Project Name : ICIN Bank

Team6 Group Project

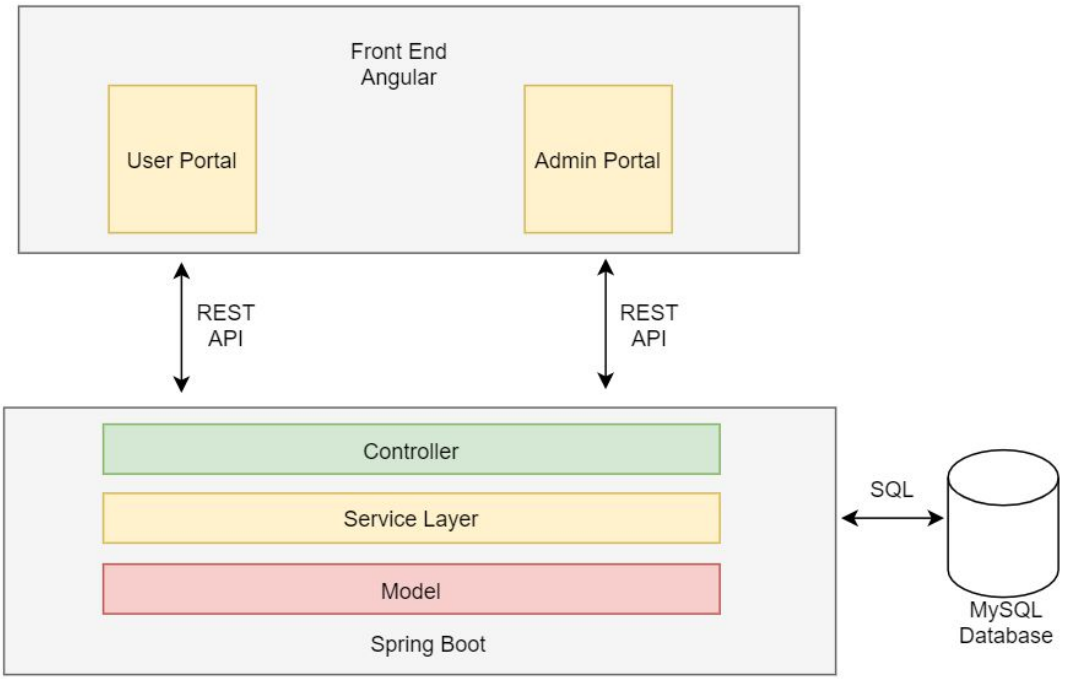
Github link: https://github.com/meetshah133/ICIN

Problem Statement: Create a dynamic and responsive Java online banking web application to deposit, withdraw, and transfer the money between the accounts.

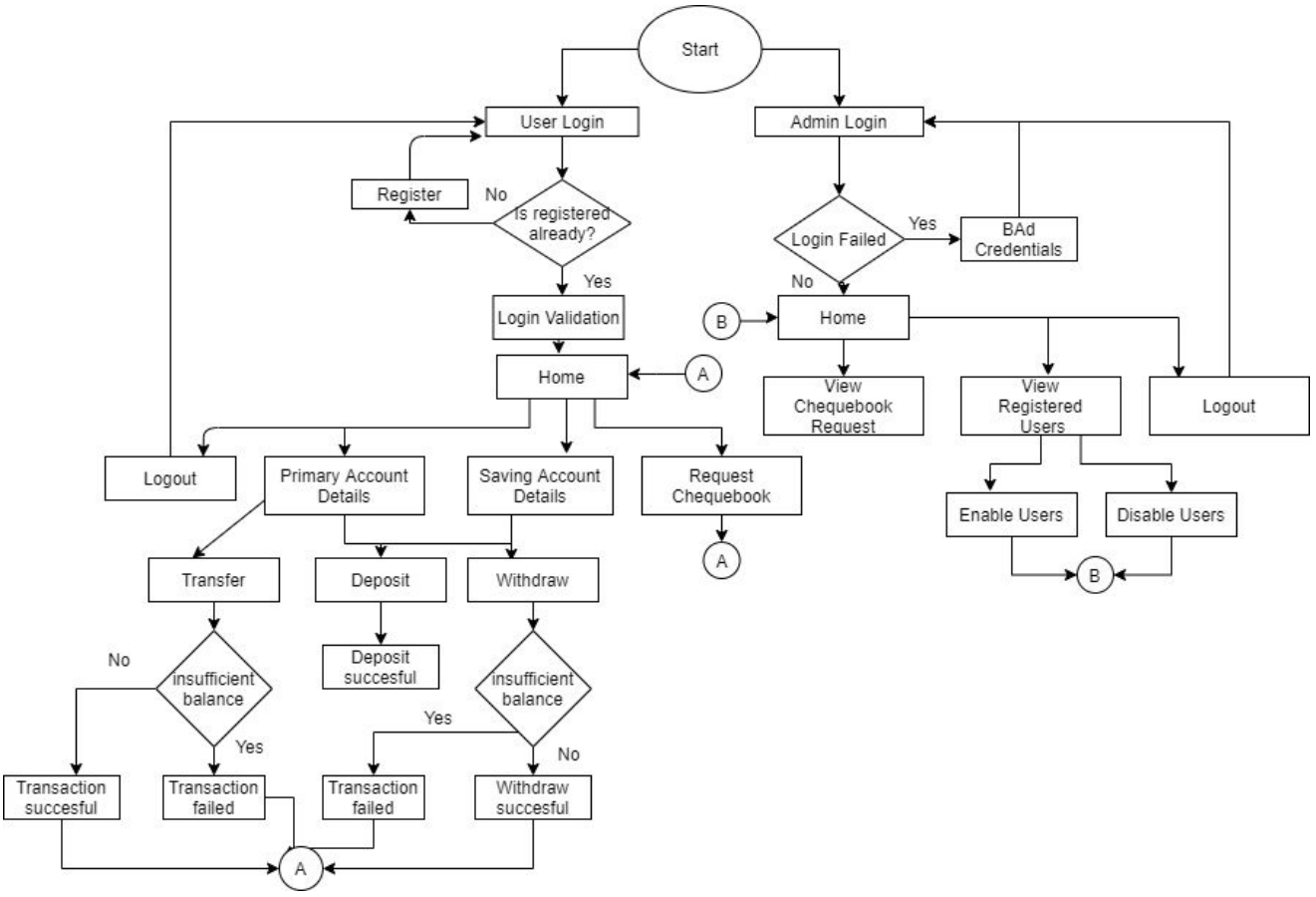
Technologies used:

1. Database management: MySQL
2. Back-end logic: Java programming, SpringBoot framework
3. Front-end development: Angular 8
4. Automation and testing technologies: Selenium and JUnit

System Architecture



Flowchart:


Start
User Login
Admin Login
Register
No
Is registered already?
Yes
Login Validation
Home
A
Logout
Primary Account Details
Saving Account Details
Request Chequebook
Transfer
Deposit
Withdraw
No
insufficient balance
Yes
Transaction succesful
Transaction failed
Deposit succesful
Yes
insufficient balance
No
Transaction failed
Withdraw succesful
A
Login Failed
Yes
BAd Credentials
No
B
Home
View Chequebook Request
View Registered Users
Logout
Enable Users
Disable Users
B
A

Features:

Admin

The admin has few higher privileges compared to a normal user and provides the following services.It has a service of disabling any user at any given point of time.It also has a service of accepting request made by the user for a cheque book.The admin is also given access to all user data which includes the user's personal details and also the primary and savings account balance details.The admin will also be able to view the transaction details of the user and the fund transfer he has done to other accounts.,

Methods:

- getAllUsers()

  This function provides a list of users that have registered to the application and the userdetails excluding the password.

- getPrimaryAccountById()

  This function allows admin to access the primary account details of the particular user by the user Id

- getSavingAccountById()

  This function allows admin to access the savings account details of the particular user by the user Id.

- getUserByEmailId()

  This function allows the admin to access a particular user's personal details using the email id that was provided during the time of registration.

- disableUser()

  This function disables the user from the user database and the user will no longer have access to the application.

- AcceptChequeBookRequest()

  This function accepts the cheque book request of a particular user.

- getAllChequeBookRequest()

  This function retrieves all the chequebook requests made.

User

Each user can access ICIN Bank portal by registering with their credentials

Registration form is divided as follow:

1.) Personal Details (Includes First and Last Name)
2.) Contact Details (Includes Email and Mobile Number)
3.) Kyc Details (Includes PanCard details and DOB)
4.) Address Details (Includes user's address)
5.) User Details (Includes username and password)

If the data entered is incorrect, the user is made aware of the required changes to be done. This involves:

1) PAN number should be alphanumeric
2) Phone number should be minimum of 10 characters.
3) Password should be of 8 or more characters
4) Age of the user has to be more than 10 years

Once the user is successfully registered he/she can login with his/her credentials

Functionalities provided to each user are

1.) View Account

    Each user has two accounts, primary and saving accounts. User can view the details such as account number and current balance of each account.

2.) Deposit Funds

    Users can deposit money in their saving or primary account via the ICIN Bank portal.

3.) Withdraw Funds

    Users can withdraw funds from their saving or primary account via the ICIN Bank portal.

    Preconditions to withdraw funds is that the user needs to have current balance greater than the requested balance for withdrawal.

4.) Transfer Funds

    Users can transfer funds to other beneficiary by using the transfer fund section.

    In order to transfer fund, user needs to enter the following:

1.) Beneficiary’s Name
2.) Beneficiary's Account Number
3.) IFSC Code
4.) Amount to be transferred

Preconditions to transfer funds are valid beneficiary’s account number and sufficient balance to perform the transaction. Exceptions are handled and appropriate error messages are displayed to the user.

5.) View Recent Transaction

Users can see their recent transactions through the portal.

Transaction details include transaction id, description, beneficiary account’s number and amount transferred.

6.) Request Cheque Book

User’s can request a cheque book for their primary account or saving account via by selecting the ‘Request’ option from the home page. User’s can enter an optional description while requesting cheque book. This request will be stored in a database and then routed to the admin. Admin can decide whether to accept or reject the request.

Account Transactions and Fund Transfer:

Security:

1.) Use of RouteGuardService

In order to prevent unauthorised users from accessing the application, we have used RouteGuardService which implements CanActivate interface. The precondition for canActivate is that the user should be authenticated before transferring funds, depositing money, withdrawing money and raising cheque book requests.

2.) Use of md5 hash

MD5 is a one way hashing function in which the string once hashed cannot be recovered from the hashed value. Hence to safeguard user credentials we are using MD5 hashing algorithm to calculate hash value of password and then this hash value is stored in the database.

Automation Testing

Selenium Webdriver is used along with TestNG framework for automation testing

Different test cases are :

1.) Registration test

In this test case all the registration fields are tested for their validity and the test result is passed if the user is successfully registered

2.) Login Test

In this test there are two scenario, first scenario user passses invalid credentials and he./she should get an Login Failed error

In second scenario user passed valid credentials and he/she will be redirected to homepage post login

3.) Deposit Fund Test

In this test, money is deposited in both primary as well as saving account. Test results are verified based on whether the transaction was successful or not.

4.) Withdraw Fund Test

In this test there are two scenarios, in the first scenario money greater than the current balance is requested for withdrawal in this cases test results are verified based on whether the ‘Insufficient Balance’ alert is displayed or not.

In the second scenario a valid amount is entered which is smaller than the current balance and the test results are verified based on whether the transaction was successful or not.

5.) Transfer Fund Test

In this test there are three scenarios, first scenario user tries to transfer money which is greater than his current balance, in this case test results are verified based on whether transaction failed and an “Insufficient Balance” alert is displayed to user or not.

In the second scenario the user tries to transfer money to an invalid beneficiary, in this cases test results are verified based on whether transaction failed or not.

In the third scenario the user tries to transfer money to a valid beneficiary and the transfer amount is less than the user's balance, in this case the test results are verified based on whether the transaction was successful or not.

6.) View Primary Account Test

In this test case primary account details of the user are displayed, test result is verified by asserting the expected and actual url.

7.) View Saving Account Test

In this test case saving account details of the user are displayed.

8.) Request Cheque Book Test

In this test case , a cheque book request is raised for the primary account, test result are verified based on whether cheque book request is submitted or not.

Backend Testing

Junit frame work is used for Back end testing in the application.

Test cases are developed for:

- User Registration
- Admin Login
- Deposit amount in primary account
- Deposit amount in savings account
- Withdraw amount from primary Account
- Withdraw amount from savings Account
- Create a Chequebook request
- Get all Cheque book requests
- Transfer amount within accounts
- Get all Accounts
- Get all Account Transactions

To run user portal

1. Open ICIN_frontend
2. Run: npm i
3. Then run : ng server -o

To run spring boot application

1. Open eclipse
2. Import new maven project
3. Select icin_bank_final
4. Select pom.xml and import

To run admin portal

1. Terminate processing running on port 4200
2. Open AdminPortal
3. Run npm i
4. Run ng serve -o

To run the test cases

Import maven project ICIN_Automation_Test

Open testing.xml file and run as TestNG Suite

To run junit test cases:

Run the TestWebApp.java class as junit test