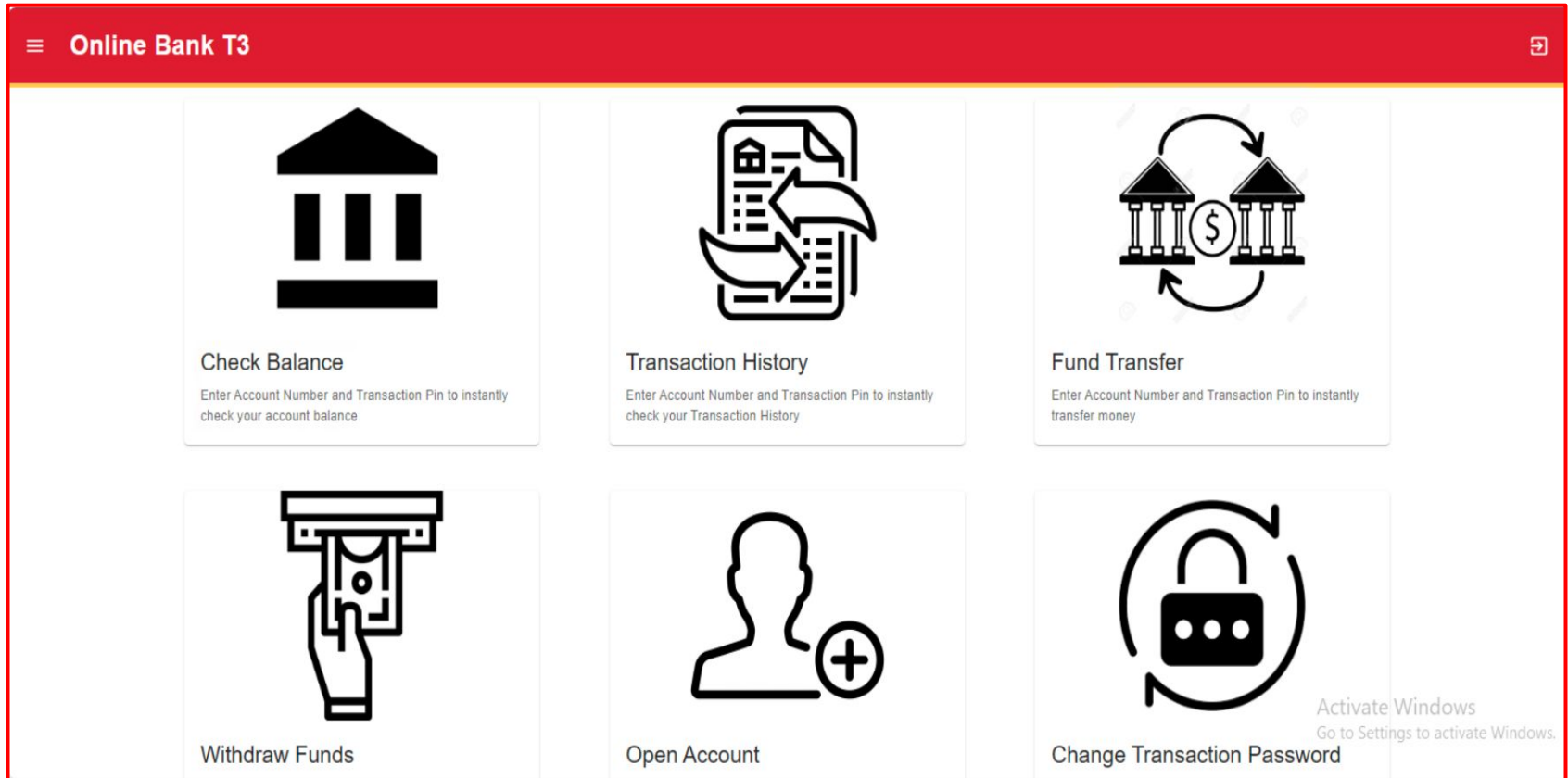
Activate Windows

Go to Settings to activate Windows.



Application Architecture - User



User Interface - User



User Interface - User

 **Online Bank T3** 

Funds Transfer

Account Number

Recipient Account Number *

Amount *

Statement *

Password *

CONFIRM TRANSACTION

Activate Windows
Go to Settings to activate Windows.

User Interface - User

≡ Online Bank T3

Transaction History

Account Number

1000002005

Password *

VIEW TRANSACTION HISTORY

Transaction ID	Sender Account Number	Recipient Account Number	Amount	Timestamp	Statement
3013	1000002005	1000002006	10	2023-10-02T13:36:15.000+00:00	try
3014	1000002005	1000002006	10	2023-10-02T13:36:39.000+00:00	try
3015	1000002005	1000002006	10	2023-10-02T13:38:53.000+00:00	try

Activate Windows
Go to Settings to activate Windows.

User Interface - User

≡ Online Bank T3

🔗

Account Balance Check

Account Number

1000002005

Transaction Password *

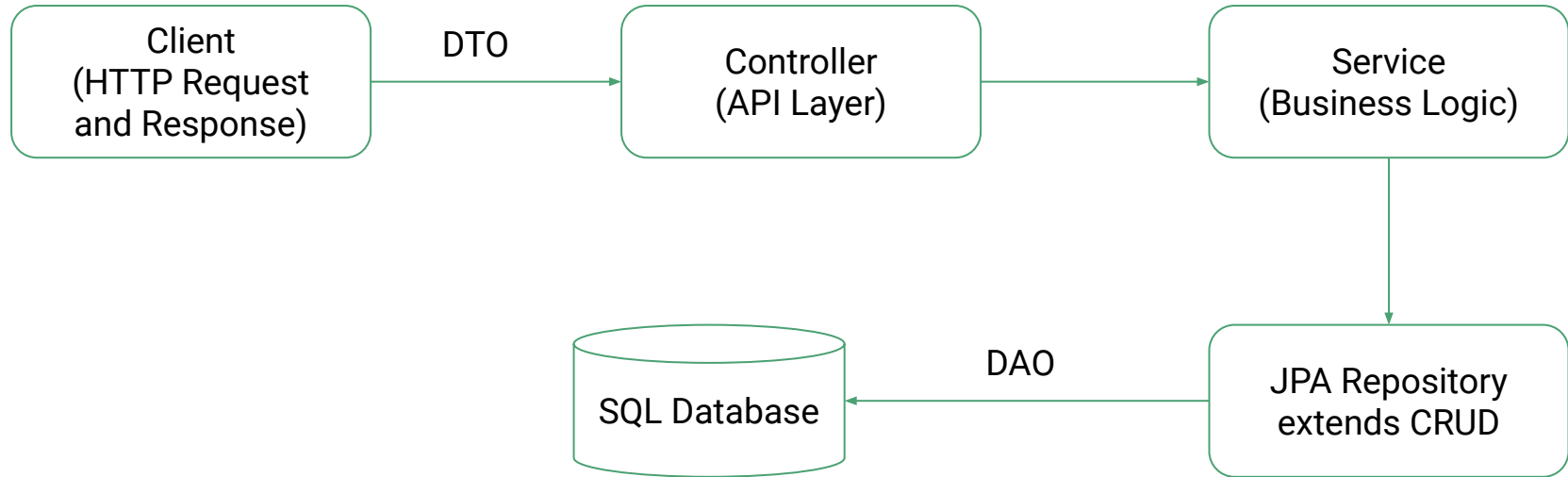
Your current Account Balance is ₹450

Close

Activate Windows

Go to Settings to activate Windows.

Application Architecture - Backend Flow



Application Architecture - Backend Flow (User)

Endpoint	Request Type	Service	Repository
/authenticate	POST	jwtService. generateToken()	
/accountDetails/getBalance	POST	userService. findBalance()	accountRepository. findByAccNo()
/allTransactionDetails	POST	userService. findAllTransaction()	transactionRepository. findAllBySenderAccNoOr RecipientAccNo()
/performTransaction	POST	userService. doTransaction()	accountRepository.save()

Application Architecture - Backend Flow (User)

Endpoint	Request Type	Service	Repository
/withdraw	POST	userService. withdrawAmount()	accountRepository. findByAccNo()
/accountDetails/createAccount	POST	userService. saveNewAccount()	accountRepository.save()

Application Architecture - Backend Flow (Admin)

Endpoint	Request Type	Service	Repository
/adminCheck	GET		
/userEnable	POST	adminService. setUserStatus()	userRepository. findUserById()
/allTransactionDetails/{accNo}	GET	adminService. findAllTransaction()	transactionRepository. findAllBySenderAccNoOr RecipientAccNo()

Challenges

- New technology stack for team
- JWT implementation using SpringBoot
- Technical Setbacks
- Securing Routes

Key Takeaways

- Experience with new Technologies
- Proper source code management
- Insights into real bank functionalities
- Teamwork
- Accountability

Future Scope

- UI enhancements
- Deployment
- Additional Functionalities (Cards, Insurances, etc.)