



Global Bank User Management Application

Team 4: Nishkal Karanth, Daksh Dave, Sankalp Mehadia, Shidharth S, Vivek Billa



Table of Contents

- Introduction
- Problem Statement
- Objectives
- Softwares Used
- Modules
 - Landing
 - Login
 - Registration
 - Home
 - Loan
 - Transaction
 - Statement



Introduction

Increased digitization in the 21st century and an increasing importance on bank decentralization and enhanced UI has led to banks adopting modern web development for their applications. Keeping in line with this trend, our team has developed a Client-side Management Web Application as part of our capstone project using a Java tech stack comprised mainly of React and Spring framework.



Problem Statement

The requirement is to build a web application with the specified layout and softwares used that allows users to login, perform transactions and apply for loans, using the softwares learnt over the course of the technical bootcamp.



Objectives

- As a user, they can register themselves to the application
- As a user, they can apply for a loan in a specific branch.
- As a user they can perform withdraw or deposit operations
- As a user they can view all their selected transactions, ie., withdraw or deposit for a specific period.
- Display all transactions for a specific period



Softwares Used

- Database - MongoDB
- View layer - React, Bootstrap, Redux
- Backend - Spring
- Database Connectivity - Spring JPA (Postman API testing)
- Version Control - Git
- Build Tools - Maven



Modules

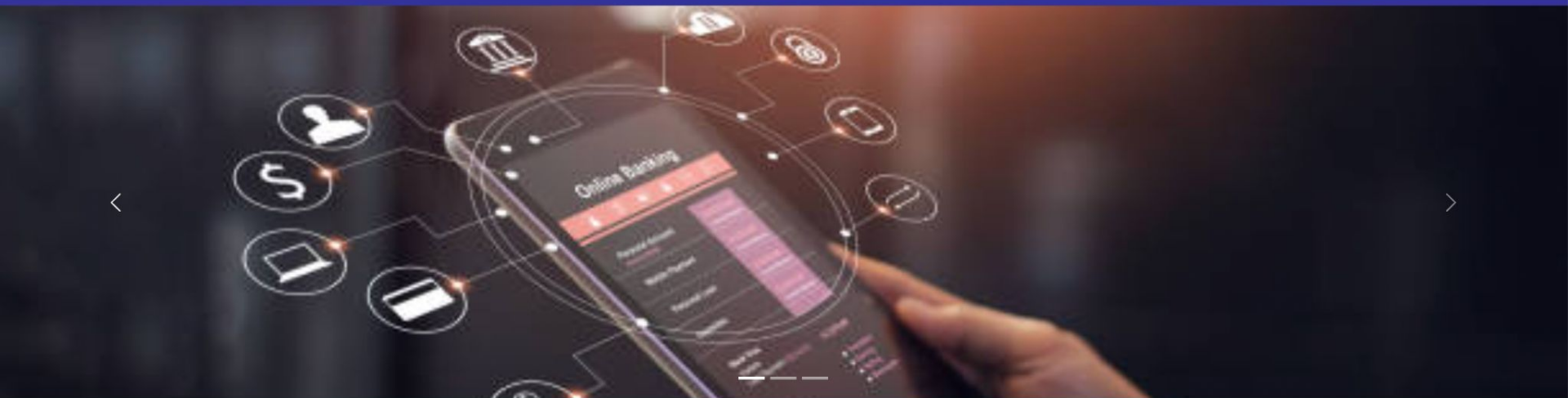
We have implemented the following components:

- Landing Page
- Login Page
- Registration Page
- Home Page
- Loan Page
- Transaction Page
- Statement Generation Page



Landing Page

- Every Customer hits the landing page first while visiting the website.
- Once the Customer logs in, they are redirected to this page, from which they can navigate.



Our Services

- Loans
- Banking
- Wealth Management

Apply for Loan

Easily apply for loans at our registered branches! Simply enter your amount and click apply!

[Apply](#)

Transact

With our easy menu, deposits and withdrawals are easier than ever!

[Transact](#)

Generate statement

Just enter the time period and leave the rest to us!

[Generate](#)



Login

- Through the login page a customer can access their account.
- This page also has a redirect to Registration page for new customers.
- If wrong or no credentials are entered then an appropriate error message pops up
- Customers should enter their customer_id and password to login.

LOGIN

Customer ID

Password

Login

First time? [Create an account.](#)

[Back to Homepage.](#)





Registration

- This page is for new customers to create an account.
- It contains the fields Customer ID, First Name, Last Name, Password, Phone Number, Email ID, City, Occupation and Date of Birth.
- Here the Customer ID is auto populated and the customer uses this for future logins.
- After the registration the customer is forwarded to the login page.
- All the fields are required without which a prompt is raised.
- This page also contains navigation back to the login page in case of existing customers.

REGISTER

Customer ID

Last name

Phone No.

City

Submit

Already a user? [Sign In.](#)

[Back to Homepage.](#)

First name

Password

Email ID

Occupation



Home

- This is the dashboard after a customer logs in.
- Here the customer has various options such as Loan, Statement, Transaction pages to redirect to.

Welcome ram!

Apply for Loan

Easily apply for loans at our registered branches! Simply enter your amount and click apply!

[Apply](#)

Generate statement

Just enter the time period and leave the rest to us!

[Generate](#)

Transact

With our easy menu, deposits and withdrawals are easier than ever!

[Transact](#)





Loan

- This page lets the customer apply for loans upto a limit of 1,00,000 INR.
- This page additionally contains a dropdown to select from a list of branches, as well as an auto populated Customer ID field.
- If either an invalid value is entered or a value greater than 1,00,000 INR is entered, a prompt appears to alert the customer.

Apply for Loan

Customer ID

1

Select Branch:

Hyderabad

Select amount

Apply



Transaction

- This page handles the transactions.
- Similar to the Loan page, it also has an auto populated field containing the CustomerID
- There is also a dropdown for the types of transactions, withdraw and deposit.
- There also exists a field that the Customer can populate with regards to the amount to be transacted.

This is Transaction

Customer ID

1

Select Tansaction Type:

Withdraw ▾

Select amount

Apply



Statement

- Similar to the Loan page, it also has an auto populated field containing the CustomerID
- This page generates a transaction statement based on the start and end date entered by the Customer.
- If a forward date is entered then a prompt is shown to the Customer.

This is Statement

Customer ID

1

Transaction Period From

dd/mm/yyyy



Transaction Period To

dd/mm/yyyy



Apply

List of Transaction:

Id : 7	Date: 2022-11-09T04:35:52.599+00:00	Type: Withdraw	Amount: 1000
Id : 8	Date: 2022-11-09T04:36:44.768+00:00	Type: Withdraw	Amount: 1000
Id : 9	Date: 2022-11-09T05:01:42.035+00:00	Type: Deposit	Amount: 10000
Id : 10	Date: 2022-11-09T05:01:47.117+00:00	Type: Withdraw	Amount: 10000
Id : 11	Date: 2022-11-09T05:05:01.591+00:00	Type: Deposit	Amount: 10000
Id : 12	Date: 2022-11-09T05:07:49.809+00:00	Type: Withdraw	Amount: 1000
Id : 13	Date: 2022-11-09T05:08:03.364+00:00	Type: Withdraw	Amount: 1000
Id : 14	Date: 2022-11-09T05:08:18.268+00:00	Type: Withdraw	Amount: 10000
Id : 15	Date: 2022-11-09T05:11:33.783+00:00	Type: Withdraw	Amount: 1000
Id : 16	Date: 2022-11-09T05:12:20.233+00:00	Type: Withdraw	Amount: 100000
Id : 17	Date: 2022-11-09T05:12:45.185+00:00	Type: Deposit	Amount: 105000
Id : 18	Date: 2022-11-09T05:15:32.316+00:00	Type: Deposit	Amount: 1000
Id : 19	Date: 2022-11-09T05:15:48.517+00:00	Type: Withdraw	Amount: 1000
Id : 20	Date: 2022-11-09T05:15:58.269+00:00	Type: Deposit	Amount: 1000

OpenAPI definition v0 OAS3

[/v3/api-docs](#)

Servers

customer-controller

**POST** /api/validate**POST** /api/transaction/{id}**POST** /api/topdf/{id}**POST** /api/loan/{id}**POST** /api/customer**GET** /api/lastid**GET** /api/customers**GET** /api/customer/id/{id}

Cover: GlobalBankUserManagementSystemApplication x CustomerServiceTest x

✓ Tests passed: 10 of 10 tests – 367 ms

✓ CustomerServiceTest (com.exe 367 ms)

- ✓ registerCustomers() 350 ms
- ✓ loginCustomerWithValidCustor 2 ms
- ✓ loginCustomerWithValidCustor 1 ms
- ✓ applyForLoan() 1 ms
- ✓ withdrawOrDeposit() 1 ms
- ✓ getByld() 1 ms
- ✓ getAllCustomer() 1 ms
- ✓ dateFilter() 8 ms
- ✓ getLastId() 1 ms
- ✓ validateWithNullCustomer() 1 ms

/Users/sankalp/Library/Java/JavaVirtualMachines/openjdk-18.0.2/Contents/Home/bin/java ...

---- IntelliJ IDEA coverage runner ----

sampling ...

include patterns:

com\example\demo\Service\..*

exclude patterns:

Thu Jan 01 05:30:00 IST 1970 Thu Jan 01 05:30:00 IST 1970

Class transformation time: 0.0171158s for 2357 classes or 7.2616885871871025E-6s per class

Process finished with exit code 0

Coverage