

BANKING APPLICATION FULL STACK PROJECT

Table of Contents

AIM-	3
TECHNOLOGIES USED-	3
Back-End:	3
Front-End:	3
Automation Testing	3
API TESTING	3
THEORY ON TECHNOLGIES USED	4
Spring Boot	4
React JS	4
Mongo DB	4
Behavior Driven Development (BDD) Testing By Cucumber	4
Junit Testing	5
CODE FLOW	5
Spring Boot, React and MongoDB Architecture	5
PROJECT STRUCTURE	6
SPRING BOOT	6
MONGO DB	12
PROJECT IMPLEMENTATION	13
CHATBOT	13
USERS	15
SIGNUP PAGE	15
LOGIN PAGE	21
HOME PAGE	23
DEPOSIT	25
WITHDRAW	26
TRANSFER	27
UPDATE	27
BALANCE	29
DEPOSIT DETAILS	29
WITHDRAW DETAILS	30

TRANSFER DETAILS	30
SHOPPING PAGE	31
LOGOUT OR LOGIN AS ADMIN	33
ADMIN	33
LOGIN PAGE	34
HOME PAGE	35
ADD USER	37
DEPOSIT	37
WITHDRAW	38
TRANSFER	38
UPDATE	39
DEPOSIT DETAILS	40
WITHDRAW DETAILS	41
TRANSFER DETAILS	41
USER LIST AND DETAILS	42
AUTOMATION TESTING USING CUCUMBER	43
PROJECT STRUCTURE	43
TESTING IMPLEMENTATION	43
JUNIT TESTING FOR API	63
ADDING A USER OR USER SIGNUP (POST)	63
GETTING BALANCE FOR A USER (GET)	65
UPDATING USER DETAILS (PUT)	66
DELETING USER	68
SCOPE OF THE PROJECT	3
CONCLUSION	69

CODE LINK - <https://github.com/adityasingh190/BANK-APP-PROJECT.git>

AIM-

To build a full stack Bank APP using Spring Boot, React and MongoDB. The back-end server uses Spring Boot with Spring Web MVC for REST APIs and Spring Data MongoDB. The front-end side will be made with React JS.

SCOPE OF THE PROJECT

The scope of the Bank Management System extends to all the users who wish for easy banking facilities. This software product will be used for storing user's account information and the transactions made by them.

SCOPE:

- Creating New Accounts- The application can be used to create two different types of accounts by the customers, which are Savings Account and Current Account. It helps save the hustle for the customer to visit the bank physically and create/use these accounts.
- Depositing Money- As the world is moving towards the limited use of paper currency, depositing or transferring money from one bank to the other will become as easy as clicking a few buttons using this application.
- Withdrawing Money- Requests can be sent through the application to ask for money transfer as well.
- Account Holder List- This is a feature for the admin. The admin can view the list of all the account holders.
- Balance Enquiry- The customer can check their balance via this application.
- Changing Passwords/PIN- The customer can easily change the passwords and pin numbers using the application.
- Closing- The customer can close their accounts too using this application.

TECHNOLOGIES USED-

Back-End:

- Java 17
- Spring Boot 2.7.0
- Spring Boot Data Mongo
- Mongo DB.

Front-End:

- React 17.0.1
- Axios 0.27.2
- Bootstrap 4.5.0

Automation Testing

- CUCUMBER (BDD)

API TESTING

- JUNIT

THEORY ON TECHNOLGIES USED

Spring Boot

Spring Boot is an open source Java-based framework used to create a micro Service. It is developed by Pivotal Team and is used to build stand-alone and production ready spring applications. Spring Boot provides a good platform for Java developers to develop a stand-alone and production-grade spring application that you can just run.

React JS

React.js is built using JSX – A combination of JavaScript and XML. Elements are created using JSX, then use JavaScript to render them on your site. While React has a steep learning curve for a junior developer, it's quickly shaping into one of the most popular and in-demand JavaScript libraries.

React is considered a JavaScript library rather than a framework, whereas the other options we'll consider today are considered frameworks. It helps to think of a library as a tool that developers could use in any project and a framework as a whole design.

Mongo DB

- MongoDB stores data in flexible, JSON-like documents, meaning fields can vary from document to document and data structure can be changed over time
- The document model maps to the objects in your application code, making data easy to work with
- Ad hoc queries, indexing, and real time aggregation provide powerful ways to access and analyze your data
- MongoDB is a distributed database at its core, so high availability, horizontal scaling, and geographic distribution are built in and easy to use

Behavior Driven Development (BDD) Testing By Cucumber

In order to get better advantage of the software testing, organizations are nowadays taking a step forward. They implement important acceptance test scenarios while development is in-progress. This approach is commonly known as **Behavior Driven Development (BDD)**.

Behavior Driven Development gives us an opportunity to create test scripts from both the developer's and the customer's perspective as well. So in the beginning, developers, project managers, QAs, user acceptance testers and the product owner (stockholder), all get together and brainstorm about which test scenarios should be passed in order to call this software/application successful. This way they come up with a set of test scenarios. All these test scripts are in simple English language, so it serves the purpose of documentation also.

It is a very simple notion, but what we need in order to get this concept implemented. The answer is, Behavior Driven Development (BDD) Framework. Cucumber is one such open source tool, which supports behavior driven development. To be more precise, Cucumber can

be defined as a testing framework, driven by plain English text. It serves as documentation, automated tests, and a development aid – all in one.

Cucumber reads the code written in plain English text (Language Gherkin – to be introduced later in this tutorial) in the feature file (to be introduced later).

It finds the exact match of each step in the step definition (a code file - details provided later in the tutorial).

The piece of code to be executed can be different software frameworks like **Selenium**, **Ruby on Rails**, etc. Not every BDD framework tool supports every tool.

This has become the reason for Cucumber's popularity over other frameworks, like **JBehave**, **JDave**, **Easyb**, etc.

Cucumber supports over a dozen different software platforms like –

- Ruby on Rails
- Selenium
- PicoContainer
- Spring Framework
- Watir

Junit Testing

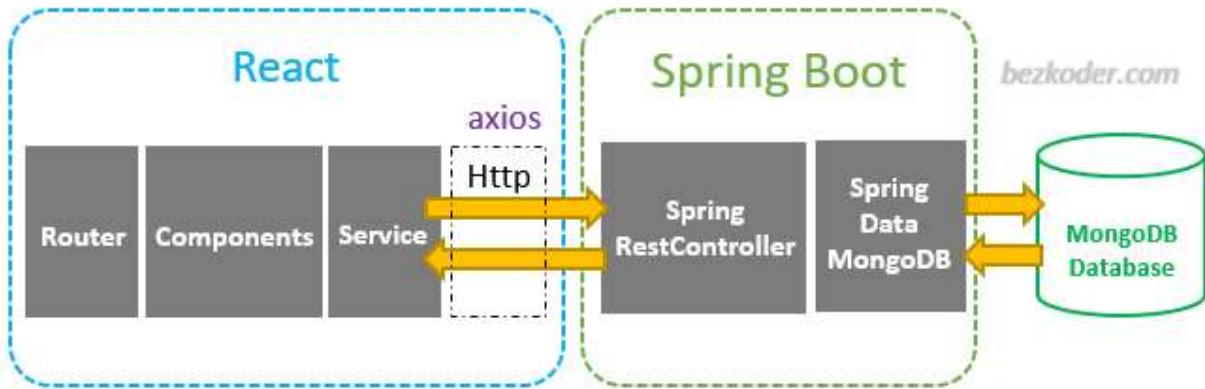
JUnit is a unit testing framework for Java programming language. It plays a crucial role test-driven development, and is a family of unit testing frameworks collectively known as xUnit.

JUnit promotes the idea of "first testing then coding", which emphasizes on setting up the test data for a piece of code that can be tested first and then implemented. This approach is like "test a little, code a little, test a little, code a little." It increases the productivity of the programmer and the stability of program code, which in turn reduces the stress on the programmer and the time spent on debugging.

CODE FLOW

Spring Boot, React and MongoDB Architecture

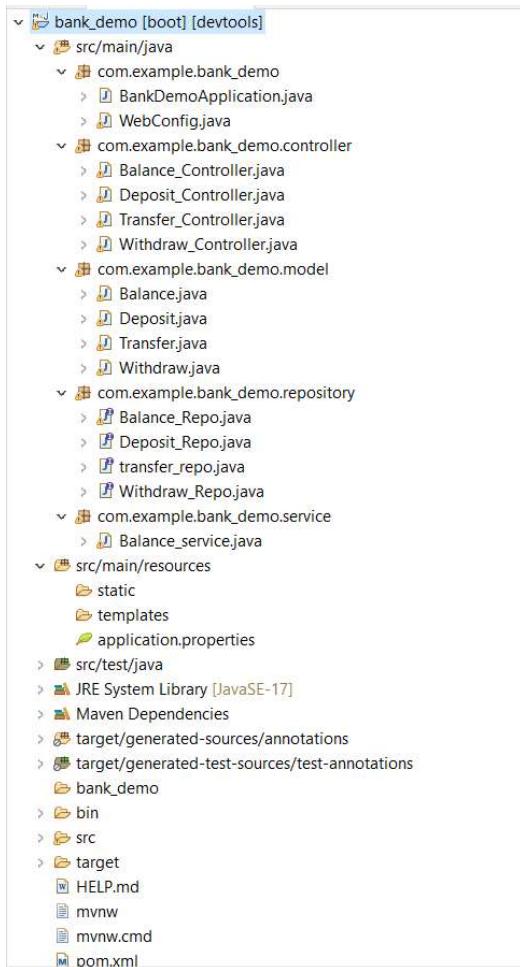
Project is built using the following architecture:



- Spring Boot exports REST APIs using Spring Web MVC & interacts with MongoDB Database using Spring Data MongoDB
- React Client sends HTTP Requests and retrieves HTTP Responses using **axios**, shows data on the components. We also use React Router for navigating to pages.

PROJECT STRUCTURE

SPRING BOOT



This project has 4 model and controller classes and 4 repository interfaces named Balance, Withdraw, Transfer and Deposit.

Configuration for Spring Data MongoDB is in **application.properties**.

```

1 spring.data.mongodb.host= localhost
2 spring.data.mongodb.port= 27017
3 spring.data.mongodb.database= Bank_Demo
4 server.port=8081

```

pom.xml contains dependencies for Spring Boot Web MVC and Spring Data MongoDB.

```

19    <dependencies>
20        <dependency>
21            <groupId>org.springframework.boot</groupId>
22            <artifactId>spring-boot-starter-data-mongodb-reactive</artifactId>
23        </dependency>
24        <dependency>
25            <groupId>org.springframework.boot</groupId>
26            <artifactId>spring-boot-starter-webflux</artifactId>
27        </dependency>
28        <dependency>
29            <groupId>io.projectreactor</groupId>
30            <artifactId>reactor-test</artifactId>
31            <scope>test</scope>
32        </dependency>
33        <dependency>
34            <groupId>org.springframework.boot</groupId>
35            <artifactId>spring-boot-devtools</artifactId>
36            <scope>runtime</scope>
37            <optional>true</optional>
38        </dependency>
39        <dependency>
40            <groupId>org.projectlombok</groupId>
41            <artifactId>lombok</artifactId>
42            <optional>true</optional>
43        </dependency>
44        <dependency>
45            <groupId>org.springframework.boot</groupId>
46            <artifactId>spring-boot-starter-test</artifactId>
47            <scope>test</scope>
48        </dependency>
49        <dependency>
50            <groupId>org.springframework.boot</groupId>
51            <artifactId>spring-boot-starter-web</artifactId>
52        </dependency>
53        <dependency>
54            <groupId>org.springframework.boot</groupId>
55            <artifactId>spring-boot-starter-data-mongodb</artifactId>
56        </dependency>
57    </dependencies>
58
59    <build>
60        <plugins>
61            <plugin>
62                <groupId>org.springframework.boot</groupId>
63                <artifactId>spring-boot-maven-plugin</artifactId>
64                <configuration>
65                    <excludes>
66                        <exclude>
67                            <groupId>org.projectlombok</groupId>
68                            <artifactId>lombok</artifactId>
69                        </exclude>
70                    </excludes>
71                </configuration>
72            </plugin>
73        </plugins>
74    </build>
75
76 </project>

```

REACT JS

✓ BANKINGAPPLICATION-MASTER

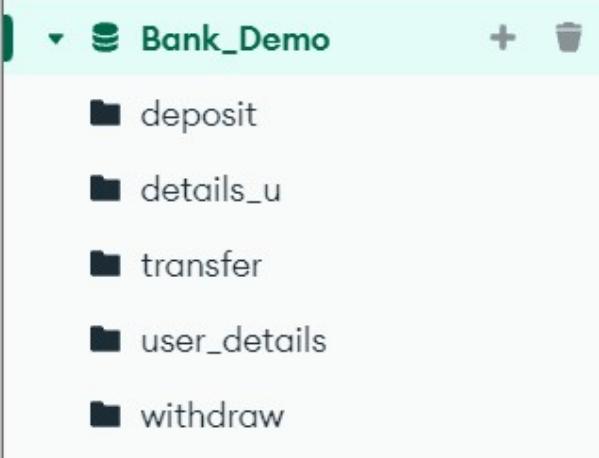
- ✓ bankingapp
 - > .idea
 - > .vscode
 - > node_modules
- ✓ public
 - > images
 - ★ favicon.ico
 - ↳ index.html
 - 🖼 logo192.png
 - 🖼 logo512.png
 - { manifest.json
 - ≡ robots.txt
- ✓ src
 - ✓ components
 - JS Accountslist.js
 - JS Addaccount.js
 - JS Adduser.js
 - JS Balance.js
 - JS Chatbot.js
 - JS Deposit.js
 - JS Deposit1.js
 - JS DepositDetails.js
 - JS DepositDetails1.js
 - 🖼 download.png
 - JS GetLoan.js
 - JS Home.js
 - 🖼 img.png
 - JS Loanslist.js
 - JS Login.js
 - JS NavigationBar.js

```
JS NavigationBar1.js
JS Register.js
JS Register1.js
JS Transfer.js
JS Transfer1.js
JS Transferdetails.js
JS TransferDetails1.js
JS Update.js
JS Update1.js
JS Userslist.js
JS Welcome.js
JS Welcome1.js
JS Withdraw.js
JS Withdraw1.js
JS WithdrawDetails.js
JS WithdrawDetails1.js
> images
  ✓ Shopping
    # cart.css
    JS Cart.js
    JS ContextCart.js
    JS Items.js
    JS products.js
    JS reducer.js
    JS Shopping.js
  ✓ style
    # customers.css
    # dashboard.css
    # edit.css
    # nav.css
    #
```

```
# nav.css
# new.css
# send_money.css
# transactions.css
# App.css
JS App.js
JS App.test.js
█ bankLogo.svg
# index.css
JS index.js
█ kam.jpg
█ logo.svg
JS serviceWorker.js
JS setupTests.js
⚙ .env
diamond .gitignore
{} package-lock.json
{} package.json
ⓘ README.md
```

Under components folder are different components that are rendered and being displayed on screen to interact with user or dealing with apis.

MONGO DB



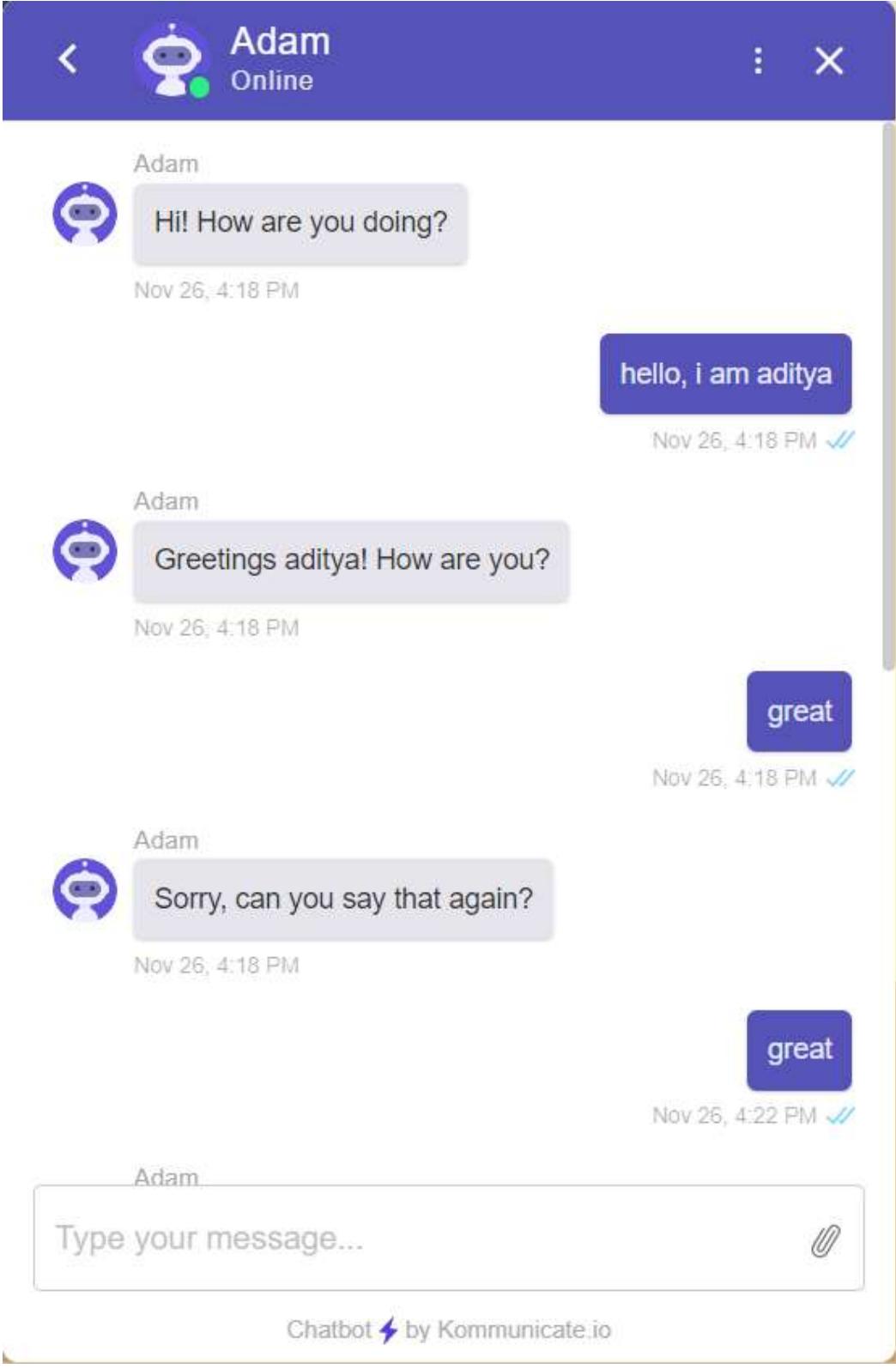
The screenshot shows the MongoDB Compass interface. At the top, there's a green header bar with the database name "Bank_Demo". Below the header, there are three icons: a minus sign, a plus sign, and a trash can. The main area contains a list of five collections, each represented by a small folder icon:

- deposit
- details_u
- transfer
- user_details
- withdraw

In mongoDB , there are 4 collections in which user details, deposit details , withdraw details and transfer details are stored.

PROJECT IMPLEMENTATION

CHATBOT



The screenshot shows a mobile messaging application interface. At the top, there's a purple header bar with a back arrow, the bot's name "Adam Online" next to a blue and white robot icon, and a close button (X). Below the header, the conversation log is displayed.

User (Aditya): Hi! How are you doing?
Nov 26, 4:18 PM

Bot (Adam): hello, i am aditya
Nov 26, 4:18 PM ✓

User (Aditya): Greetings aditya! How are you?
Nov 26, 4:18 PM

Bot (Adam): great
Nov 26, 4:18 PM ✓

User (Aditya): Sorry, can you say that again?
Nov 26, 4:18 PM

Bot (Adam): great
Nov 26, 4:22 PM ✓

User Input: Type your message... 

Chatbot  by Kommunicate.io

Adam
Online

great

Nov 26, 4:22 PM ✓✓

Adam



How can I be of assistance

Nov 26, 4:22 PM

What service this app provides?

Nov 26, 4:26 PM ✓✓

Adam



This Bank App provides money depositing, money withdrawing, money transferring, updating account details and viewing current balance.

Nov 26, 4:26 PM

i want to know each serivice

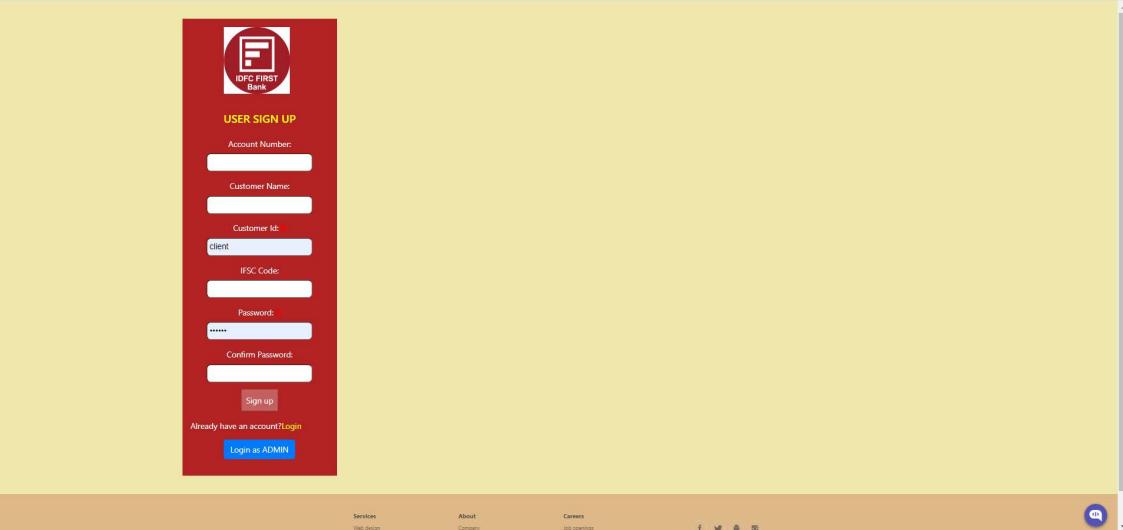
Nov 26, 4:26 PM ✓✓

USERS

SIGNUP PAGE

← → ⌂ localhost:3000

Apps Google Online Evaluation S... [edit] TCS NQT Course by...



USER SIGN UP

Account Number:

Customer Name:

Customer Id:

client

IFSC Code:

Password:

Confirm Password:

Sign up

Already have an account? [Login](#)

Login as ADMIN

Services

About

Careers

Facebook

Twitter

LinkedIn

YouTube



USER SIGN UP

Account Number:

Customer Name:

Customer Id: *

IFSC Code:

Password: *

Confirm Password:

Sign up

Already have an account? [Login](#)

[Login as ADMIN](#)

- All inputs are having some valid formats in which they are required to be filled with.
- If any or all the input fields are filled with invalid inputs , Sign up button is disabled and users can see the correct pattern in which the given fields can be filled with.



USER SIGN UP

Account Number: ✗

ⓘ Must be a Number
9 to 18 digits range should be there

Customer Name: ✓

Customer Id: ✗

ⓘ Must be a number and 6 digits

IFSC Code: *

sbin23

 Must start with 4 capital characters followed by '0' and then followed by 6 letters which can be characters as well as digits.

Password: *

.....

Confirm Password: *

.....

Sign up

Already have an account? [Login](#)[Login as ADMIN](#)

- If all the fields are filled with correct validation, then user can sign up and alert will be displayed saying “Account created” and Page will be navigated to Home Page

USER SIGN UP

Account Number: ✓

11100911273

Customer Name: ✓

abbas

Customer Id: ✓

124567

IFSC Code: ✓

SBIN0172458

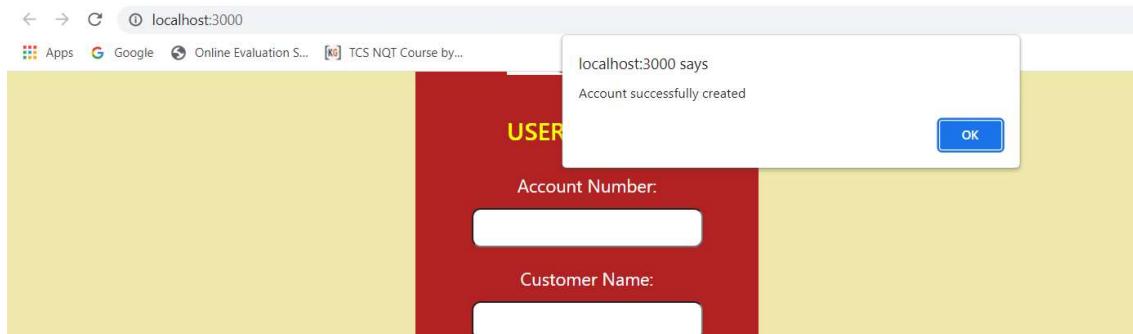
Password: ✓

.....

Confirm Password: ✓

.....|

Sign up



Mongo db



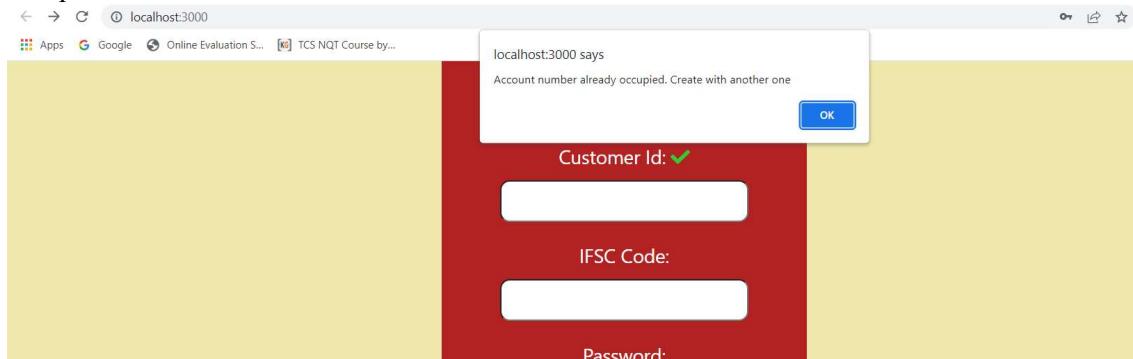
A screenshot of a MongoDB interface showing a document. The document contains the following fields and values:

```

_id: "11100911273"
custName: "abbas"
customerId: 124567
ifscCode: "SBIN0172458"
balance: 0
password: "Abbas@123"
_class: "com.example.bank_demo.model.Balance"

```

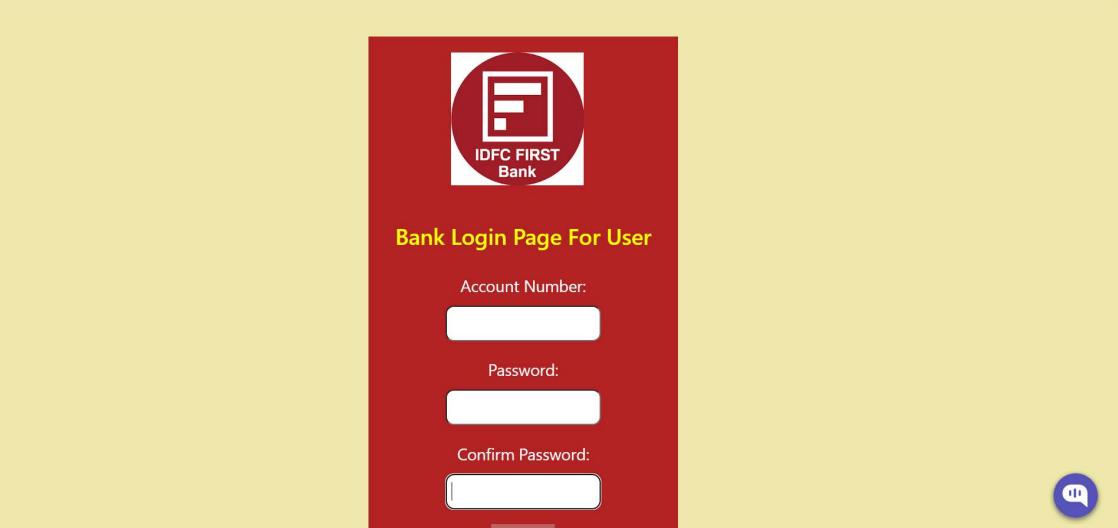
- If user already have account then alert will be displayed saying account number already occupied.



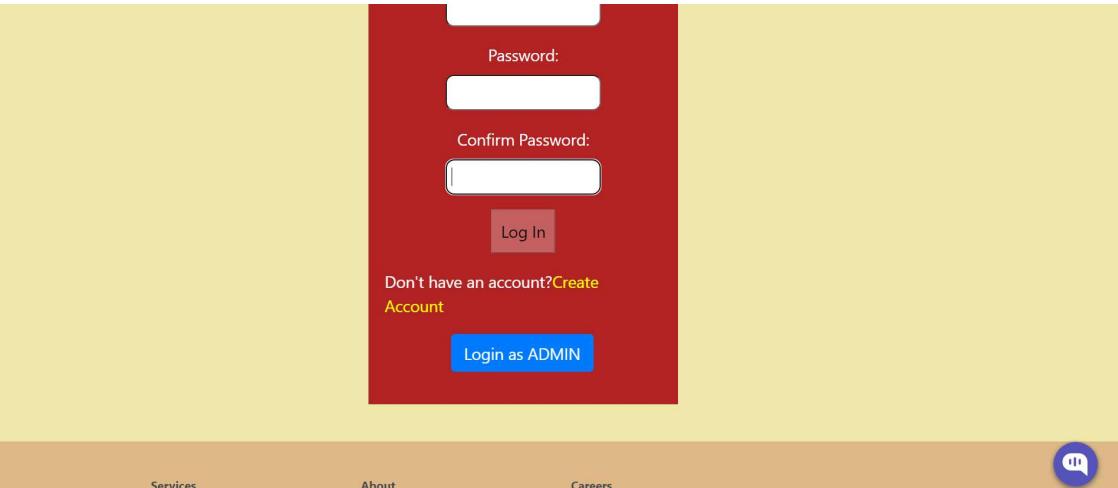
A screenshot of a web browser window titled 'localhost:3000'. The address bar shows 'localhost:3000'. The page content is a red form with fields for 'Customer Id:', 'IFSC Code:', and 'Password:'. A blue alert dialog box is overlaid on the page, displaying the message 'localhost:3000 says Account number already occupied. Create with another one' with an 'OK' button.

- So he/she is user or admin, then he/she can just click link to login or Login as Admin Link.

LOGIN PAGE



The screenshot shows the initial state of the login form. It features a red header with the bank's logo and the text "Bank Login Page For User". Below this, there are three input fields: "Account Number:", "Password:", and "Confirm Password:". Each field has a placeholder text inside it. To the right of the fields is a blue circular icon with a white speech bubble symbol.



The screenshot shows the login form after attempting to submit with empty fields. The "Log In" button at the bottom of the form is now grayed out and disabled. The other input fields remain visible with their respective placeholder text. The blue circular icon with the speech bubble is still present to the right of the fields.

- All inputs are having some valid formats in which they are required to be filled with.
- If any or all the input fields are filled with invalid inputs , Log in button is disabled and users can see the correct pattern in which the given fields can be filled with.

Bank Login Page For User

Account Number: *

Password: *

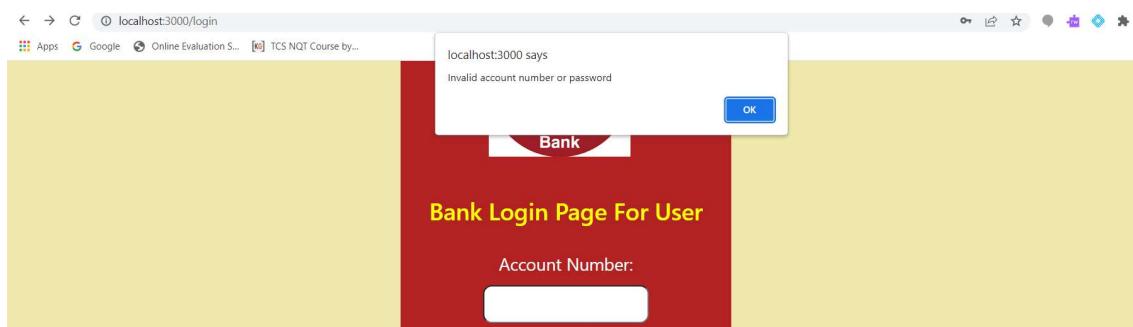
Confirm Password: *

! Must match the first password input field.

Log In

Don't have an account? [Create](#)

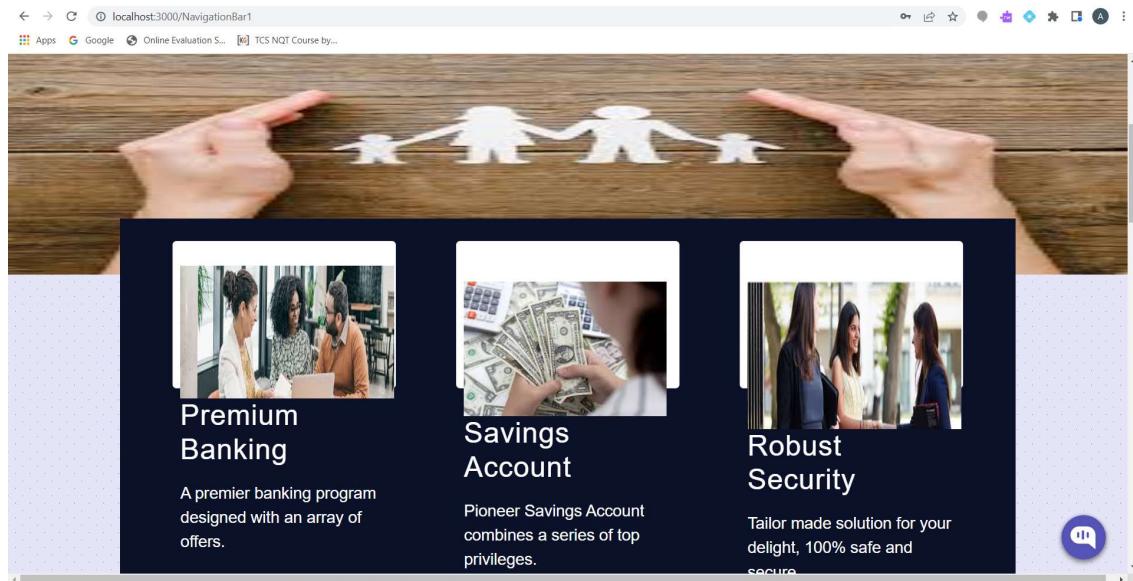
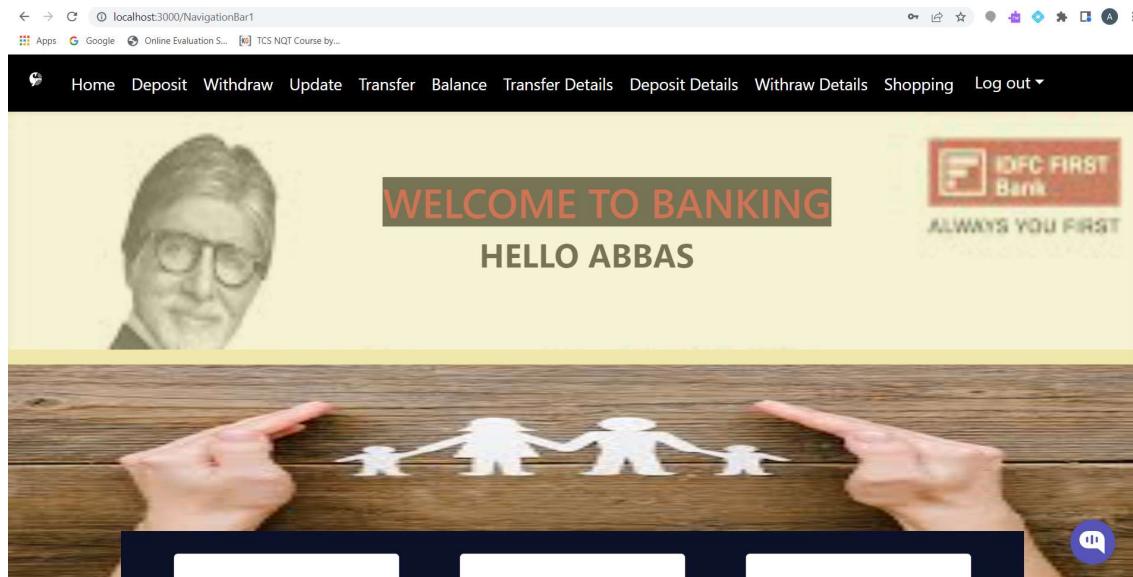
- If account number and password does not match or there is no such account number then alert will be displayed displaying accordingly



- If user's credentials are correct then alert will be displayed saying "Logged in" it will navigate to home page.



HOME PAGE





About us

The Bank has gone through the various phases of its growth trajectory over hundred years of its existence. Growth of BootCat Bank was phenomenal, attaining the status of a national level player in terms of geographical reach and clientele segments. Eighties was characterized by business diversification for the Bank.

Mission

To provide a secure, agile, dynamic and conducive banking environment to customers with commitment to values and unshaken confidence, deploying the best technology, standards, processes and procedures where customer convenience is of significant importance and to increase the stakeholders' value.



Mission

To provide a secure, agile, dynamic and conducive banking environment to customers with commitment to values and unshaken confidence, deploying the best technology, standards, processes and procedures where customer convenience is of significant importance and to increase the stakeholders' value.



Vision

To emerge as a 'Preferred Bank' by pursuing global benchmarks in profitability, operational efficiency, asset quality, risk management and expanding the global reach. Provide advanced and creative banking products and services for all our clients, both locally and internationally.



Stock Market Live Data





Services
Web design
Development
Hosting

About
Company
Team
Legacy

Careers
Job openings
Employee success
Benefits

f t y i
Company Name © 2018

DEPOSIT

Banking Application × +
localhost:3000/Deposit1
Apps Google Online Evaluation S... TCS NQT Course by...

Home Deposit Withdraw Update Transfer Balance Transfer Details Deposit Details Withdraw Details Shopping Log out

 WELCOME TO BANKING DEPOSIT FORM 
ALWAYS YOU FIRST
WEBSITE MOBILE APP

ENTER REQUIRED DETAILS

Amount
Enter Amount

Type here to search 25°C Haze ENG 4:56 PM IN 11/28/2022

With account number logged in, Deposit page will only ask for amount to be deposited to self.

Banking Application × +
localhost:3000/Deposit1
localhost:3000 says
Amount Deposited
OK
WELCOME TO BANKING DEPOSIT FORM 
ALWAYS YOU FIRST
WEBSITE MOBILE APP

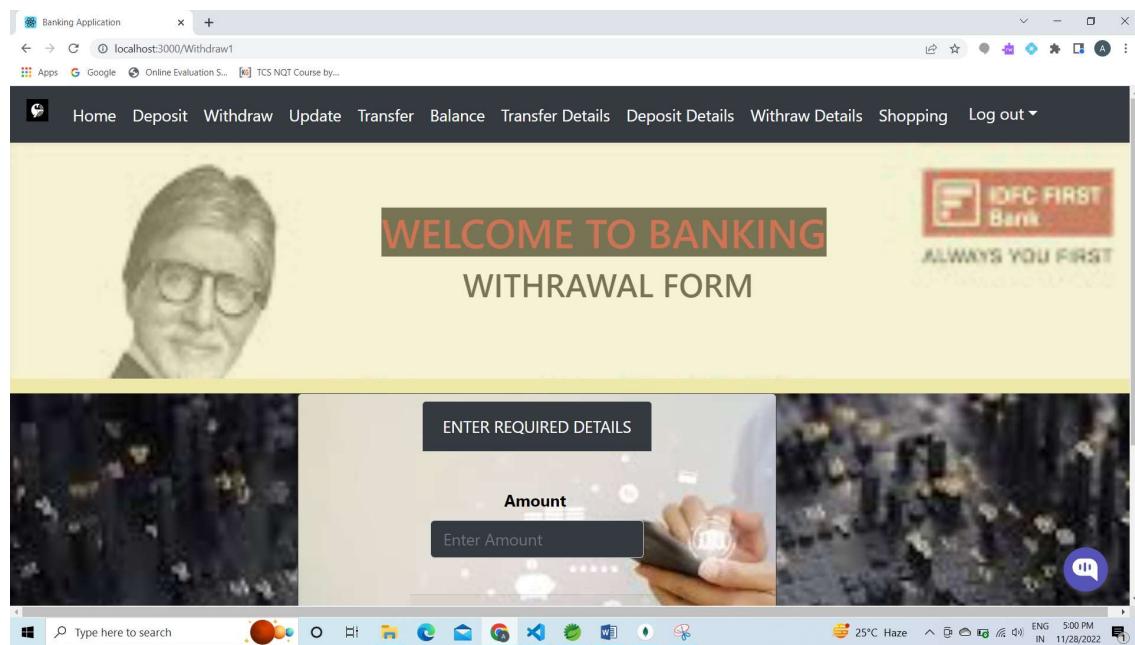
ENTER REQUIRED DETAILS

Amount
1700

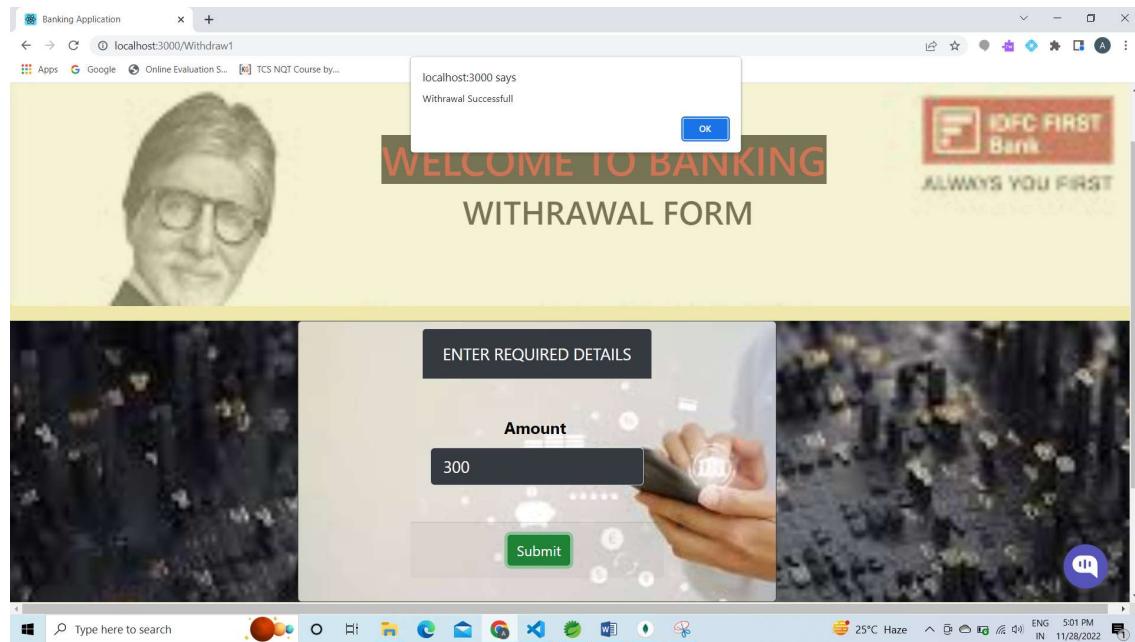
Submit

Type here to search 25°C Haze ENG 4:58 PM IN 11/28/2022

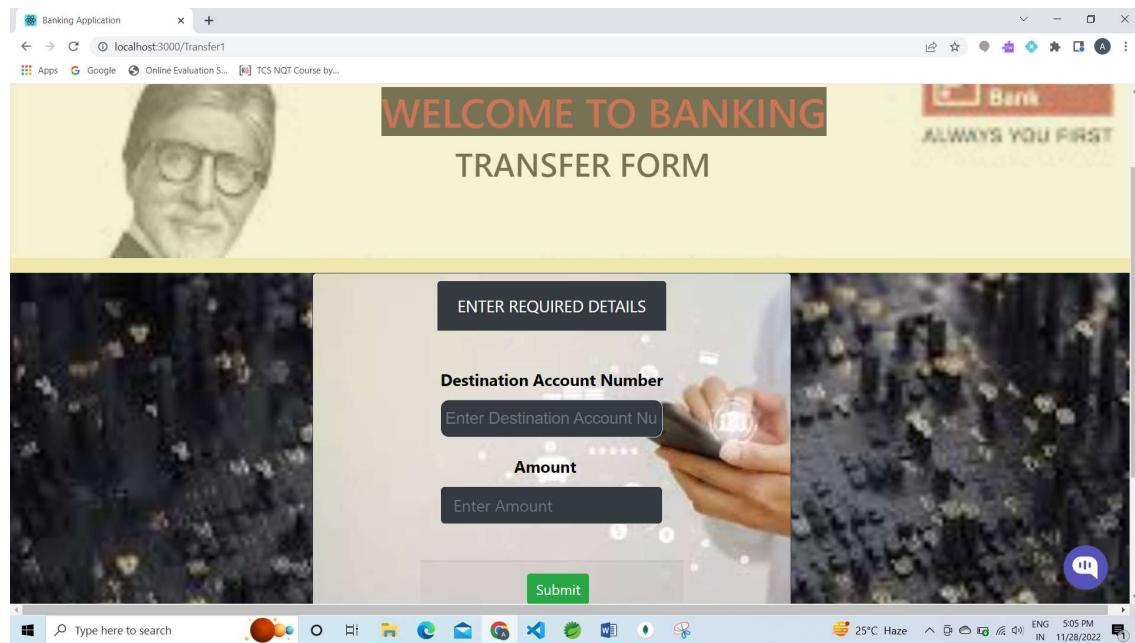
WITHDRAW



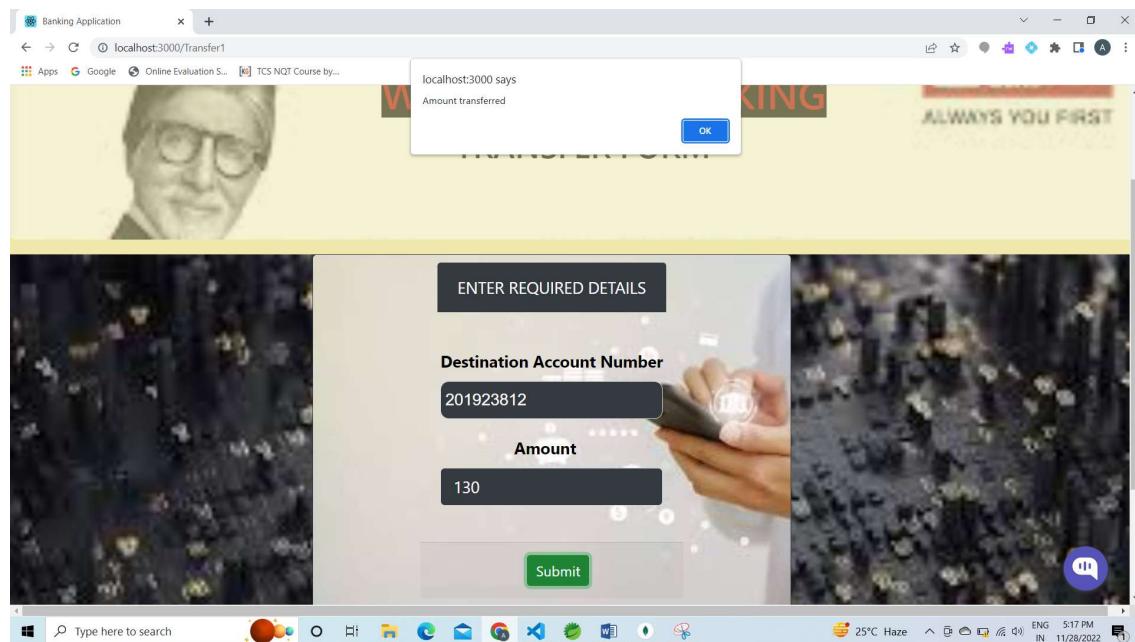
With account number logged in, Deposit page will only ask for amount to be withdrawn from self.



TRANSFER



User have to give the account number to which amount will be transferred.



UPDATE



Banking Application localhost:3000/Update1

New Account number
Enter new Account number

Customer Name
Enter Customer Name

Customer Id
Enter Customer Id

IFSC Code
Enter IFSC code

Password
Enter password

Update

Type here to search 25°C Haze ENG 5:22 PM IN 11/28/2022

User must enter all the fields and in correct format otherwise it won't update the newly entered details.

Banking Application localhost:3000/Update1

localhost:3000 says
IFSC Code is invalid

Customer Name
aditya

Customer Id
123123

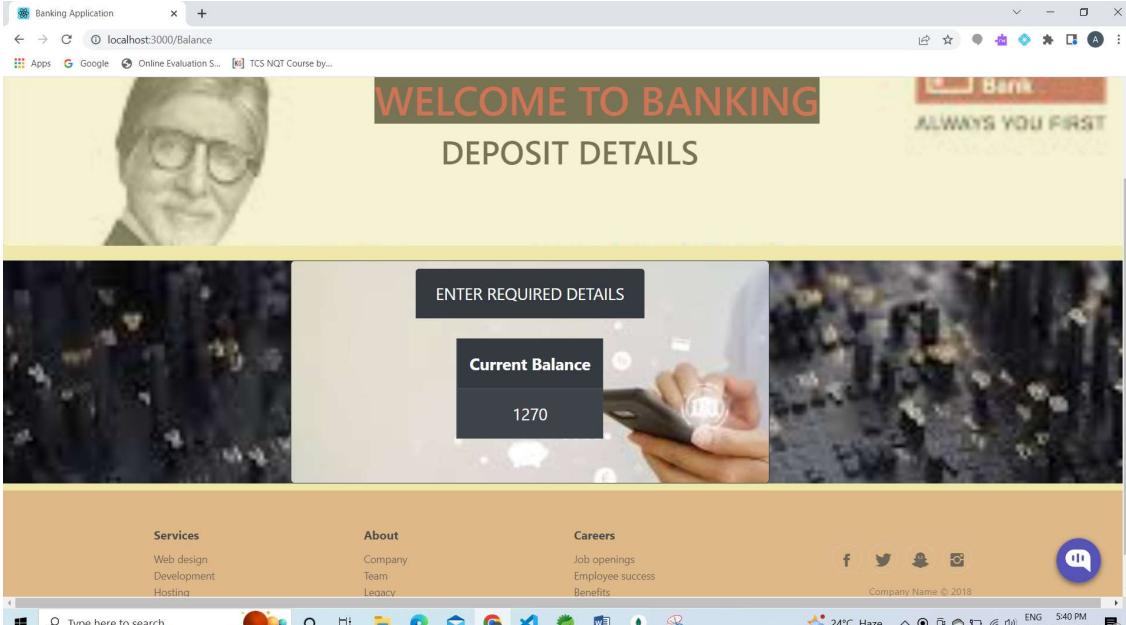
IFSC Code
sc

Password
Adidady123!

Update

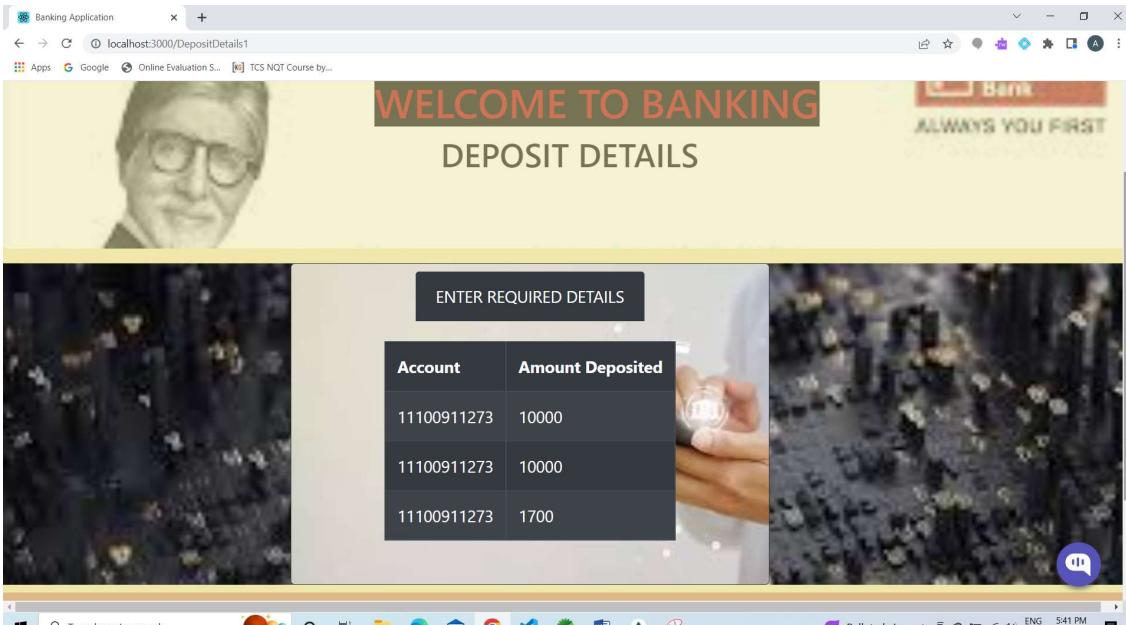
Type here to search 24°C Haze ENG 5:39 PM IN 11/28/2022

BALANCE



The screenshot shows a banking application window titled "WELCOME TO BANKING" with a sub-section "DEPOSIT DETAILS". A large portrait of a man with glasses is on the left. In the center, there is a dark rectangular box containing the text "ENTER REQUIRED DETAILS" and "Current Balance 1270". Below this is a photograph of a hand holding a smartphone. At the bottom, there is a navigation bar with links for "Services", "About", and "Careers", along with social media icons and a weather widget.

DEPOSIT DETAILS



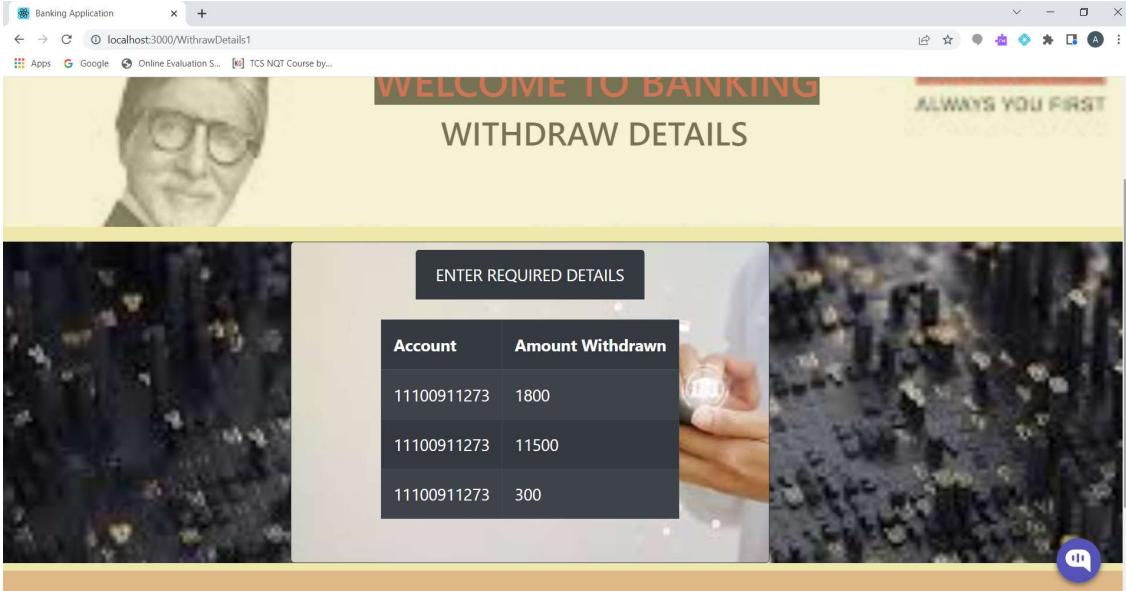
The screenshot shows a banking application window titled "WELCOME TO BANKING" with a sub-section "DEPOSIT DETAILS". A large portrait of a man with glasses is on the left. In the center, there is a dark rectangular box containing the text "ENTER REQUIRED DETAILS" and a table titled "Amount Deposited". The table lists three entries:

Account	Amount Deposited
11100911273	10000
11100911273	10000
11100911273	1700

Below the table is a photograph of a hand holding a smartphone. At the bottom, there is a navigation bar with links for "Services", "About", and "Careers", along with social media icons and a weather widget.

Details will be displayed only of logged in or signed up account number.

WITHDRAW DETAILS

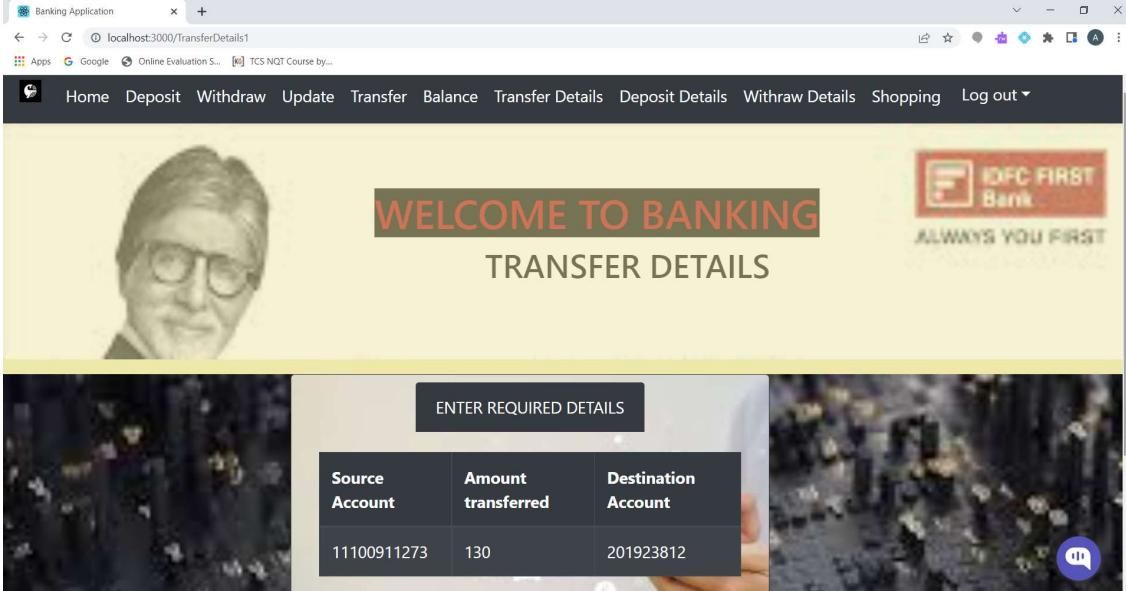


ENTER REQUIRED DETAILS

Account	Amount Withdrawn
11100911273	1800
11100911273	11500
11100911273	300

Details will be displayed only of logged in or signed up account number.

TRANSFER DETAILS



ENTER REQUIRED DETAILS

Source Account	Amount transferred	Destination Account
11100911273	130	201923812

Details will be displayed only of logged in or signed up account number.

SHOPPING PAGE

Banking Application localhost:3000/Shopping

WELCOME TO BANKING
PURCHASE ITEMS

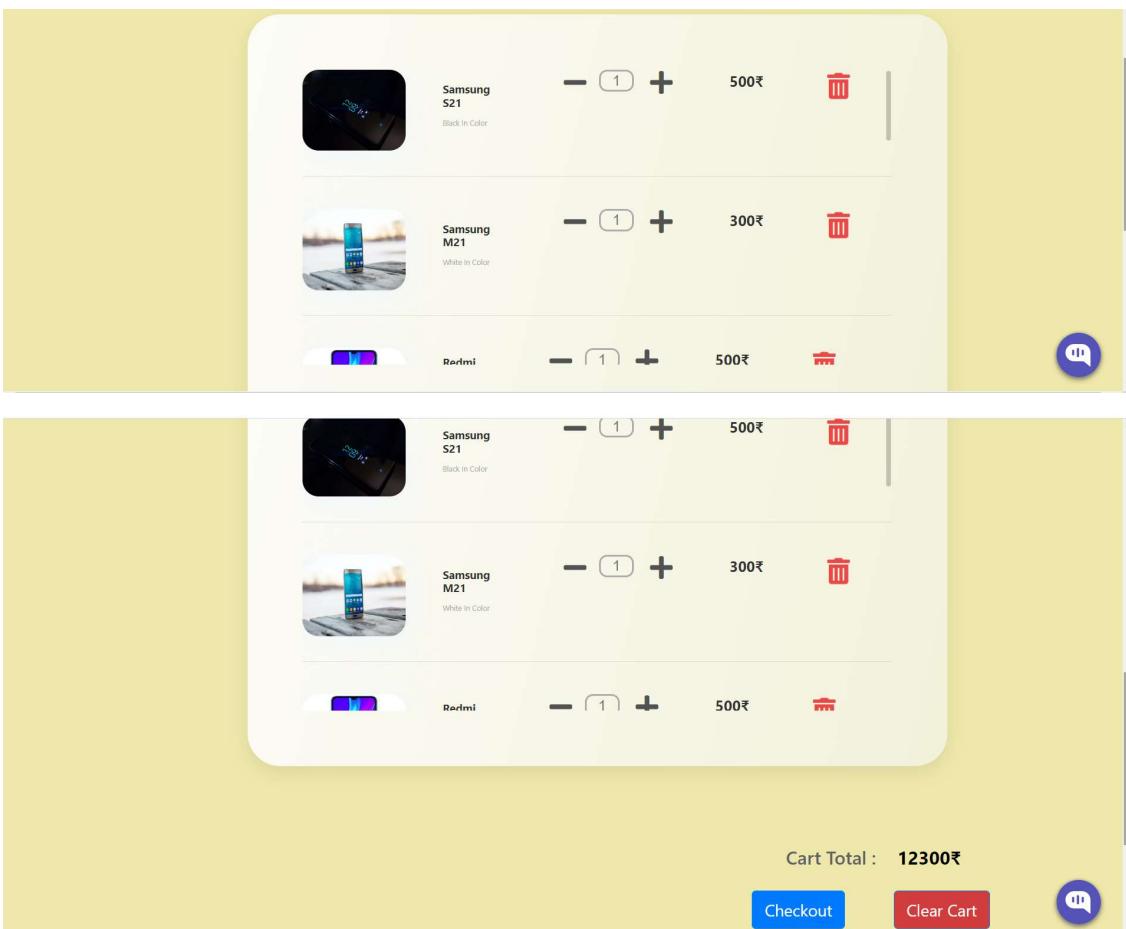
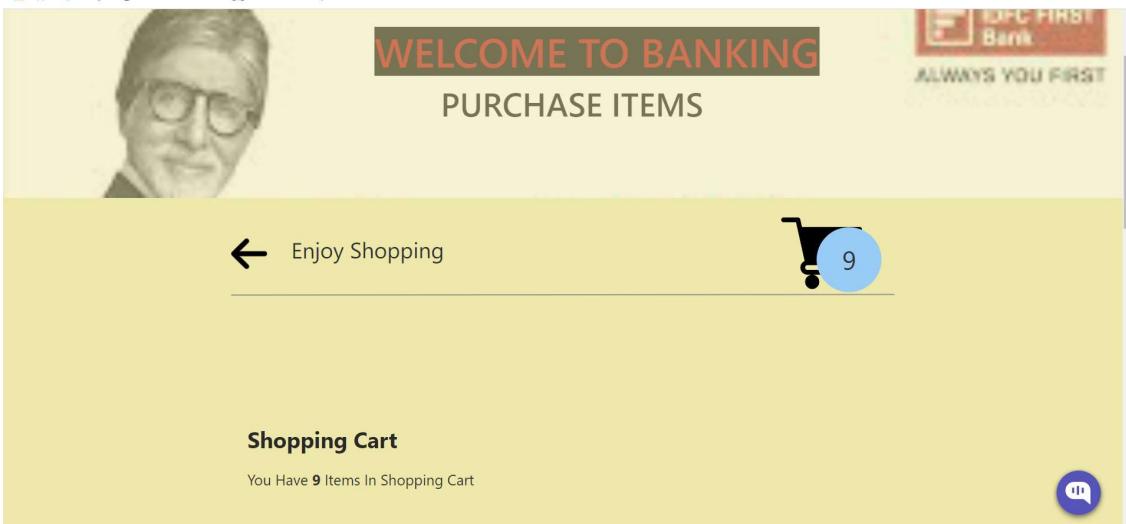
IDFC FIRST Bank
ALWAYS YOU FIRST

Enjoy Shopping

9

Shopping Cart

You Have 9 Items In Shopping Cart



Cart Total : 12300₹

Checkout

Clear Cart



This page is for users to purchase items by deleting or adding to cart. Total amount is being displayed at end.

User can checkout , if total amount is less than current balance of user, then alert will be displayed accordingly and purchase will not happen otherwise purchase will be successful

A screenshot of a web browser window titled "Banking Application". The URL is "localhost:3000/Shopping". The main content shows a shopping cart with a single item: a Redmi 9 smartphone. The cart total is "12300₹". A modal dialog box in the center says "localhost:3000 says Purchase cannot happen due to insufficient balance" with an "OK" button. Below the cart, there are "Checkout" and "Clear Cart" buttons. At the bottom of the page, there is a footer with sections for "Services" (Web design, Development, Hosting), "About" (Company, Team, Legacy), and "Careers" (Job openings, Employee success, Benefits). Social media links (Facebook, Twitter, LinkedIn) and a copyright notice ("Company Name © 2018") are also present. The taskbar at the bottom shows various application icons and the system clock.

A screenshot of a web browser window titled "Banking Application". The URL is "localhost:3000/Shopping". The main content shows a shopping cart with a single item: a Redmi 9 smartphone. The cart total is "1000₹". A modal dialog box in the center says "localhost:3000 says Purchase Successfull.1000.0 Deducted from your current balance" with an "OK" button. Below the cart, there are "Checkout" and "Clear Cart" buttons. At the bottom of the page, there is a footer with sections for "Services" (Web design, Development, Hosting), "About" (Company, Team, Legacy), and "Careers" (Job openings, Employee success, Benefits). Social media links (Facebook, Twitter, LinkedIn) and a copyright notice ("Company Name © 2018") are also present. The taskbar at the bottom shows various application icons and the system clock.



Banking Application × +

localhost:3000/Balance

Apps Google Online Evaluation S... TCS NQT Course by...

Home Deposit Withdraw Update Transfer Balance Transfer Details Deposit Details Withdraw Details Shopping Log out

WELCOME TO BANKING

DEPOSIT DETAILS

ENTER REQUIRED DETAILS

Current Balance
270

22°C Haze ENG 6:03 PM IN 11/28/2022

Current Balance is now 270 from 1270.

LOGOUT OR LOGIN AS ADMIN

Banking Application × +

localhost:3000/Balance

localhost:3000/Login

Apps Google Online Evaluation S... TCS NQT Course by...

Home Deposit Withdraw Update Transfer Balance Transfer Details Deposit Details Withdraw Details Shopping Log out

WELCOME TO BANKING

DEPOSIT DETAILS

ENTER REQUIRED DETAILS

Current Balance
270

Log out

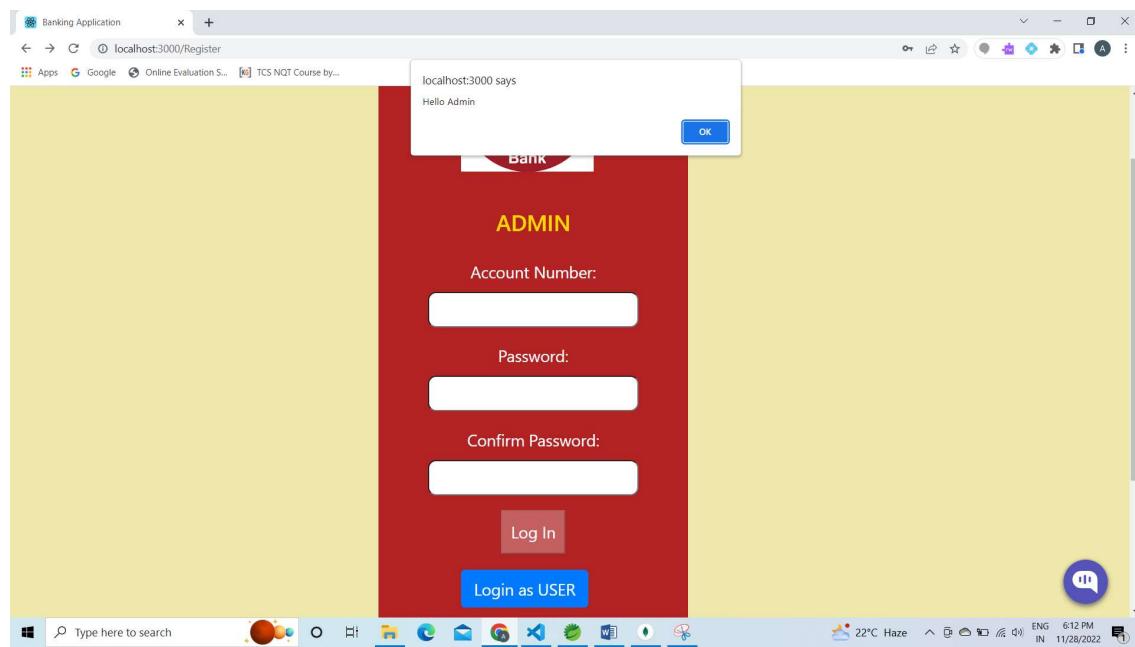
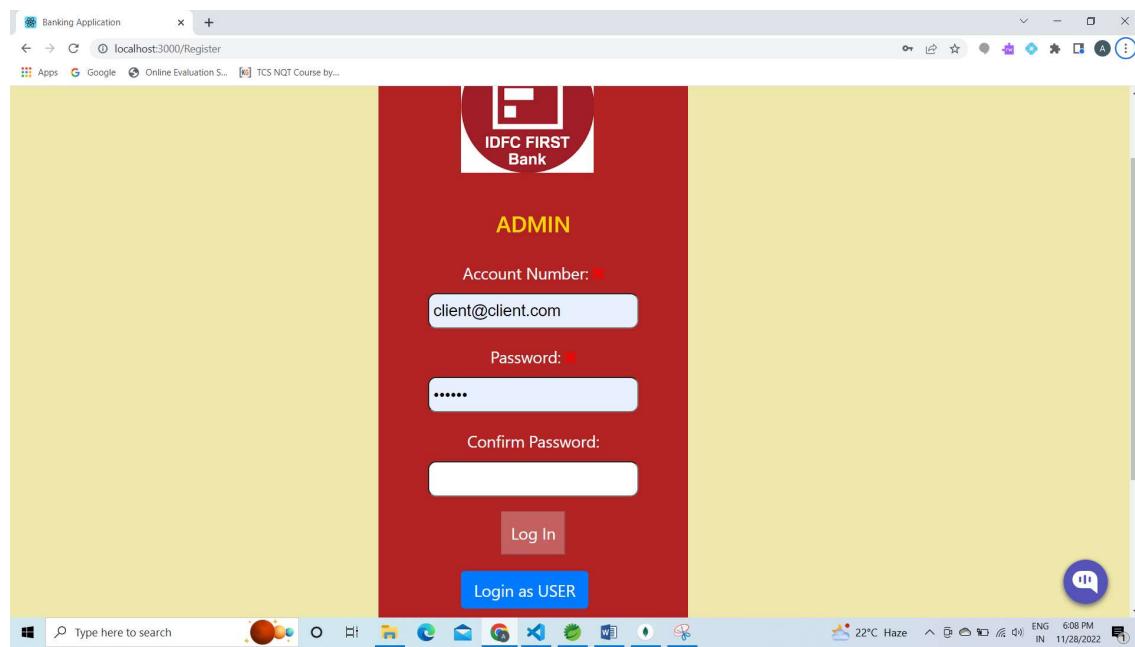
Log in as Admin

22°C Haze ENG 6:04 PM IN 11/28/2022

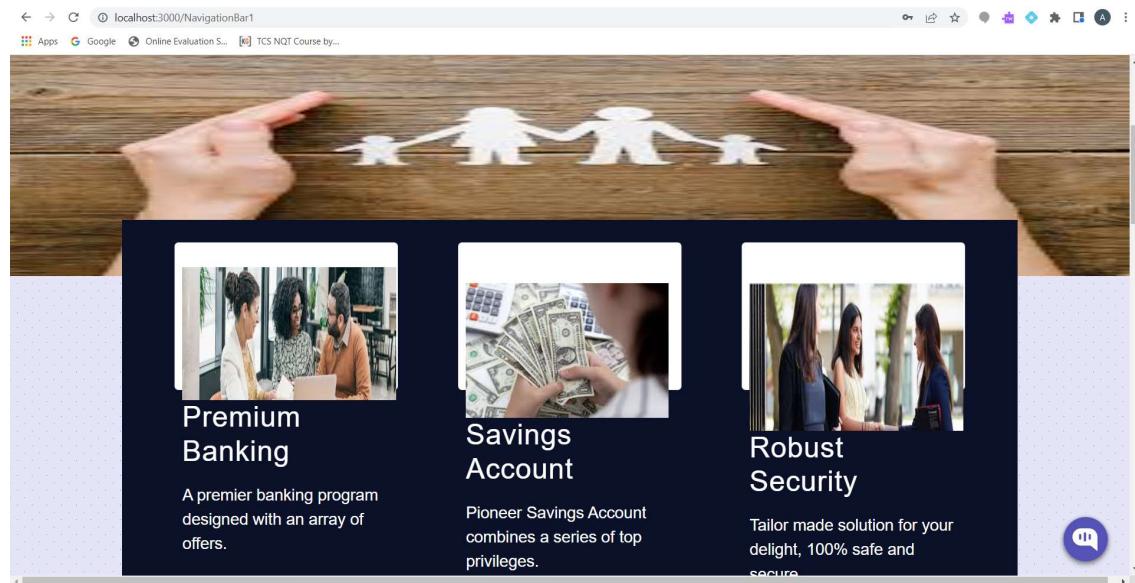
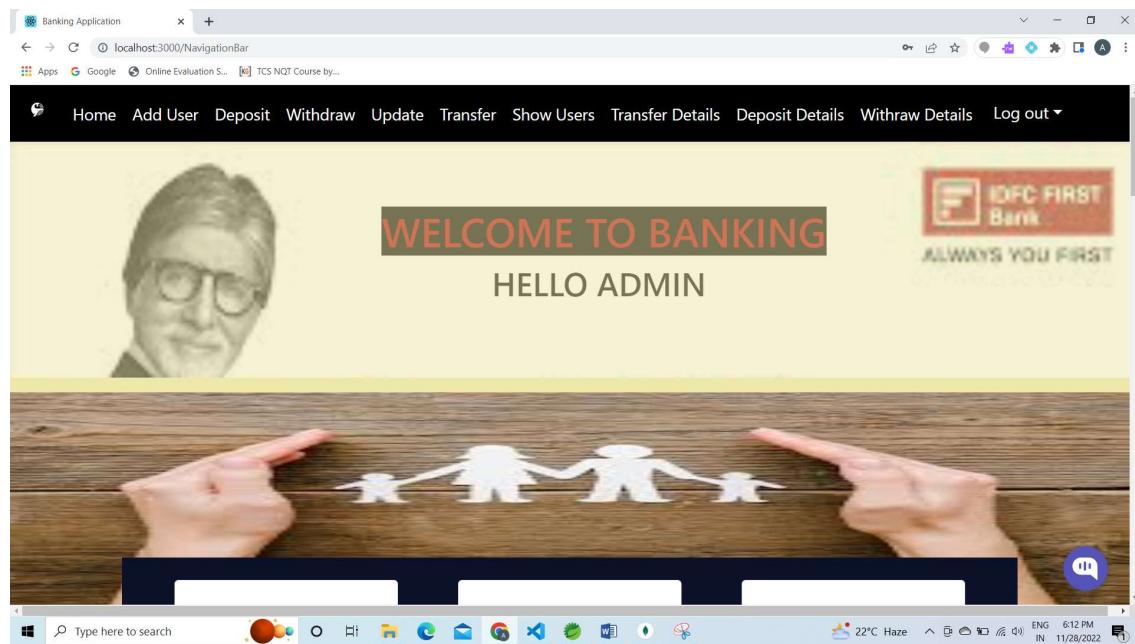
ADMIN

As above operations are being carried out similarly admin can do all of it for all the users and this time admin has to specify the account number.

LOGIN PAGE



HOME PAGE





About us

The Bank has gone through the various phases of its growth trajectory over hundred years of its existence. Growth of BootCat Bank was phenomenal, attaining the status of a national level player in terms of geographical reach and clientele segments. Eighties was characterized by business diversification for the Bank.

Mission

To provide a secure, agile, dynamic and conducive banking environment to customers with commitment to values and unshaken confidence, deploying the best technology, standards, processes and procedures where customer convenience is of significant importance and to increase the stakeholders' value.



Mission

To provide a secure, agile, dynamic and conducive banking environment to customers with commitment to values and unshaken confidence, deploying the best technology, standards, processes and procedures where customer convenience is of significant importance and to increase the stakeholders' value.

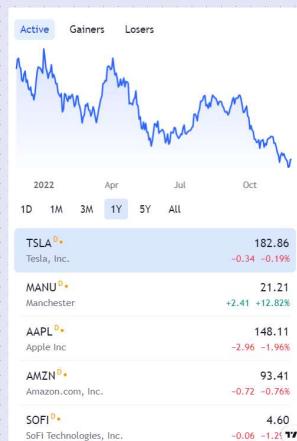


Vision

To emerge as a 'Preferred Bank' by pursuing global benchmarks in profitability, operational efficiency, asset quality, risk management and expanding the global reach. Provide advanced and creative banking products and services for all our clients, both locally and internationally.



Stock Market Live Data





ADD USER

Account number
Enter Account number

Customer Name
Enter Customer Name

Customer Id
Enter Customer Id

IFSC Code
Enter IFSC code

Balance
Enter password

Password
Enter password

If any entry is invalid and user is already there , then user will not be added otherwise user will be created .

DEPOSIT

WELCOME TO BANKING

DEPOSIT FORM

ENTER REQUIRED DETAILS

Account Number
Enter Account Number

Amount
Enter Amount

Submit



Admin has to give account number and amount for deposit operation to happen.

WITHDRAW

Admin has to give account number and amount for withdraw operation to happen.

TRANSFER

Admin has to give source account number, destination account number and amount for transfer operation to happen.

UPDATE

Banking Application × +

localhost:3000/Update

Apps Google Online Evaluation S... TCS NQT Course by...

WELCOME TO BANKING

UPDATE FORM

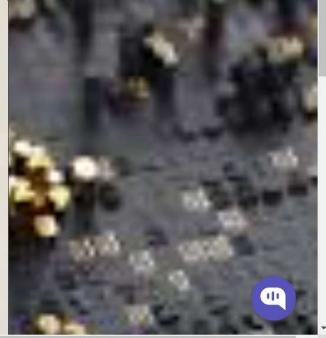


Enter required details

Old Account number
Enter Account number

New Account number
Enter new Account number

Customer Name
Enter Customer Name



22°C Haze 6:23 PM ENG IN 11/28/2022

Banking Application × +

localhost:3000/Update

Apps Google Online Evaluation S... TCS NQT Course by...

Customer Id
Enter Customer Id

IFSC Code
Enter IFSC code

Balance
Enter Balance

Password
Enter password

Submit



Services About Careers 22°C Haze 6:23 PM ENG IN 11/28/2022

DEPOSIT DETAILS

Banking Application +

localhost:3000/DepositDetails

Apps Google Online Evaluation S... TCS NQT Course by...

WELCOME TO BANKING

DEPOSIT DETAILS

DEPOSIT LIST

Account	Amount Deposited to self
23456	123
23410	123
23456	10
11100911283	10

Type here to search 22°C Clear ENG IN 6:27 PM 11/28/2022

Banking Application +

localhost:3000/DepositDetails

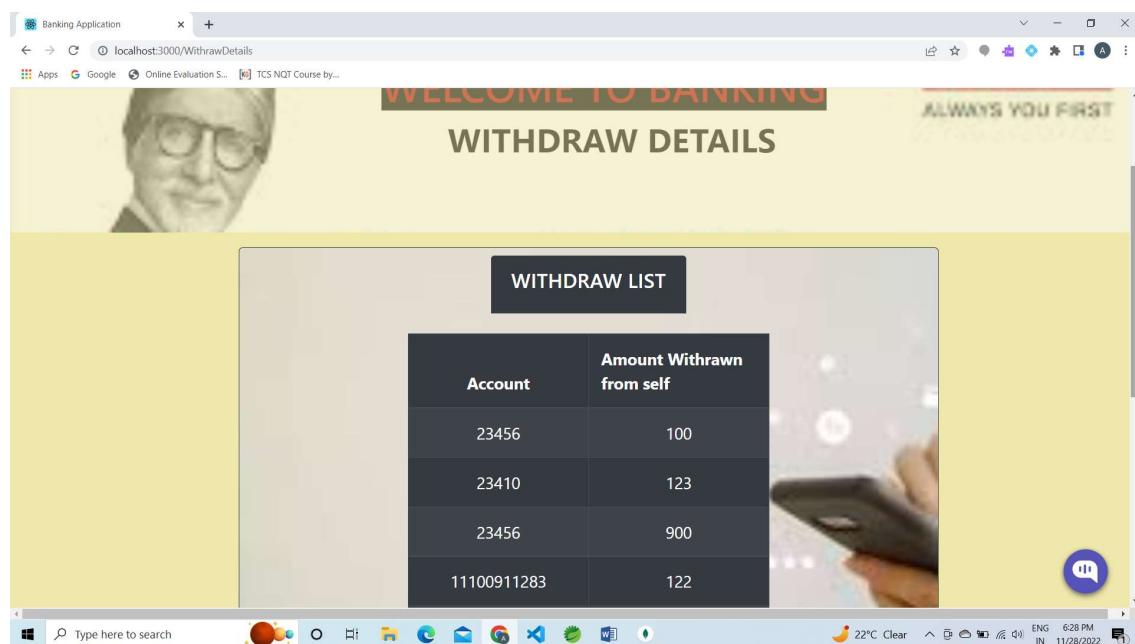
Apps Google Online Evaluation S... TCS NQT Course by...

11100911285	10
11100911283	14
11100911284	13
11100911284	109
11100911273	10000
11100911273	10000
11100911273	1700

Type here to search 22°C Clear ENG IN 6:28 PM 11/28/2022

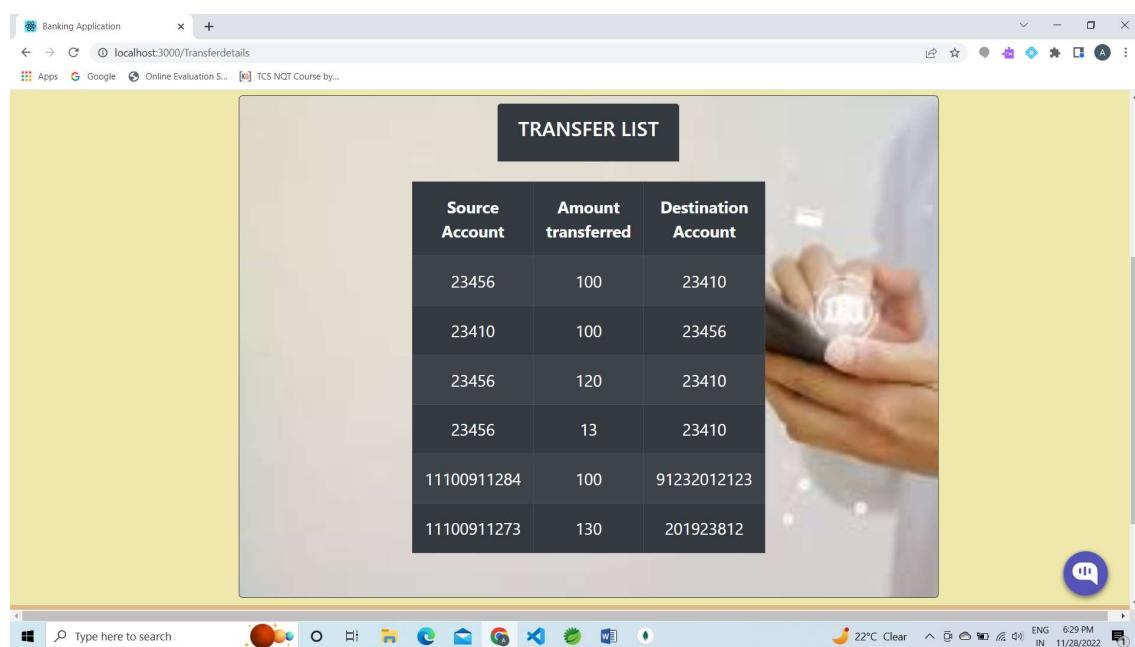
All the users details will be displayed.

WITHDRAW DETAILS



Account	Amount Withdrawn from self
23456	100
23410	123
23456	900
11100911283	122

TRANSFER DETAILS



Source Account	Amount transferred	Destination Account
23456	100	23410
23410	100	23456
23456	120	23410
23456	13	23410
11100911284	100	91232012123
11100911273	130	201923812

USER LIST AND DETAILS

Banking Application × +

localhost:3000/Userslist

All Entries Secured

User ID	User Name	Customer Id	IFSC Code	Balance	Password
23456	Aditya	145789	SBIN0178012	12000	Adidady123!
23410	lak	201913		12012	Adidady123!
11100911212	John	123121	SBIN0125620	123121	Alana@14
138123123	Manya Surve	123123	SBIN0181231	0	Adidady123!

Banking Application × +

localhost:3000/Userslist

11100911212	John	123121	SBIN0125620	123121	Alana@14
138123123	Manya Surve	123123	SBIN0181231	0	Adidady123!
91232012123	Naman	190223	SBIN0701921	100	Adidady123!
11100911284	Nirmit	109231	SBIN0125620	3190	Adidady123!
11100911273	abbas	124567	SBIN0172458	270	Abbas@123
201923814	Brock Lesnar	187290	SBIN0182745	10030	SigmaMale@123

Services

Web design
Development
Hosting

About

Company
Team
Legacy

Careers

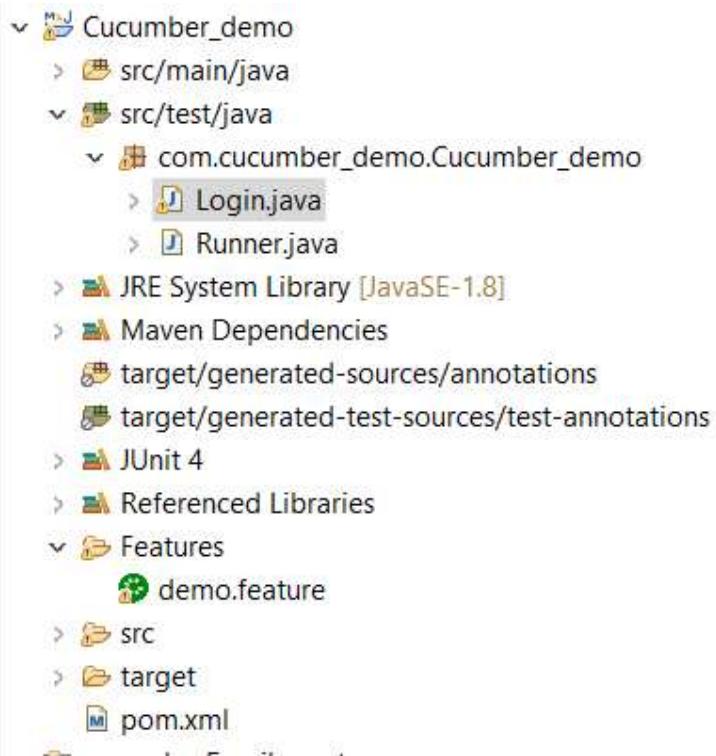
Job openings
Employee success
Benefits

f t y i

Company Name © 2018

AUTOMATION TESTING USING CUCUMBER

PROJECT STRUCTURE



TESTING IMPLEMENTATION

- **Test Scenario- Login Functionality**

Navigate to login page

Test case id	Test case description	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
TU01	Check Customer Login with valid Data	Enter correct account number and password	Accno-11100911273 Password-Abbas@123	User should login to the application	As expected	Pass
TU02	Check Customer Login with valid Password and invalid acc number	Enter incorrect account number and correct password	Accno-11100911212 Password-Abbas@123	User shouldn't be able to login to application	As expected	Pass
TU03	Check Customer Login with invalid Password	Enter correct account number and	Accno-11100911273 Password-Abbas@129	User shouldn't be able to login to application	As expected	Pass



	and valid acc number	incorrect password				
--	-------------------------	-----------------------	--	--	--	--

OUTPUT

TEST CASE

- TU01-** Login Page with valid data

A screenshot of a web browser window titled "Banking Application". The URL bar shows "localhost:3000/Login". The page itself is titled "Bank Login Page For User". It features the IDFC FIRST Bank logo at the top. Below it, there are three input fields: "Account Number:" containing "11100911273" with a green checkmark, "Password:" with an empty field, and "Confirm Password:" with an empty field. The browser's taskbar at the bottom shows various pinned icons and the system tray on the right.

- Credentials being entered

A screenshot of a web browser window titled "Banking Application". The URL bar shows "localhost:3000/Login". The page is identical to the previous one, but now the "Password:" field contains "*****" and has a green checkmark, indicating it is also valid. The other fields and the overall layout remain the same. The browser's taskbar and system tray are visible at the bottom.

Bank Application x + ← → C localhost:3000/Login Chrome is being controlled by automated test software.

IDFC FIRST Bank

Bank Login Page For User

Account Number: ✓
11100911273

Password: ✓
.....

Confirm Password: ✓
.....

Log In

Don't have an account? [Create Account](#)

Type here to search

localhost:3000 says
Logged in

OK

Bank Login Page For User

Account Number:

Password:

Confirm Password:

Log In

Don't have an account? [Create Account](#)

Type here to search

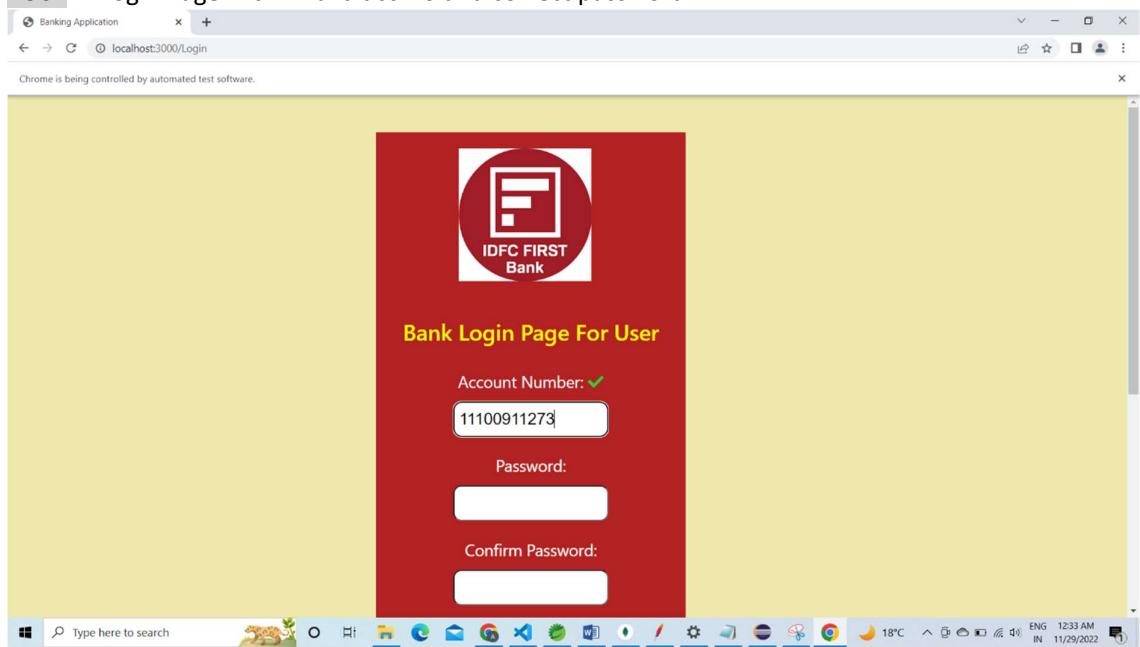
18°C ENG 12:34 AM IN 11/29/2022

18°C ENG 12:36 AM IN 11/29/2022

- Navigate to home page after logged in



- TU02- Login Page with invalid acc No and correct password



Incorrect credentials being entered



Bank Application x +

localhost:3000/Login

Chrome is being controlled by automated test software.



Bank Login Page For User

Account Number: ✓
11100911273

Password: ✓

Confirm Password:

Don't have an account? [Create](#)

Type here to search

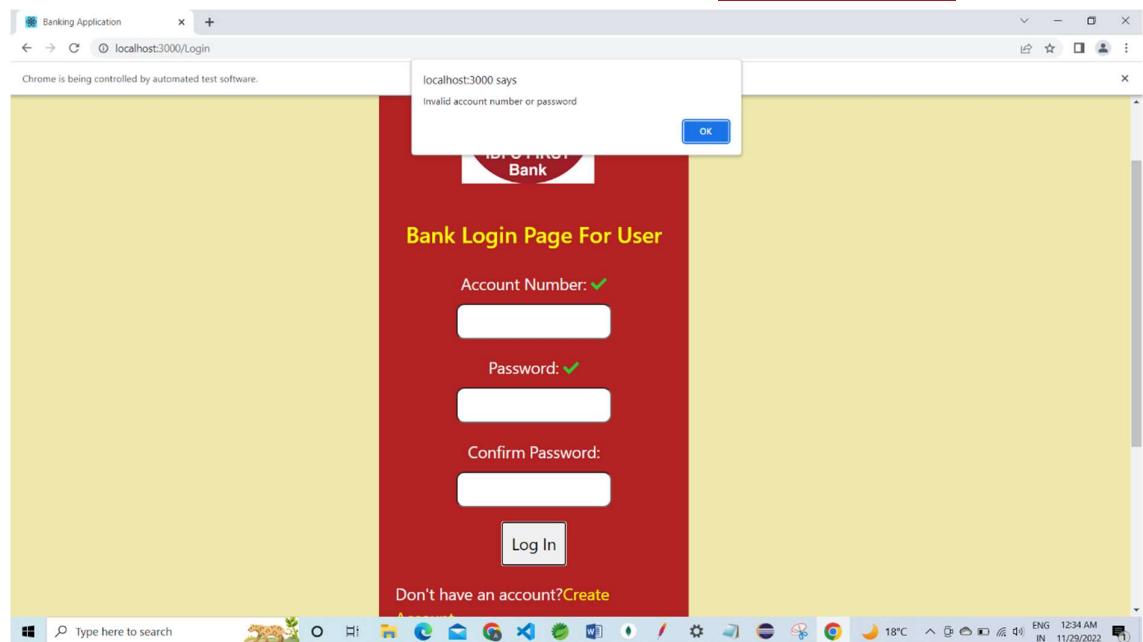
Banking Application x +

localhost:3000/Login

Chrome is being controlled by automated test software.

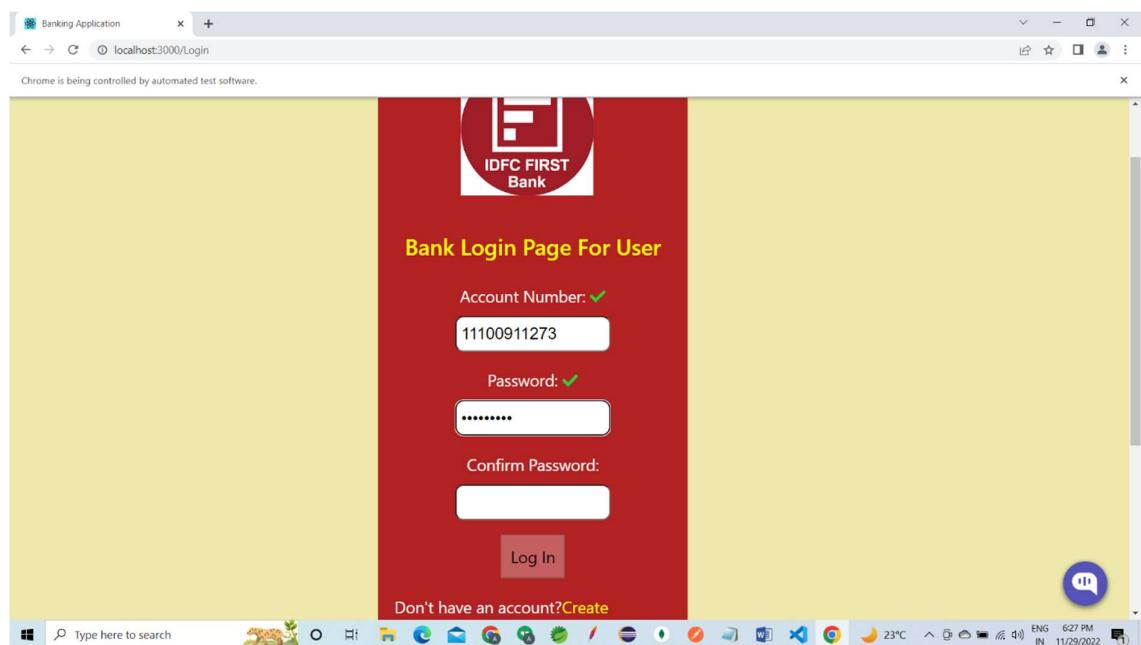
18°C ENG 12:33 AM IN 11/29/2022

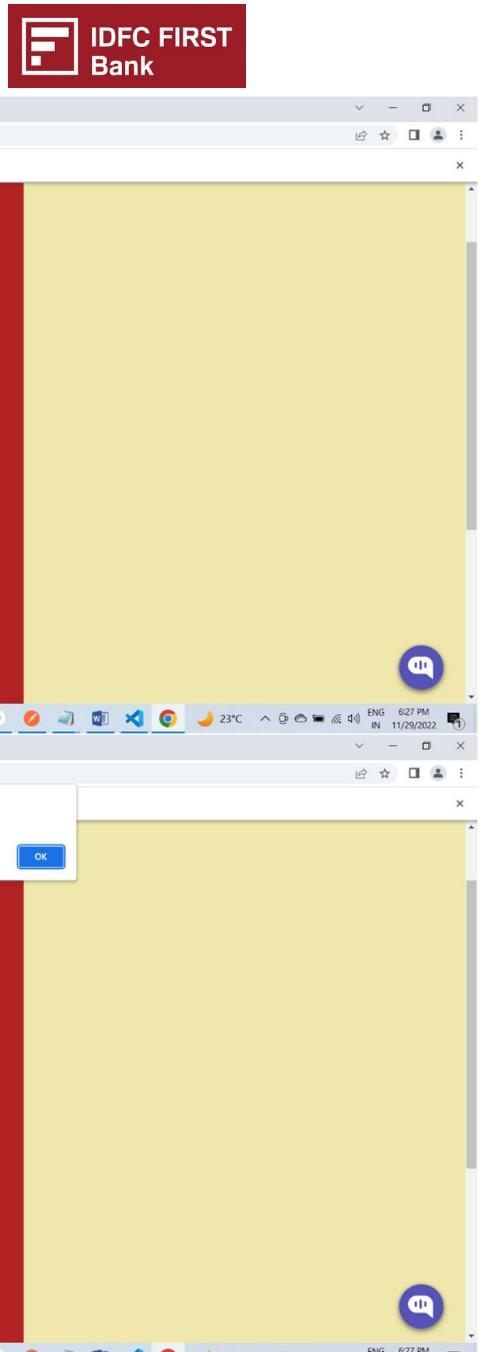
18°C ENG 12:34 AM IN 11/29/2022



Invalid accNo or password

- **TU03-** Login Page with invalid acc No and correct password



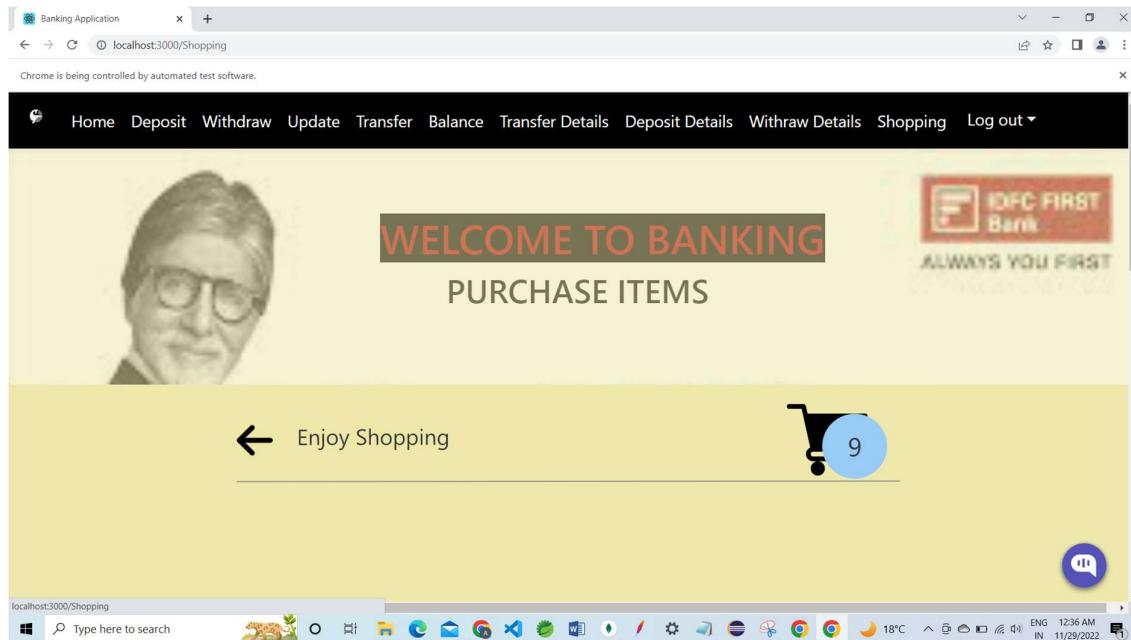


➤ **Test scenario – Checking Shopping cart Functionality**

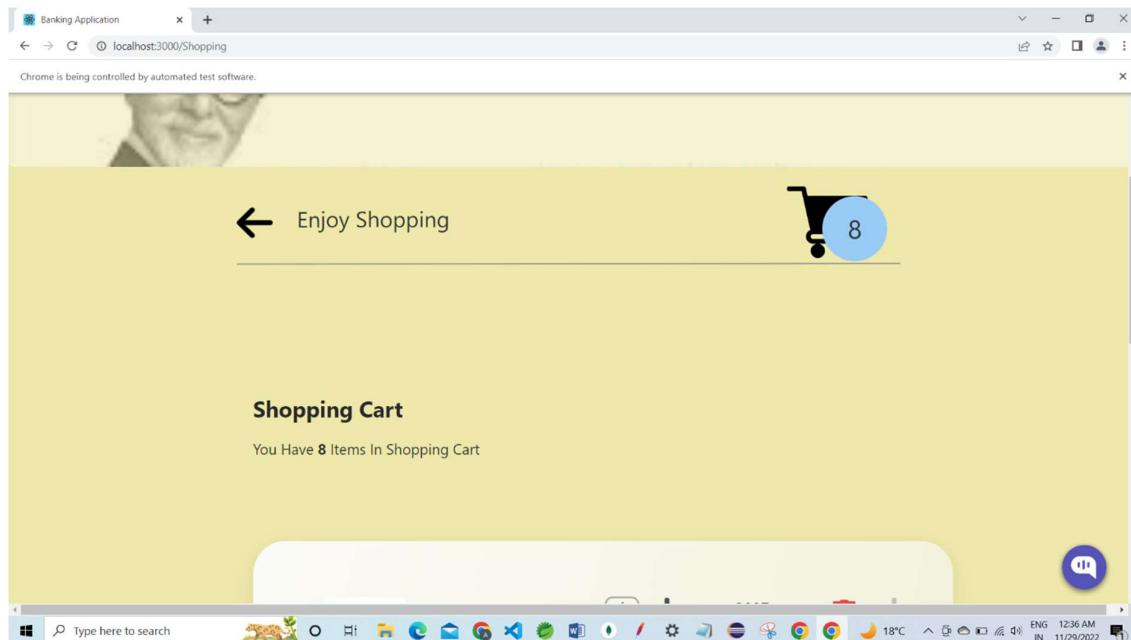
Navigate to Shopping page

Test case id	Test case description	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
TS01	Check Shopping Cart functionality if checkout amount	Remove 2 items and then add 1 item.	Remove Samsung S21 and Smasung M21 and	Purchase should be unsuccessfull.	As expected	PASS

	generated is greater than current balance.		then add Redmi 9. Then click to checkout button.			
--	--	--	--	--	--	--



- Remove 2 items and add 1 item





Banking Application localhost:3000/Shopping

Chrome is being controlled by automated test software.

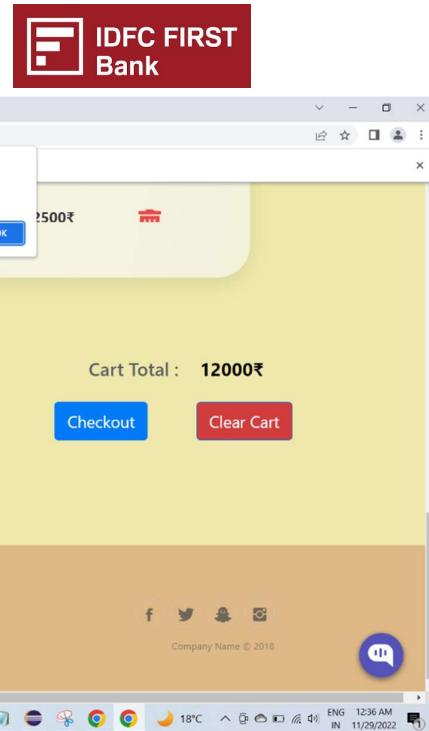
A screenshot of a web browser window titled "Banking Application" at "localhost:3000/Shopping". The page features a header with a user profile picture and the text "Enjoy Shopping". Below this is a shopping cart icon with a blue circle containing the number "7". The main content area is labeled "Shopping Cart" and displays the message "You Have 7 Items In Shopping Cart".

A screenshot of a web browser window titled "Banking Application" at "localhost:3000/Shopping". The page features a header with a user profile picture and the text "Enjoy Shopping". Below this is a shopping cart icon with a blue circle containing the number "7". The main content area is labeled "Shopping Cart" and displays the message "You Have 7 Items In Shopping Cart".

A screenshot of a web browser window titled "Banking Application" at "localhost:3000/Shopping". The page features a header with a user profile picture and the text "Enjoy Shopping". Below this is a shopping cart icon with a blue circle containing the number "8". The main content area is labeled "Shopping Cart" and displays the message "You Have 8 Items In Shopping Cart".

A screenshot of a web browser window titled "Banking Application" at "localhost:3000/Shopping". The page features a header with a user profile picture and the text "Enjoy Shopping". Below this is a shopping cart icon with a blue circle containing the number "8". The main content area is labeled "Shopping Cart" and displays the message "You Have 8 Items In Shopping Cart".

- Proceed to checkout

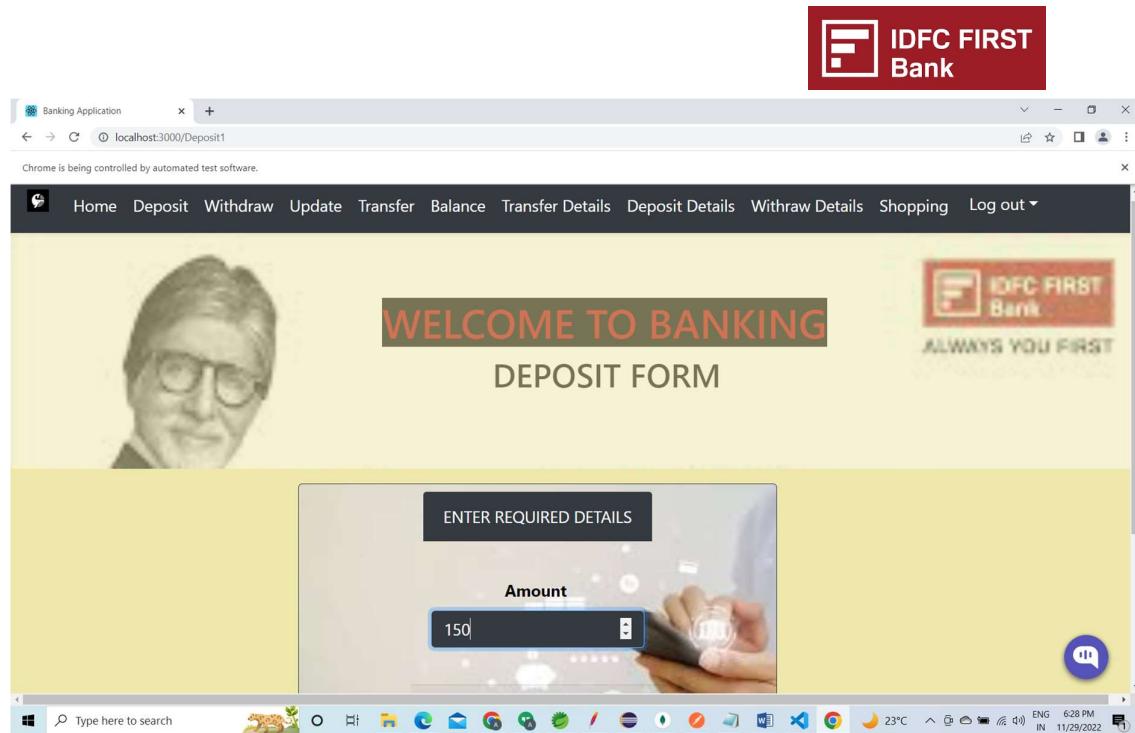


- **Test scenario – Checking deposit amount functionality**
- Navigate to Deposit page

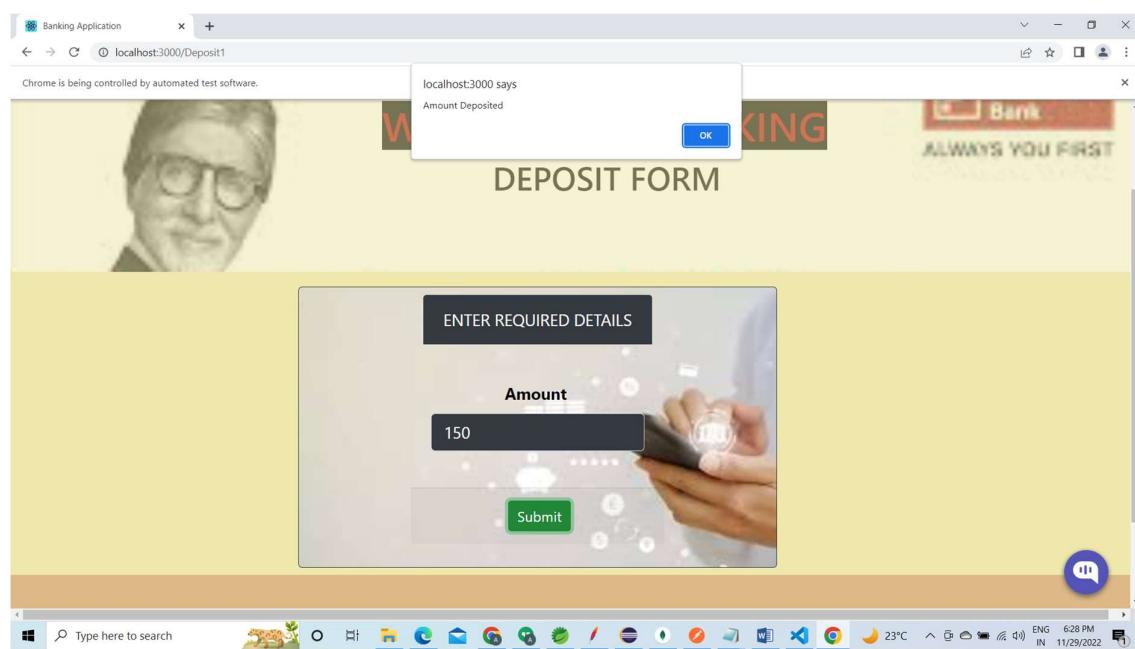
Test case id	Test case description	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
TD01	Check Deposit functionality	Enter deposit amount	Amount-200	Amount should be deposited	As expected	PASS

OUTPUT

- TDO1 - Check Deposit functionality



Enter the amount



➤ **Test scenario – Checking withdraw amount functionality**

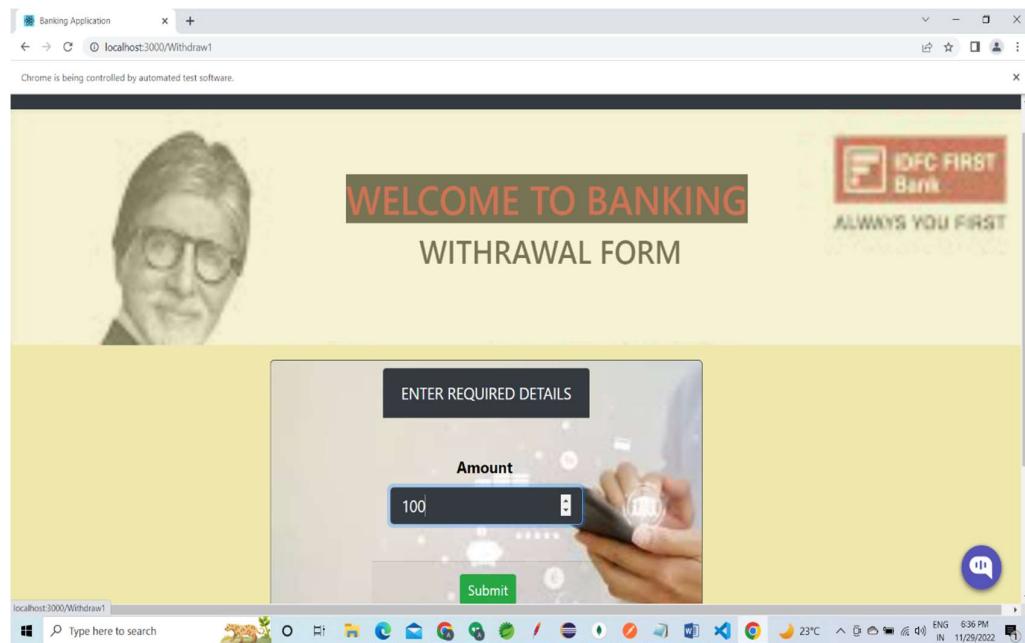
Navigate to Withdraw page

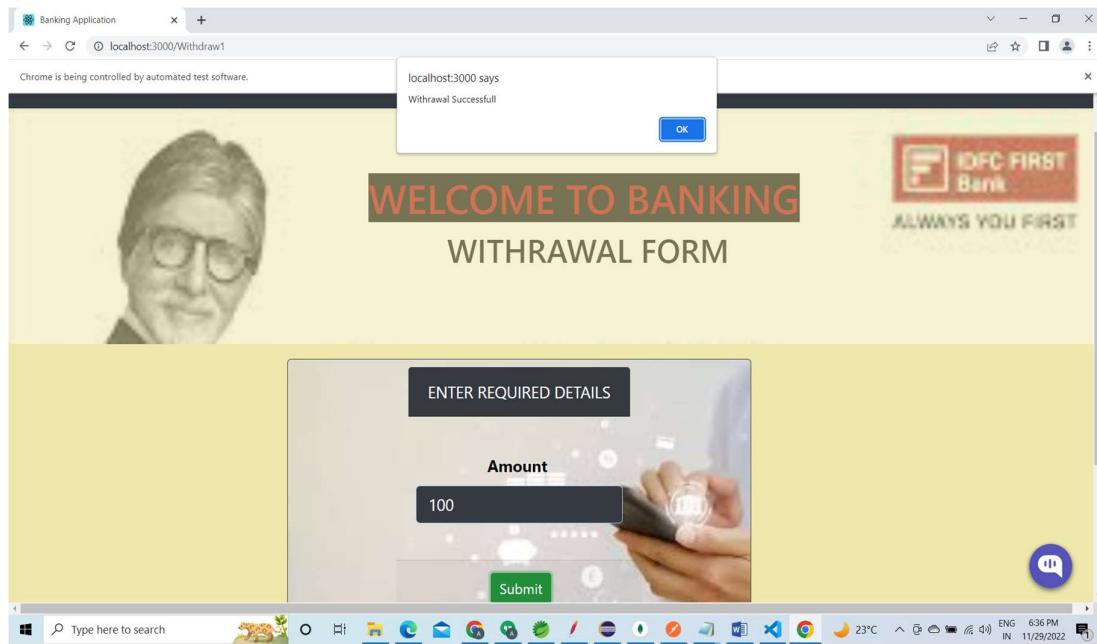
Test case id	Test case description	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
TW01	Check Withdraw functionality with	Enter withdraw amount	Amount-100	Amount should be Withdrawn from	As expected	PASS

	amount less than current balance			current balance		
TW02	Check Withdraw functionality with amount more than current balance	Enter withdraw amount	Amount-11120	Amount shouldn't be Withdrawn from current balance	As expected	PASS

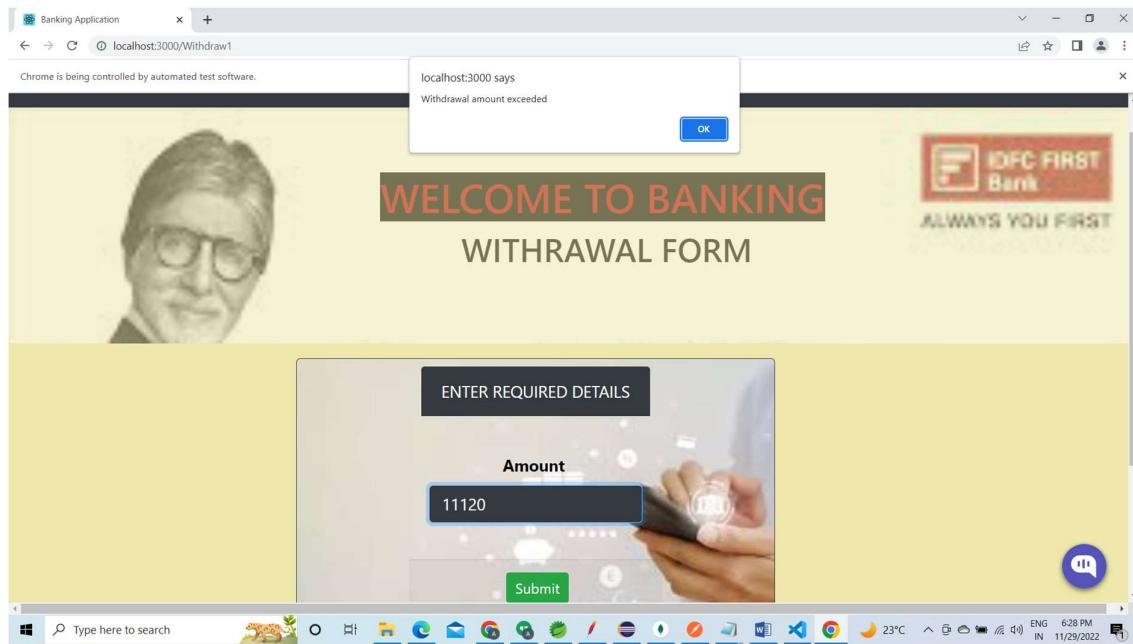
OUTPUT

- TW01 - Withdraw amount with amount less than the current balance





- TW02- Withdraw amount with amount more than the current balance



➤ **Test scenario – Transfer Amount**

Navigate to Transfer page

Test case id	Test case description	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
TA01	Check Transfer functionality	Enter Transfer amount	Amount-150	Amount should be	As expected	PASS

	with amount less than current balance and correct destination account number	and destination account number	Account number- 12938109710	Transferred from current balance to destination account number		
TA02	Check Transfer functionality with amount more than current balance and correct destination account number	Enter Transfer amount and destination account number	Amount- 2800000, accno- 12938109710	Amount shouldn't be Transferred from current balance to destination account number	As expected	PASS
TA03	Check Transfer functionality with amount less than current balance and incorrect destination account number.	Enter Transfer amount and destination account number	Accnumber- 13331023231 Amount -120	Amount shouldn't be Transferred from current balance to destination account number	As expected	PASS

OUTPUT

- TA01 – Transfer money with amount less than current balance and correct destination account number.

Banking Application × + ← → C ⌂ localhost:3000/Transfer1 Chrome is being controlled by automated test software.

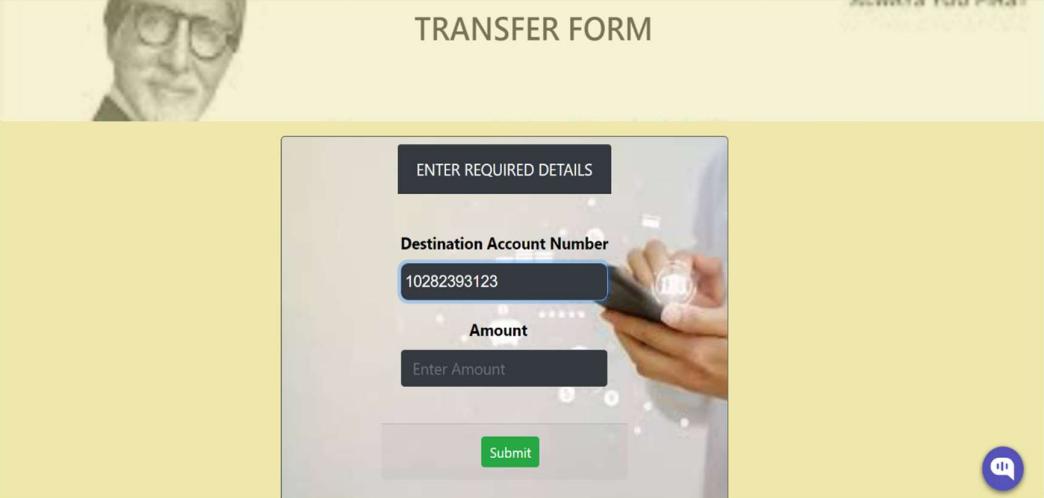
TRANSFER FORM

ENTER REQUIRED DETAILS

Destination Account Number
10282393123

Amount
Enter Amount

Submit



Type here to search



23°C ENG 6:37 PM IN 11/29/2022

Banking Application × + ← → C ⌂ localhost:3000/Transfer1 Chrome is being controlled by automated test software.

localhost:3000 says
Amount transferred

OK

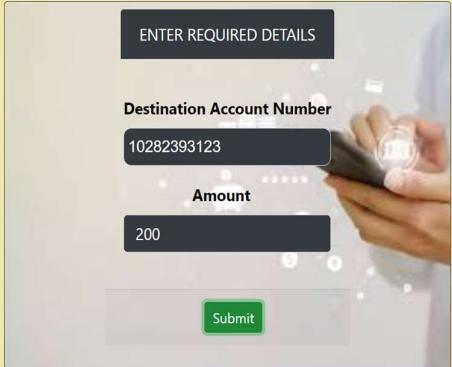
TRANSFER FORM

ENTER REQUIRED DETAILS

Destination Account Number
10282393123

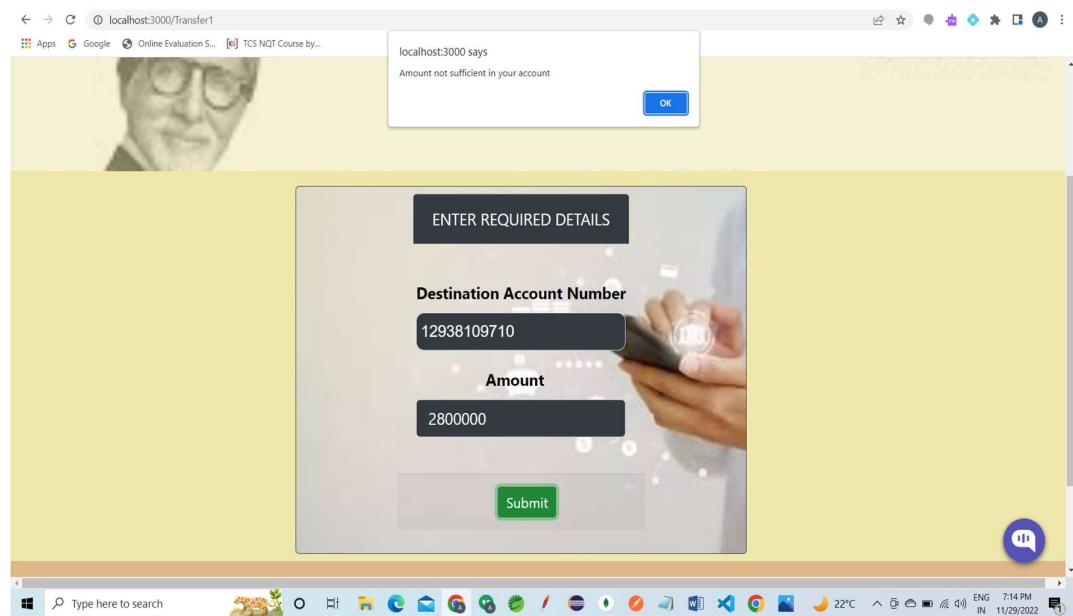
Amount
200

Submit

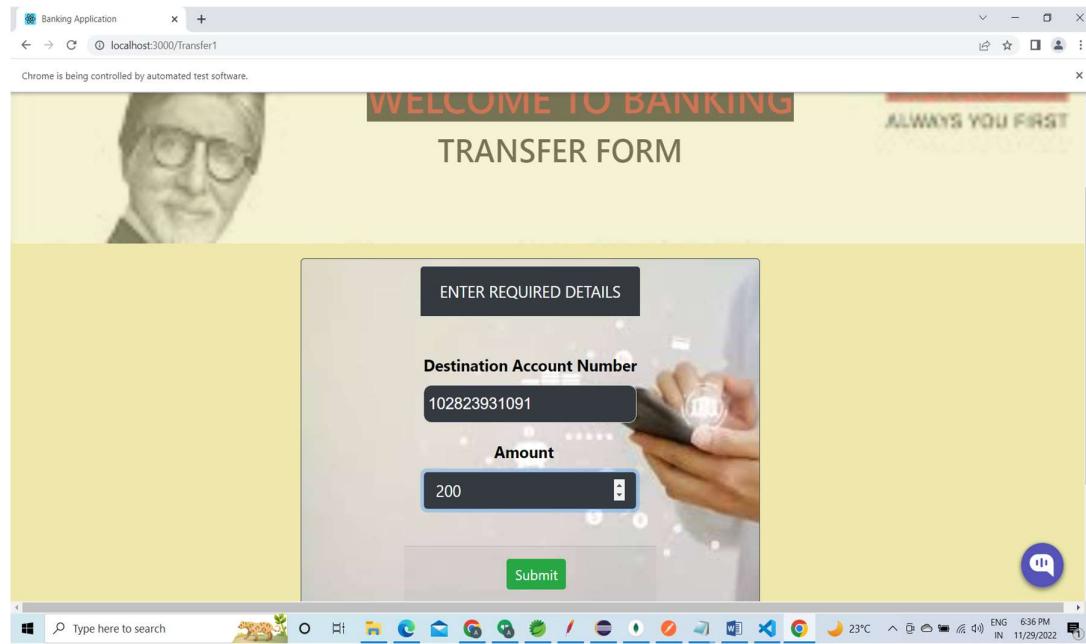


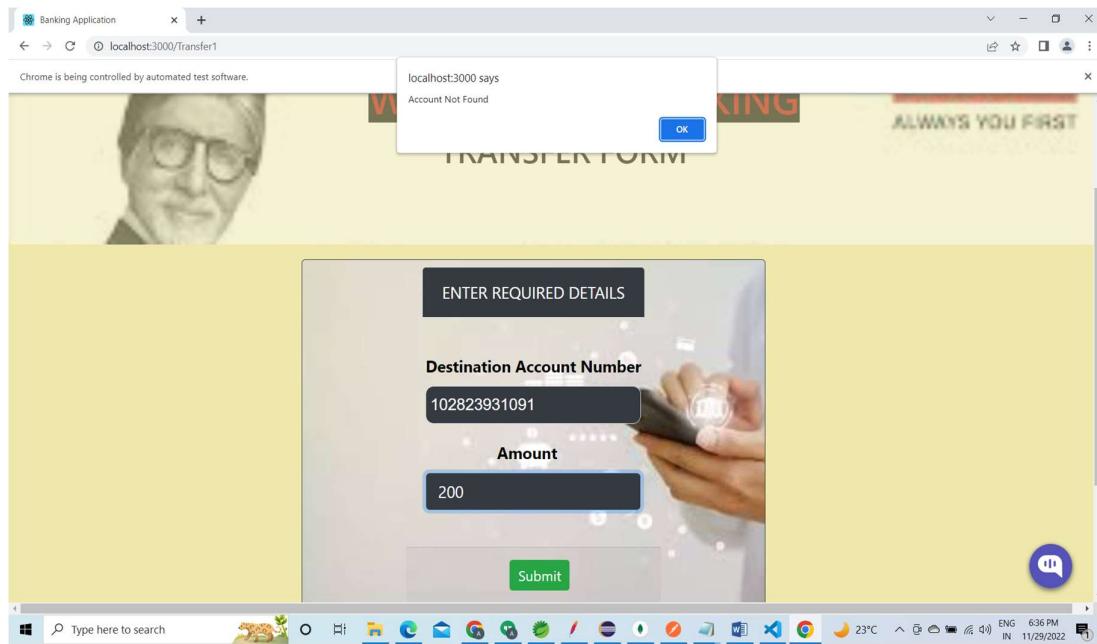
23°C ENG 6:37 PM IN 11/29/2022

- TA02 – Transfer functionality with amount more than current balance and correct destination account number



- TA03 – Transfer functionality with amount less than current balance and incorrect destination account number





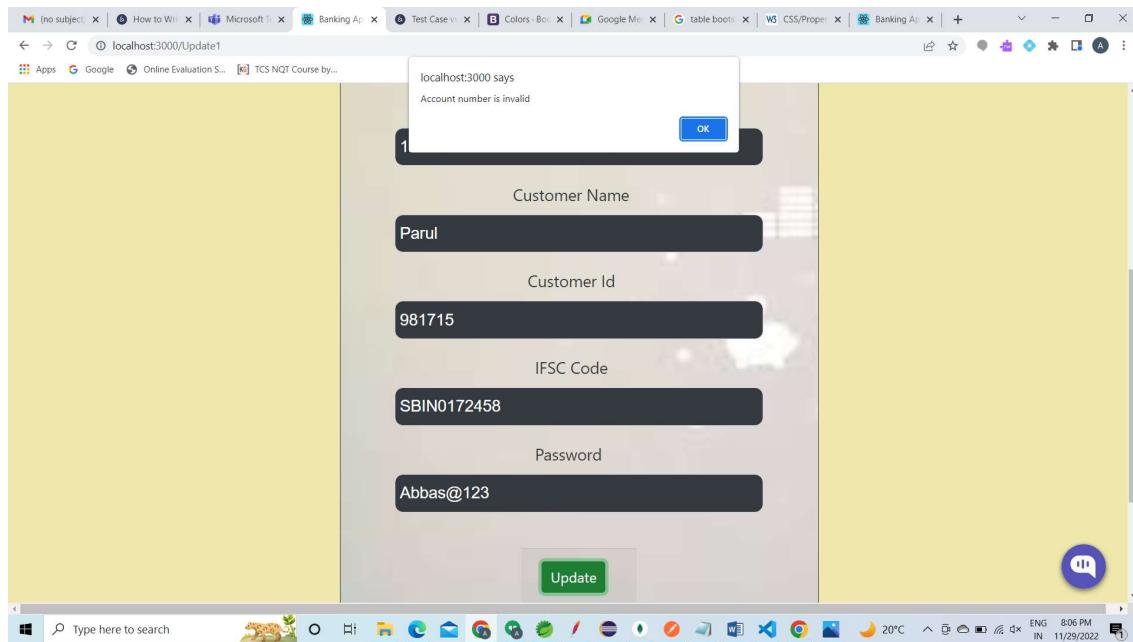
➤ **Test Scenario – Checking Update Functionality**

Navigate to Update page

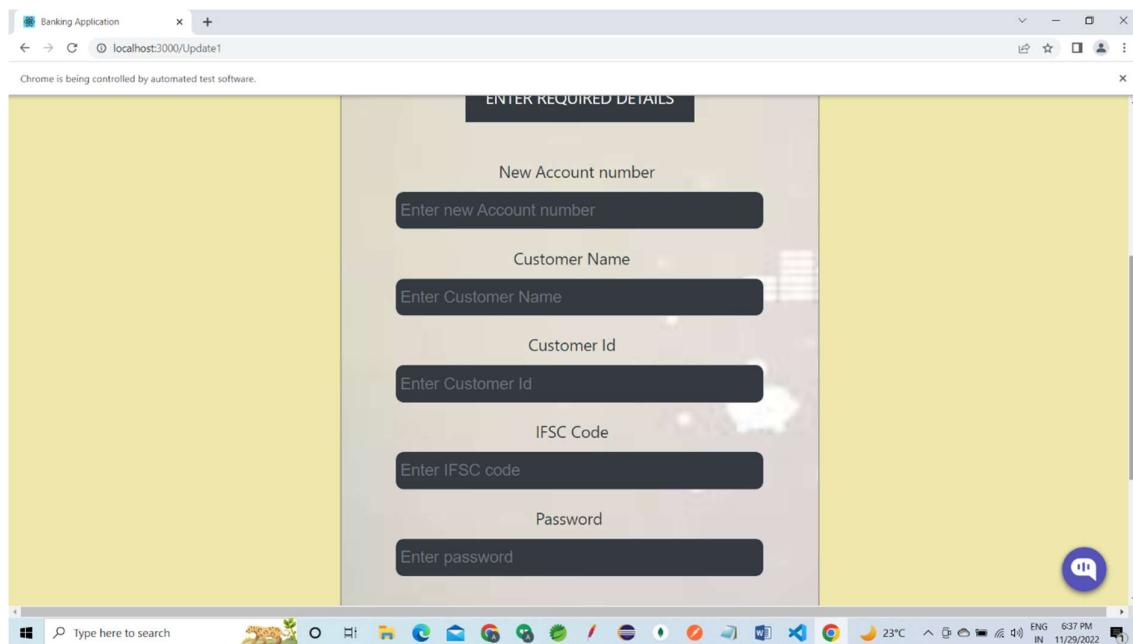
Test case id	Test case description	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
UU01	Check Update functionality with invalid account number and remaining details correctly	Enter incorrect account number and correct customer name, customer ID, IFSC code and password	Account number = 11100911245 , customer name= Parul,customer id= 9817156,IFSC code= SBIN0172458,password= Abbas@123	Details should not be updated	As expected	Pass
UU02	Check Update functionality with all valid details.	Enter correct account number, customer name, customer ID, IFSC code and password	Account number = 11100911273 , customer name= Parul,customer id= 9817156,IFSC code= SBIN0172458,password= Abbas@123	Details should be updated	As expected	Pass

OUTPUT

- **UU01** – Checking update functionality with all correct data except account number



- **UU02- Checking update functionality with all correct data .**





Banking Application × +

localhost:3000/Update1

Chrome is being controlled by automated test software.

ENTER REQUIRED DETAILS

New Account number
11100911273

Customer Name
Parul

Customer Id
Enter Customer Id

IFSC Code
Enter IFSC code

Password
Enter password

Update

Feedback icon

Type here to search

Windows taskbar icons

23°C ENG 6:37 PM IN 11/29/2022

Banking Application × +

localhost:3000/Update1

Chrome is being controlled by automated test software.

New Account number
11100911273

Customer Name
Parul

Customer Id
981715

IFSC Code
Enter IFSC code

Password
Enter password

Update

Feedback icon

Type here to search

Windows taskbar icons

23°C ENG 6:37 PM IN 11/29/2022



Banking Application × +

localhost:3000/Update1

Chrome is being controlled by automated test software.

New Account number
11100911273

Customer Name
Parul

Customer Id
981715

IFSC Code
SBIN0172458

Password
Enter password

Update

Feedback icon

Type here to search

Windows taskbar icons

Banking Application × +

localhost:3000/Update1

Chrome is being controlled by automated test software.

Customer Name
Parul

Customer Id
981715

IFSC Code
SBIN0172458

Password
Abbas@123

Update

Details updated

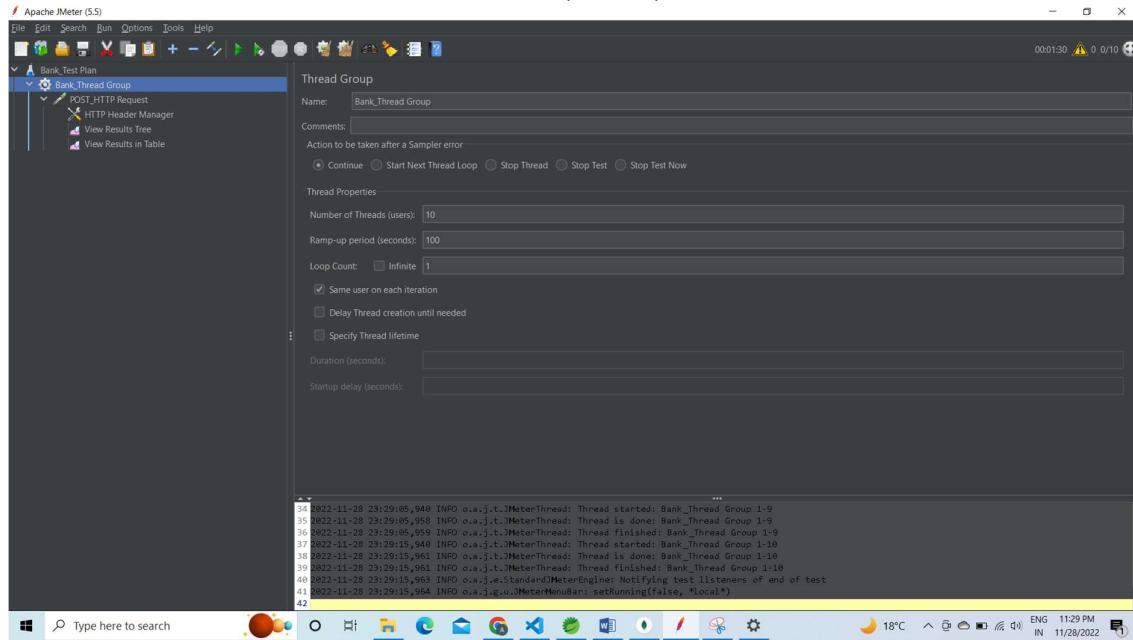
Feedback icon

Type here to search

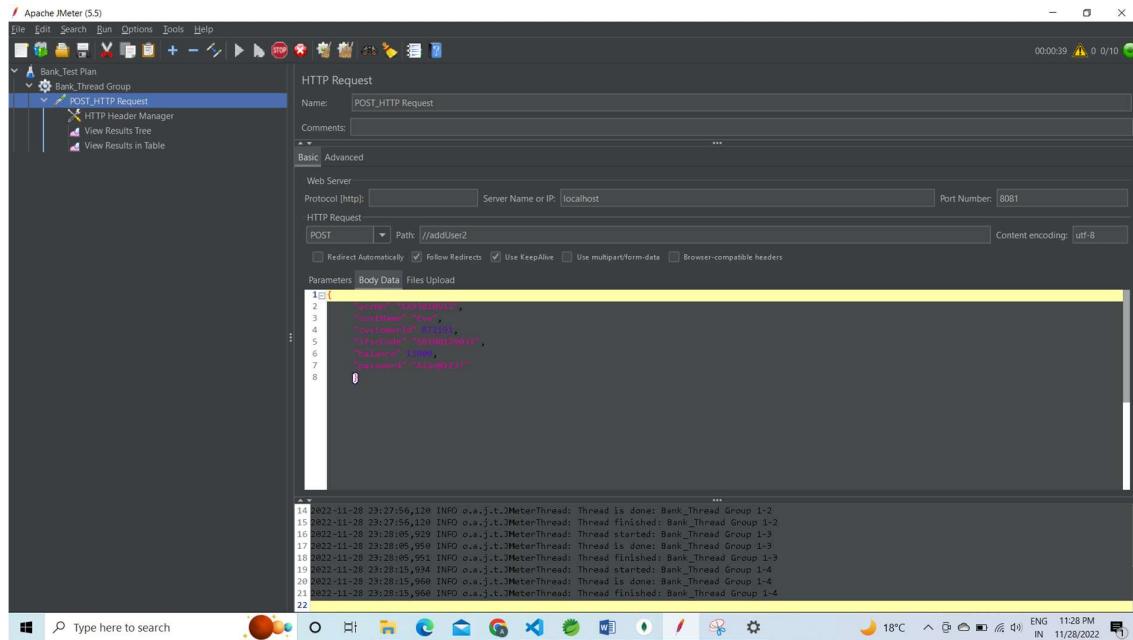
Windows taskbar icons

JUNIT TESTING FOR API

ADDING A USER OR USER SIGNUP (POST)



Thread group has properties such that 10 users should access to create new user in 100 seconds in 1 loop.



Request is POST and api is running in localhost. Post Body has been given to create new document in corresponding collection in mongoDB



Apache JMeter (5.5)

File Edit Search Run Options Tools Help

Bank Test Plan

Bank_Thread Group

POST_1 HTTP Request

HTTP Header Manager

View Results Tree

View Results in Table

View Results Tree

Name: View Results Tree

Comments:

Write results to file / Read from file

Filename: []

Search: [] Case sensitive [] Regular exp. [] Search [] Reset []

Text

Sampler result Request Response data

Response Body Response headers

["accNo":"1293810912","custName":"Eve","customerId":872191,"ifscCode":"SBIN0129031","balance":13000,"password":"Alan@123!"]

34 2022-11-28 23:29:05,948 INFO o.a.j.t.JMeterThread: Thread started: Bank_Thread Group 1-9

35 2022-11-28 23:29:05,958 INFO o.a.j.t.JMeterThread: Thread is done: Bank_Thread Group 1-9

36 2022-11-28 23:29:05,959 INFO o.a.j.t.JMeterThread: Thread finished: Bank_Thread Group 1-9

37 2022-11-28 23:29:05,960 INFO o.a.j.t.JMeterThread: Thread started: Bank_Thread Group 1-10

38 2022-11-28 23:29:05,961 INFO o.a.j.t.JMeterThread: Thread is done: Bank_Thread Group 1-10

39 2022-11-28 23:29:05,961 INFO o.a.j.t.JMeterThread: Thread finished: Bank_Thread Group 1-10

40 2022-11-28 23:29:05,961 INFO o.a.j.e.StandardJMeterEngine: Notifying test listeners of end of test

41 2022-11-28 23:29:05,964 INFO o.a.j.g.u.JMeterMenuBar: setRunning(false, *local*)

42

Windows Taskbar: Type here to search, File Explorer, Internet Explorer, Mail, Google Chrome, File, Task View, Taskbar settings, System, Taskbar, 18°C, ENG IN 11/28/2022

View Result tree listener

Apache JMeter (5.5)

File Edit Search Run Options Tools Help

Bank Test Plan

Bank_Thread Group

POST_1 HTTP Request

HTTP Header Manager

View Results Tree

View Results in Table

View Results in Table

Name: View Results in Table

Comments:

Write results to file / Read from file

Filename: []

Browse... Log/Display Only [] Errors [] Successes [] Configure []

Sample #	Start Time	Thread Name	Label	SampleTime(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(ms)
1	23:27:45.924	Bank_Thread Grou...	POST_1 HTTP Requ...	744	[green]	385	350	743	3
2	23:27:55.935	Bank_Thread Grou...	POST_1 HTTP Requ...	186	[green]	210	350	186	5
3	23:28:05.933	Bank_Thread Grou...	POST_1 HTTP Requ...	17	[green]	210	350	17	4
4	23:28:15.937	Bank_Thread Grou...	POST_1 HTTP Requ...	22	[green]	210	350	22	7
5	23:28:25.937	Bank_Thread Grou...	POST_1 HTTP Requ...	20	[green]	210	350	20	4
6	23:28:35.938	Bank_Thread Grou...	POST_1 HTTP Requ...	18	[green]	210	350	18	3
7	23:28:45.943	Bank_Thread Grou...	POST_1 HTTP Requ...	15	[green]	210	350	15	3
8	23:28:55.940	Bank_Thread Grou...	POST_1 HTTP Requ...	15	[green]	210	350	15	3
9	23:29:05.943	Bank_Thread Grou...	POST_1 HTTP Requ...	15	[green]	210	350	15	3
10	23:29:15.944	Bank_Thread Grou...	POST_1 HTTP Requ...	17	[green]	210	350	17	4

34 2022-11-28 23:29:05,948 INFO o.a.j.t.JMeterThread: Thread started: Bank_Thread Group 1-9

35 2022-11-28 23:29:05,958 INFO o.a.j.t.JMeterThread: Thread is done: Bank_Thread Group 1-9

36 2022-11-28 23:29:05,959 INFO o.a.j.t.JMeterThread: Thread finished: Bank_Thread Group 1-9

37 2022-11-28 23:29:05,960 INFO o.a.j.t.JMeterThread: Thread started: Bank_Thread Group 1-10

38 2022-11-28 23:29:05,961 INFO o.a.j.t.JMeterThread: Thread is done: Bank_Thread Group 1-10

39 2022-11-28 23:29:05,961 INFO o.a.j.t.JMeterThread: Thread finished: Bank_Thread Group 1-10

40 2022-11-28 23:29:05,961 INFO o.a.j.e.StandardJMeterEngine: Notifying test listeners of end of test

41 2022-11-28 23:29:05,964 INFO o.a.j.g.u.JMeterMenuBar: setRunning(false, *local*)

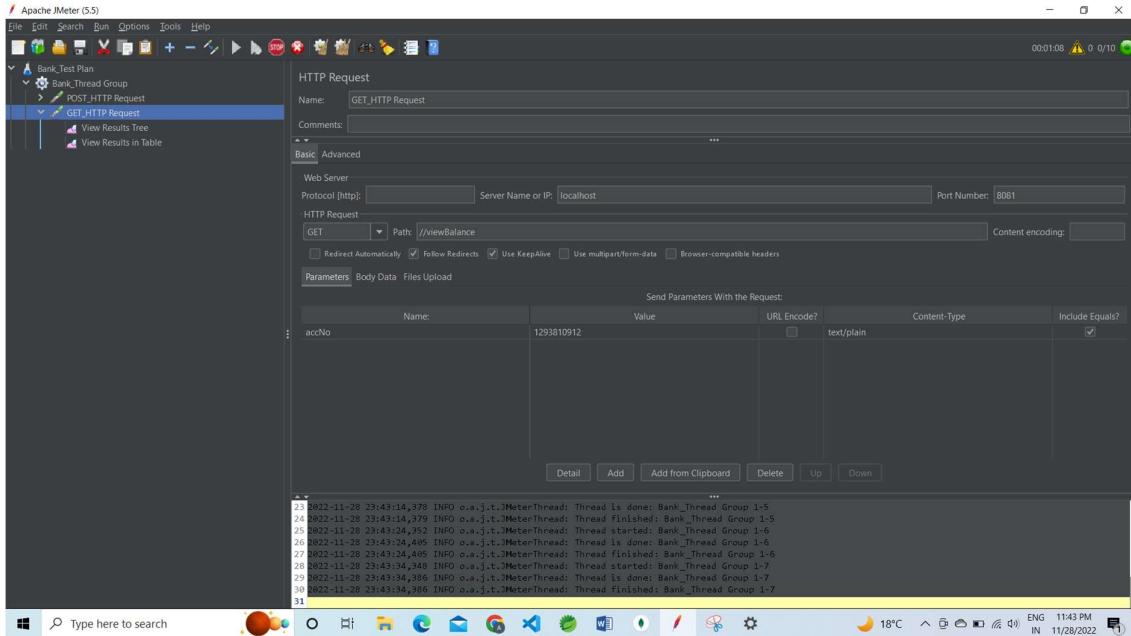
42

Windows Taskbar: Type here to search, File Explorer, Internet Explorer, Mail, Google Chrome, File, Task View, Taskbar settings, System, Taskbar, 18°C, ENG IN 11/28/2022

View results in table

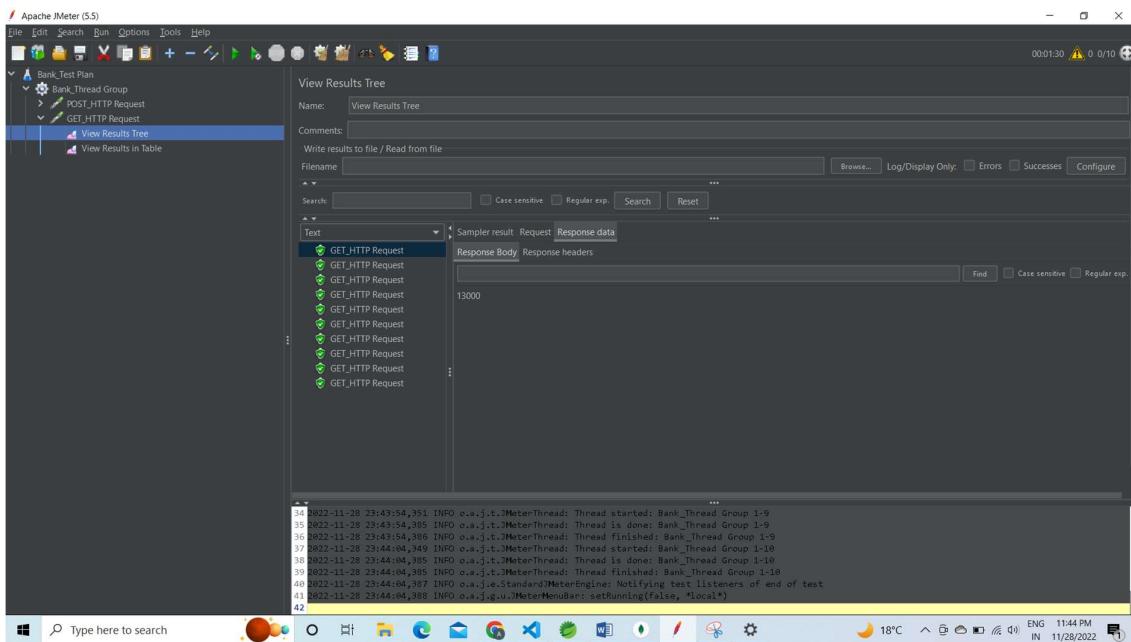
GETTING BALANCE FOR A USER (GET)

Apache JMeter (5.5)



The screenshot shows the Apache JMeter interface with a test plan named "Bank_Test Plan". It contains a "Bank_Thread Group" with two requests: "POST_HTTP Request" and "GET_HTTP Request". The "GET_HTTP Request" is selected. In the "Basic" tab of the request editor, the "Web Server" section is configured with "Protocol [http]" and "Server Name or IP: localhost" at "Port Number: 8081". The "HTTP Request" section shows a "GET" method with "Path: //viewBalance". Under "Parameters", there is a single parameter "accNo" with value "1238810912". The "Advanced" tab is collapsed. The status bar at the bottom right shows "00:01:08 ⚠ 0 0/10".

Apache JMeter (5.5)



The screenshot shows the Apache JMeter interface with the same "Bank_Test Plan" and "Bank_Thread Group" setup. The "View Results Tree" listener is selected. The results pane displays a tree view of the "GET_HTTP Request" sampler results. The "Response Body" tab is selected, showing the response content "13000". The status bar at the bottom right shows "00:01:30 ⚠ 0 0/10".

View results tree



Apache JMeter (5.5)

File Edit Search Run Options Tools Help

Bank Test Plan

View Results in Table

Name: View Results in Table

Comments:

Write results to file / Read from file

Filename: Browse... Log/Display Only Errors Successes Configure

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(ms)
1	2342:34.360	Bank_Thread Group	GET_1	19	Success	260	146	19	0
2	2342:44.389	Bank_Thread Group	GET_1	9	Success	260	146	9	0
3	2342:44.366	Bank_Thread Group	GET_1	10	Success	260	146	10	0
4	2343:04.371	Bank_Thread Group	GET_1	12	Success	260	146	12	0
5	2343:14.367	Bank_Thread Group	GET_1	10	Success	260	146	10	0
6	2343:24.395	Bank_Thread Group	GET_1	11	Success	260	146	11	0
7	2343:34.374	Bank_Thread Group	GET_1	11	Success	260	146	11	0
8	2343:44.377	Bank_Thread Group	GET_1	15	Success	260	146	15	0
9	2343:54.376	Bank_Thread Group	GET_1	10	Success	260	146	10	0
10	2344:04.372	Bank_Thread Group	GET_1	12	Success	260	146	12	0

```
34 2022-11-28 23:43:54,351 INFO o.a.j.t.JMeterThread: Thread started: Bank_Thread Group 1-9
35 2022-11-28 23:43:54,385 INFO o.a.j.t.JMeterThread: Thread is done: Bank_Thread Group 1-9
36 2022-11-28 23:43:54,386 INFO o.a.j.t.JMeterThread: Thread finished: Bank_Thread Group 1-9
37 2022-11-28 23:44:04,385 INFO o.a.j.t.JMeterThread: Thread is done: Bank_Thread Group 1-10
38 2022-11-28 23:44:04,385 INFO o.a.j.t.JMeterThread: Thread finished: Bank_Thread Group 1-10
39 2022-11-28 23:44:04,385 INFO o.a.j.t.JMeterThread: Thread is done: Bank_Thread Group 1-18
40 2022-11-28 23:44:04,387 INFO o.a.j.e.StandardJMeterEngine: Notifying test listeners of end of test
41 2022-11-28 23:44:04,388 INFO o.a.j.g.u.JMeterMenuBar: setRunning(false, *local*)
42
```

Type here to search 18°C ENG 11:45 PM IN 11/28/2022

View results in table

UPDATING USER DETAILS (PUT)

Apache JMeter (5.5)

File Edit Search Run Options Tools Help

Bank Test Plan

PUT_1

HTTP Request

Name: PUT_1

Comments:

Basic Advanced

Web Server

Protocol [http]: Server Name or IP: localhost Port Number: 8081

HTTP Request

PUT Path: //Update1/1293810912 Content encoding: UTF-8

Redirect Automatically Follow Redirects Use KeepAlive Use multipart/form-data Browser-compatible headers

Parameters Body Data Files Upload

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
```

```
2022-11-29 00:08:49,572 INFO o.a.j.t.JMeterThread: Thread is done: Bank_Thread Group 1-4
21 2022-11-29 00:08:49,573 INFO o.a.j.t.JMeterThread: Thread finished: Bank_Thread Group 1-5
22 2022-11-29 00:08:49,573 INFO o.a.j.t.JMeterThread: Thread is done: Bank_Thread Group 1-5
23 2022-11-29 00:08:49,572 INFO o.a.j.t.JMeterThread: Thread is done: Bank_Thread Group 1-5
24 2022-11-29 00:08:49,573 INFO o.a.j.t.JMeterThread: Thread finished: Bank_Thread Group 1-5
25 2022-11-29 00:09:09,518 INFO o.a.j.t.JMeterThread: Thread started: Bank_Thread Group 1-6
26 2022-11-29 00:09:09,573 INFO o.a.j.t.JMeterThread: Thread is done: Bank_Thread Group 1-6
27 2022-11-29 00:09:09,573 INFO o.a.j.t.JMeterThread: Thread finished: Bank_Thread Group 1-6
28
```

Type here to search 18°C ENG 12:09 AM IN 11/29/2022

Only updating customer name from Eve to Vesper in account number – 1293810912



Apache JMeter (5.5)

File Edit Search Run Options Tools Help

Bank Test Plan

- Bank_Thread Group
 - POST, HTTP Request
 - GET, HTTP Request
 - PUT, HTTP Request
 - HTTP Header Manager
- View Results Tree
- View Results in Table

Name: View Results Tree

Comments:

Write results to file / Read from file

Filename: []

Search: [] Case sensitive Regular exp. Search Reset

Text Sampler result Request Response data

Response Body Response headers

["accNo":"129810912","custName":"Vesper","customerId":872191,"ifscCode":"SBIN0129031","balance":13000,"password":"Alan@123"}]

34 0822-11-29 00:09:39,513 INFO o.a.j.t.JMeterThread: Thread started: Bank_Thread Group 1-9
35 0822-11-29 00:09:39,569 INFO o.a.j.t.JMeterThread: Thread is done: Bank_Thread Group 1-9
36 0822-11-29 00:09:39,569 INFO o.a.j.t.JMeterThread: Thread finished: Bank_Thread Group 1-9
37 0822-11-29 00:09:49,512 INFO o.a.j.t.JMeterThread: Thread started: Bank_Thread Group 1-10
38 0822-11-29 00:09:49,582 INFO o.a.j.t.JMeterThread: Thread is done: Bank_Thread Group 1-10
39 0822-11-29 00:09:49,582 INFO o.a.j.t.JMeterThread: Thread finished: Bank_Thread Group 1-10
40 0822-11-29 00:09:49,583 INFO o.a.j.e.StandardJMeterEngine: Notifying test listeners of end of test
41 0822-11-29 00:09:49,583 INFO o.a.j.g.u.JMeterMenuBar: setRunning(false, *Local*)

42

Type here to search 18°C ENG 12:11 AM IN 11/29/2022

Apache JMeter (5.5)

File Edit Search Run Options Tools Help

Bank Test Plan

- Bank_Thread Group
 - PUT, HTTP Request
 - GET, HTTP Request
 - PUT, HTTP Request
- View Results in Table

Name: View Results in Table

Comments:

Write results to file / Read from file

Filename: []

Browse... Log/Display Only Errors Successes Configure

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(ms)
1	00:08:19,541	Bank_Thread Group 1-1	PUT, HTTP Request	28	green	388	362	28	0
2	00:08:29,564	Bank_Thread Group 1-2	PUT, HTTP Request	20	green	388	362	20	0
3	00:08:39,567	Bank_Thread Group 1-3	PUT, HTTP Request	18	green	388	362	17	0
4	00:08:49,557	Bank_Thread Group 1-4	PUT, HTTP Request	16	green	388	362	16	0
5	00:08:59,550	Bank_Thread Group 1-5	PUT, HTTP Request	22	green	388	362	22	0
6	00:09:09,554	Bank_Thread Group 1-6	PUT, HTTP Request	19	green	388	362	19	0
7	00:09:19,549	Bank_Thread Group 1-7	PUT, HTTP Request	30	green	388	362	29	0
8	00:09:29,531	Bank_Thread Group 1-8	PUT, HTTP Request	12	green	388	362	11	0
9	00:09:39,550	Bank_Thread Group 1-9	PUT, HTTP Request	18	green	388	362	18	0
10	00:09:49,557	Bank_Thread Group 1-10	PUT, HTTP Request	24	green	388	362	24	0

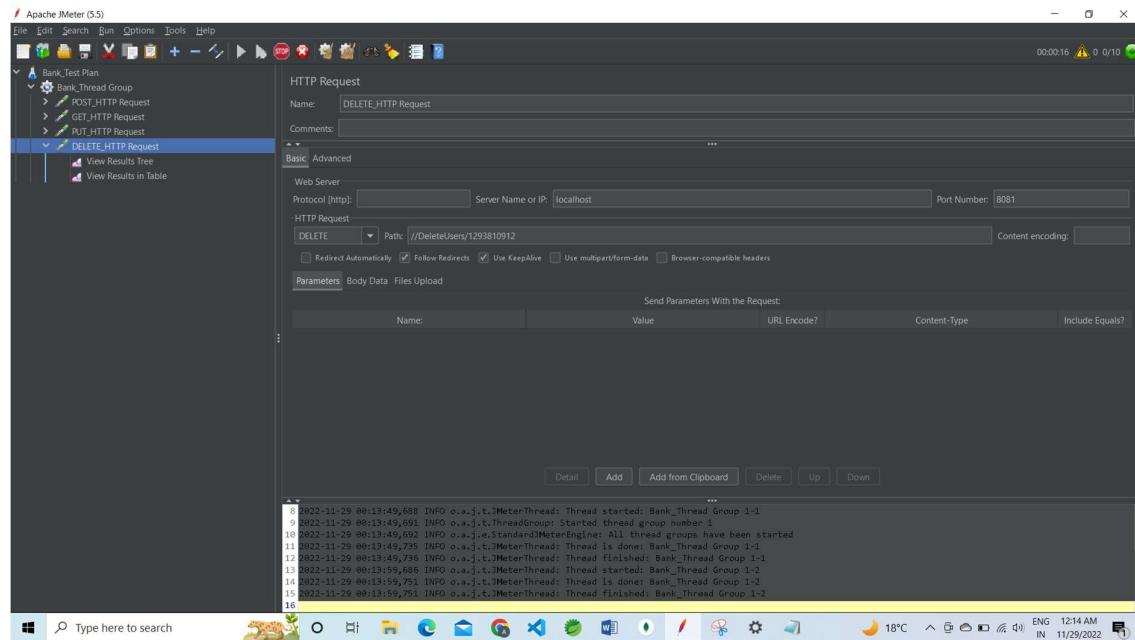
34 0822-11-29 00:09:39,513 INFO o.a.j.t.JMeterThread: Thread started: Bank_Thread Group 1-9
35 0822-11-29 00:09:39,569 INFO o.a.j.t.JMeterThread: Thread is done: Bank_Thread Group 1-9
36 0822-11-29 00:09:39,569 INFO o.a.j.t.JMeterThread: Thread finished: Bank_Thread Group 1-9
37 0822-11-29 00:09:49,512 INFO o.a.j.t.JMeterThread: Thread started: Bank_Thread Group 1-10
38 0822-11-29 00:09:49,582 INFO o.a.j.t.JMeterThread: Thread is done: Bank_Thread Group 1-10
39 0822-11-29 00:09:49,582 INFO o.a.j.t.JMeterThread: Thread finished: Bank_Thread Group 1-10
40 0822-11-29 00:09:49,583 INFO o.a.j.e.StandardJMeterEngine: Notifying test listeners of end of test
41 0822-11-29 00:09:49,583 INFO o.a.j.g.u.JMeterMenuBar: setRunning(false, *Local*)

42

Type here to search 18°C ENG 12:11 AM IN 11/29/2022

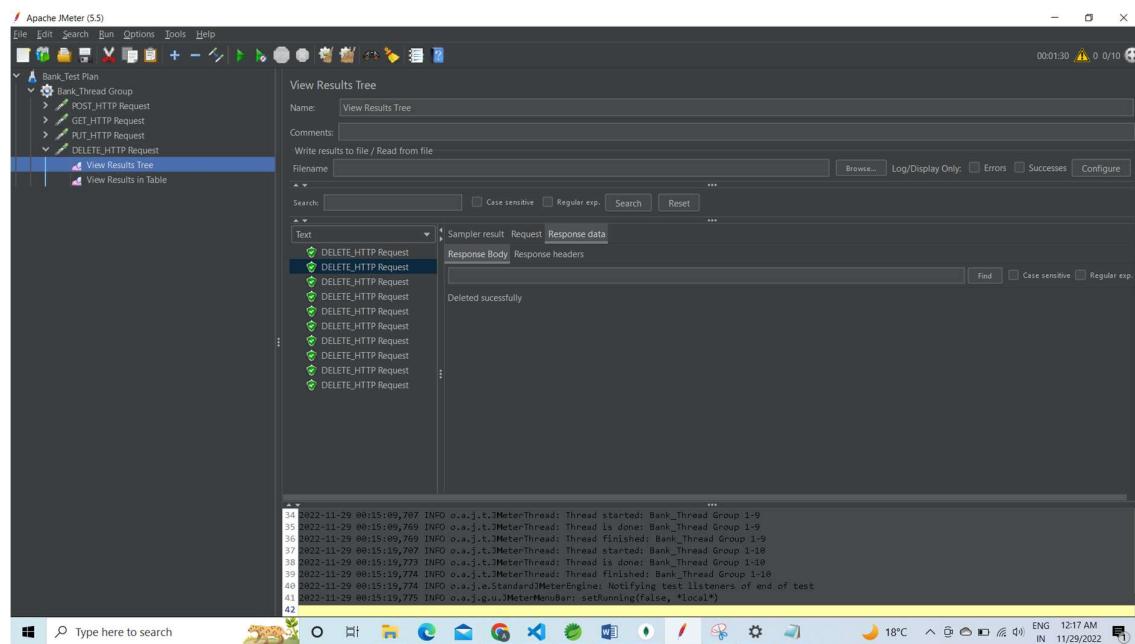
DELETING USER

Apache JMeter (5.5)



The screenshot shows the Apache JMeter interface with a test plan named "Bank_Test_Plan". A "DELETE_HTTP Request" sampler is selected in the tree view. The "Basic" tab of the request configuration panel is visible, showing the protocol as "DELETE", the path as "/DeleteUsers/1239810912", and various request headers and parameters. The "View Results Tree" button is highlighted in the tree view.

Apache JMeter (5.5)



The screenshot shows the "View Results Tree" panel of the Apache JMeter interface. It displays a tree of sampler results for the "DELETE_HTTP Request" samplers. The response body for one of the samplers shows the message "Deleted sucessfully". The "Log/Display Only" checkbox is checked in the panel's toolbar. The status bar at the bottom indicates "00:01:30" and "0/0/10".

Apache JMeter (5.5)

File Edit Search Run Options Tools Help

Bank Test Plan

View Results in Table

Name: View Results in Table

Comments:

Write results to file / Read from file

Filename: Browse... Log/Display Only Errors Successes Configure

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(ms)
1	00:13:49.729	Bank_Thread Group	DELETE_HTTP Req.	6	275	226	226	6	0
2	00:13:59.744	Bank_Thread Group	DELETE_HTTP Req.	7	275	226	226	7	0
3	00:14:09.714	Bank_Thread Group	DELETE_HTTP Req.	5	275	226	226	5	0
4	00:14:19.767	Bank_Thread Group	DELETE_HTTP Req.	12	275	226	226	12	0
5	00:14:29.752	Bank_Thread Group	DELETE_HTTP Req.	12	275	226	226	11	0
6	00:14:39.752	Bank_Thread Group	DELETE_HTTP Req.	6	275	226	226	6	0
7	00:14:49.770	Bank_Thread Group	DELETE_HTTP Req.	12	275	226	226	12	0
8	00:14:59.769	Bank_Thread Group	DELETE_HTTP Req.	11	275	226	226	11	0
9	00:15:09.762	Bank_Thread Group	DELETE_HTTP Req.	7	275	226	226	7	0
10	00:15:19.767	Bank_Thread Group	DELETE_HTTP Req.	6	275	226	226	6	0

...

```

34 2022-11-29 00:15:09.787 INFO o.a.j.t.MeterThread: Thread started: Bank_Thread Group 1-9
35 2022-11-29 00:15:09.769 INFO o.a.j.t.MeterThread: Thread is done: Bank_Thread Group 1-9
36 2022-11-29 00:15:09.769 INFO o.a.j.t.MeterThread: Thread finished: Bank_Thread Group 1-9
37 2022-11-29 00:15:09.769 INFO o.a.j.t.MeterThread: Thread is done: Bank_Thread Group 1-10
38 2022-11-29 00:15:19.773 INFO o.a.j.t.MeterThread: Thread is done: Bank_Thread Group 1-10
39 2022-11-29 00:15:19.774 INFO o.a.j.e.StandardMeterEngine: Thread finished: Bank_Thread Group 1-10
40 2022-11-29 00:15:19.774 INFO o.a.j.g.u.StandardMeterEngine: Notifying test listeners of end of test
41 2022-11-29 00:15:19.775 INFO o.a.j.g.u.StandardMeterEngine: setRunning(false, *local*)
42

```

Windows Taskbar: Type here to search, File Explorer, Task View, Start, Internet Explorer, Mail, Google Chrome, File Manager, Java, System, Control Panel, Task Manager, 18°C, ENG, 12:17 AM, IN, 11/29/2022

CONCLUSION

This banking application project is successfully built and tested for users and admin to achieve all the basic bank operations, thereby automating the whole money managing process within a bank. Many concepts I got to explore including react, spring boot, axios, Mongo DB, and RESTful APIs which helped me build this amazing Banking project.