

Full Stack Development with MERN

Frontend Development Report

Date	10-07-2024
Team ID	SWTID1720106020
Project Name	Banking Management
Maximum Marks	10

Project Title: Banking Management

Date: 10-07-2024

Prepared by: Kopparapu Shashank, Rangaiah Gari Harshavardhan Reddy, Tirumala Mukesh Goud and Nimmala Sai Lahari

Objective

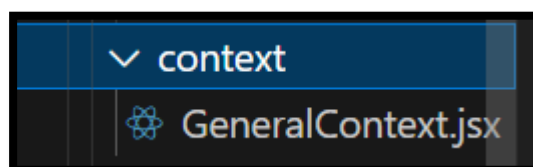
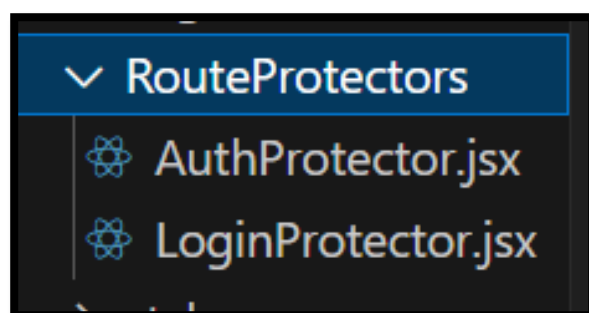
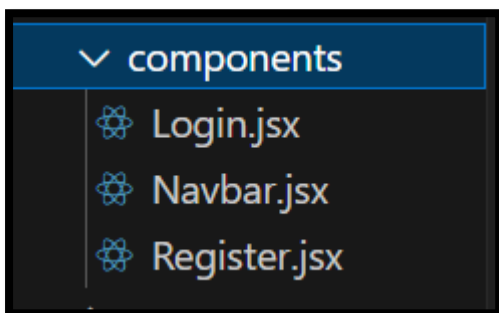
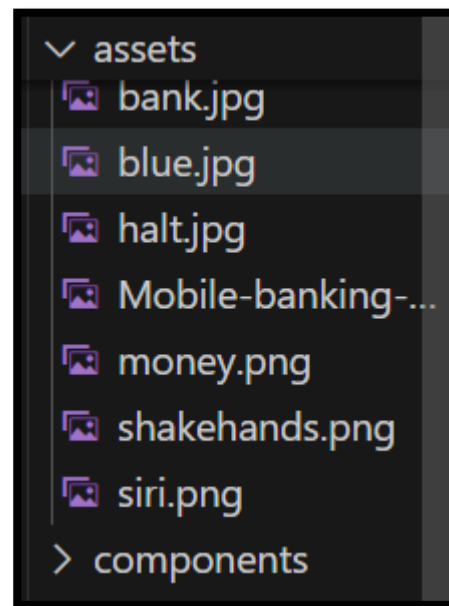
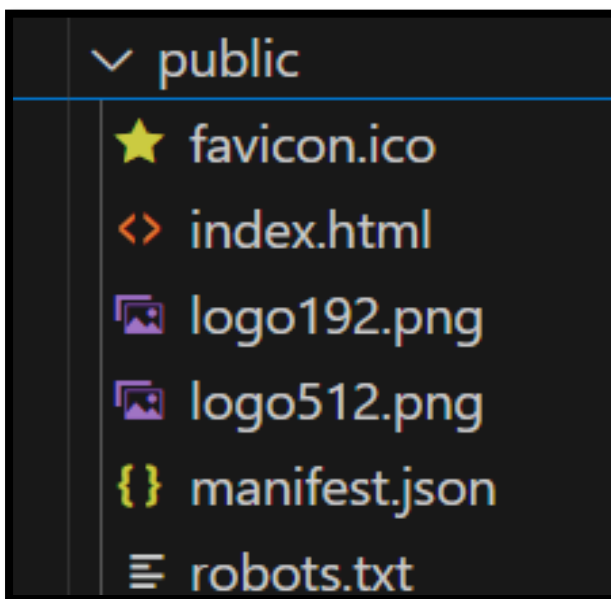
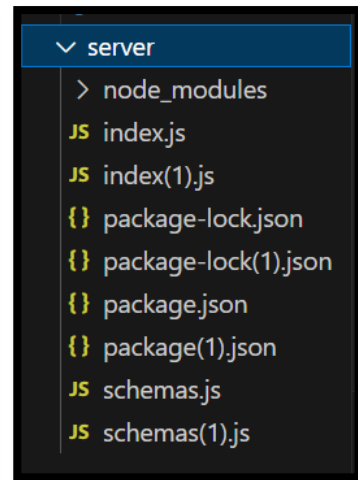
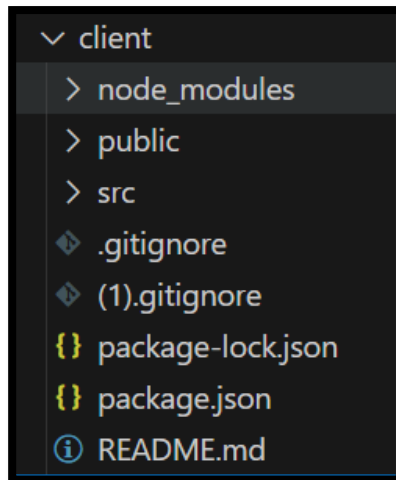
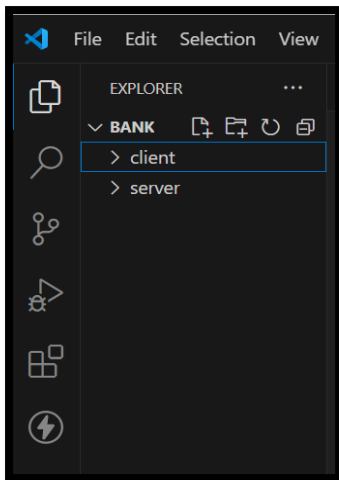
This report's goal is to record the frontend development process and important facets of the Banking Management project's user interface implementation.

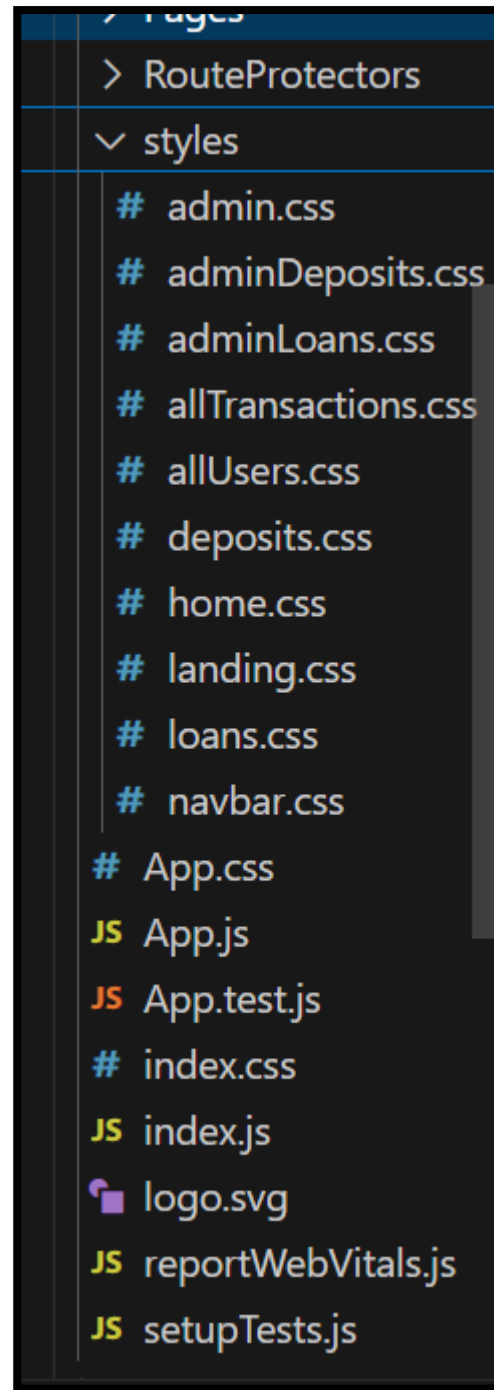
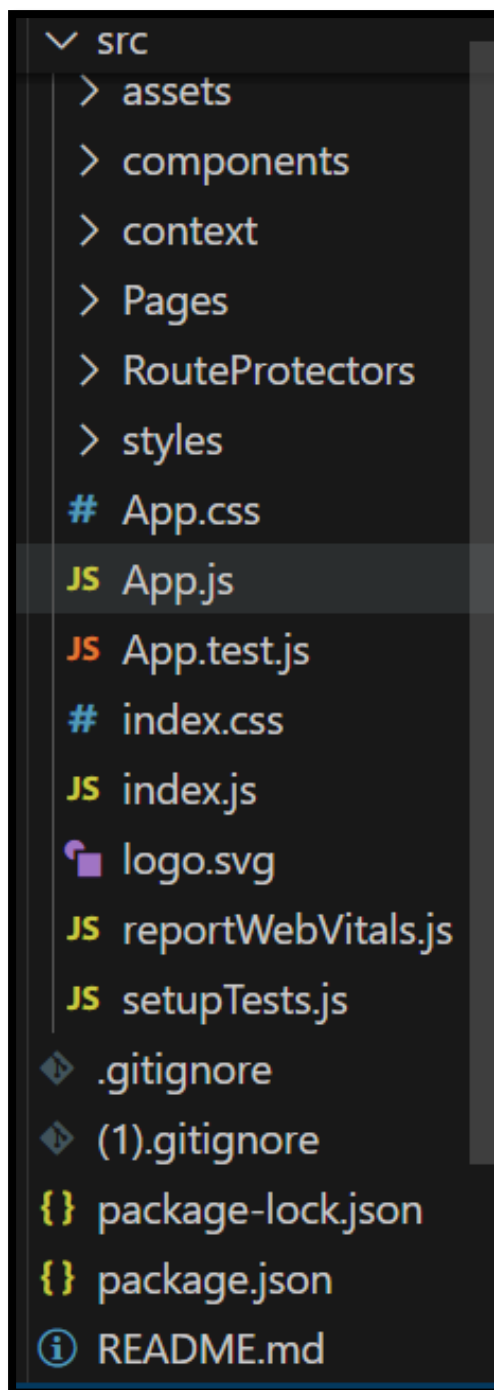
Technologies Used

- **Frontend Framework:** React.js
- **State Management:** Redux/Context API
- **UI Framework/Libraries:** Material-UI, Bootstrap
- **API Libraries:** Axios

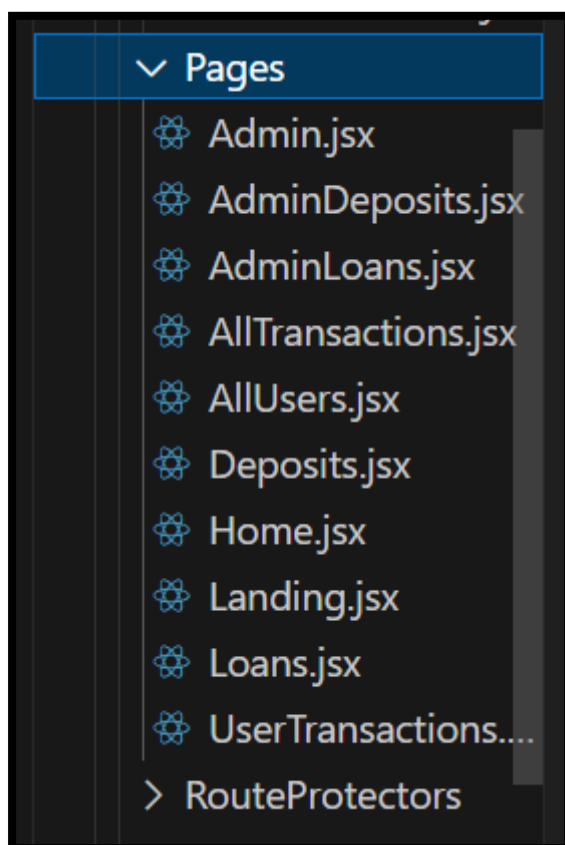
Project Structure:

Give Screenshot of Frontend Project with explanation

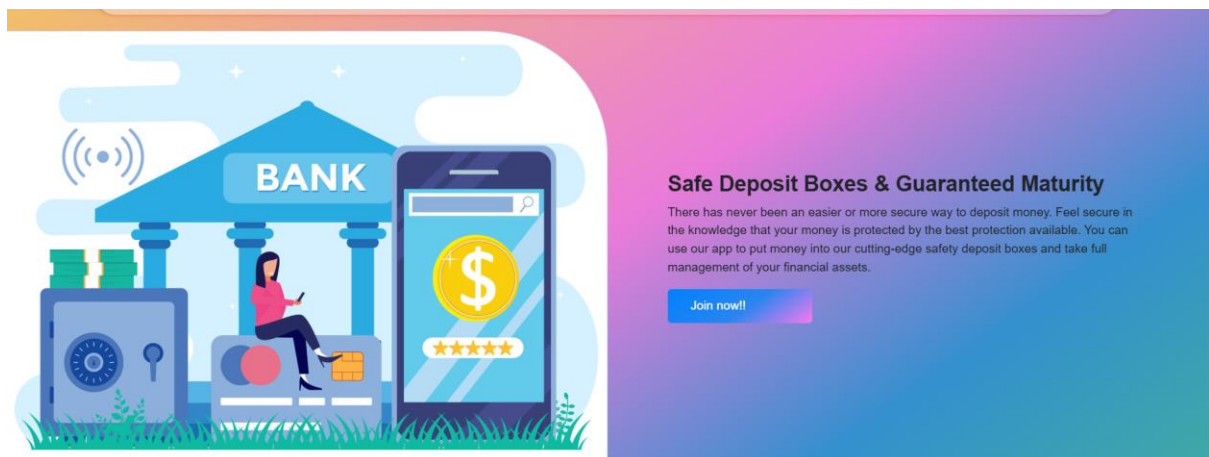
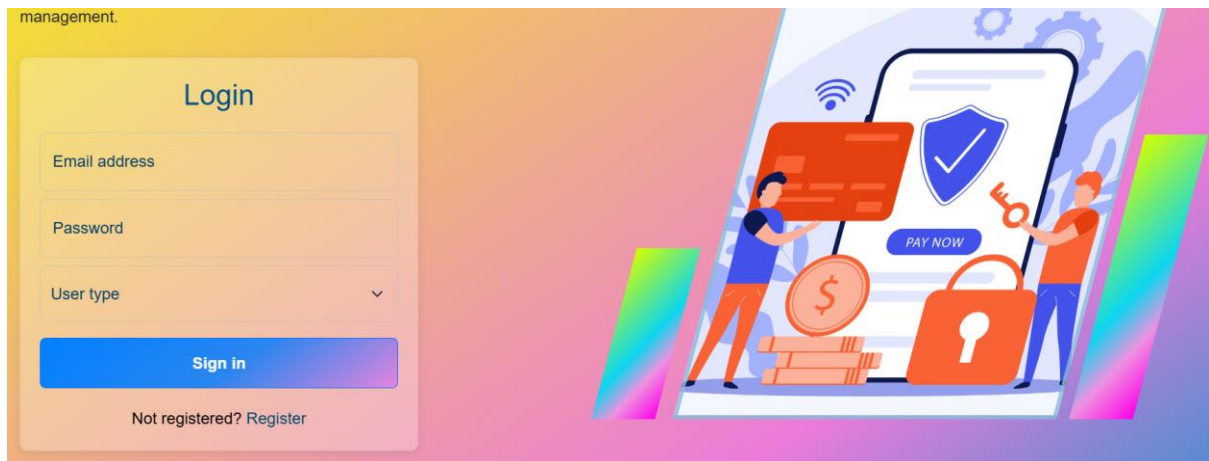
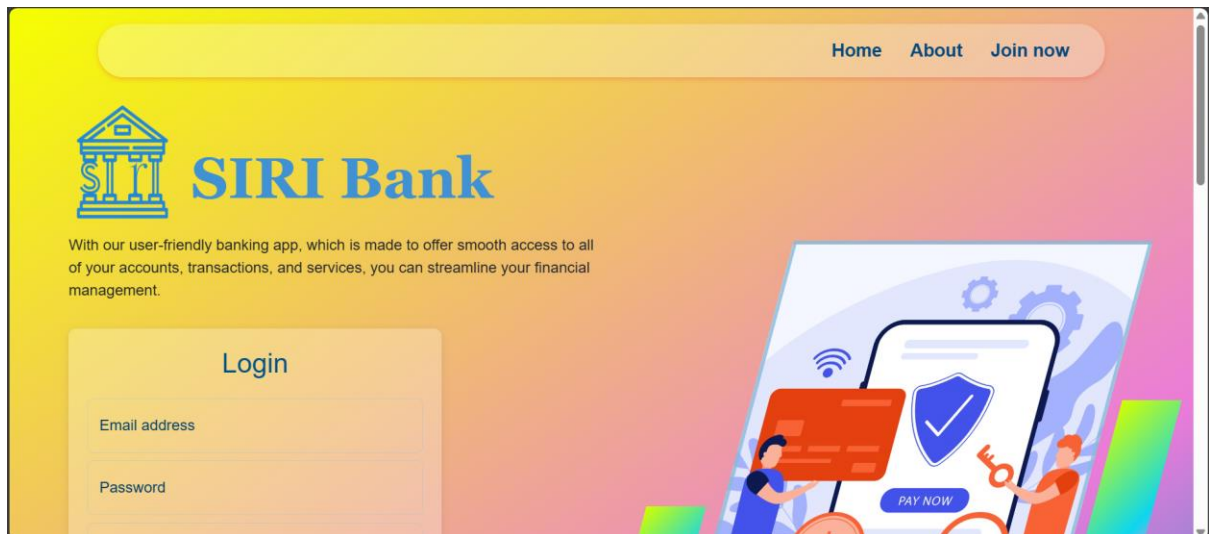




OUR LOGO



Landing Page:



Reaching your financial objectives and opening up new opportunities. LoanLift offers a variety of customised loan options to meet your demands, whether you want to launch a business, go back to school, or remodel your house. Our easy-to-use application procedure makes getting a loan simple, and our reasonable interest rates guarantee timely repayments.

All rights reserved - © Bank.com

- Responsible for routing and main application layout.

```

client > src > JS App.js > App
1  import './App.css';
2  import { Route, Routes } from 'react-router-dom';
3  import Landing from './Pages/Landing.jsx';
4  import Home from './Pages/Home';
5  import Admin from './Pages/Admin';
6  import Deposits from './Pages/Deposits';
7  import Loans from './Pages/Loans';
8  import AdminDeposits from './Pages/AdminDeposits';
9  import AdminLoans from './Pages/AdminLoans';
10 import AllTransactions from './Pages/AllTransactions';
11 import AllUsers from './Pages/AllUsers';
12 import LoginProtector from './RouteProtectors/AuthProtector';
13 import AuthProtector from './RouteProtectors/LoginProtector';
14 import UserTransactions from './Pages/UserTransactions';
15
16 function App() {
17   return (
18     <div className="App">
19
20       <Routes>
21         <Route exact path="/" element={<LoginProtector> <Landing /> </LoginProtector> } />
22         <Route path="/home" element={<AuthProtector><Home /></AuthProtector>} />
23         <Route path="/deposits" element={ <AuthProtector><Deposits /></AuthProtector>} />
24         <Route path="/loans" element={ <AuthProtector><Loans /></AuthProtector>} />
25         <Route path="/transactions" element={<AuthProtector><UserTransactions /></AuthProtector>} />
26
27         <Route path="/admin" element={ <AuthProtector><Admin /></AuthProtector>} />
28         <Route path="/all-users" element={ <AuthProtector><AllUsers /></AuthProtector>} />
29         <Route path="/all-deposits" element={ <AuthProtector><AdminDeposits /></AuthProtector>} />
30         <Route path="/all-loans" element={ <AuthProtector><AdminLoans /></AuthProtector>} />
31         <Route path="/all-transactions" element={<AuthProtector><AllTransactions /></AuthProtector>} />
32       </Routes>
33     </div>
34   );
35 }
36
37 export default App;
38

```

2.components

- Contains reusable UI components used across the application.
- Login.jsx

Login

[Not registered? Register](#)

```

client > src > components > Login.jsx > Login
1
2 import React, { useContext } from 'react'
3 import { GeneralContext } from '../context/GeneralContext';
4
5 const Login = ({setIsLoginBox}) => {
6
7   const {setEmail, setPassword, login, setUserType} = useContext(GeneralContext);
8
9   const handleLogin = async (e) =>{
10     e.preventDefault();
11     await login();
12   }
13   return (
14     <form className="authForm">
15       <h2>Login</h2>
16       <div className="form-floating mb-3 authFormInputs">
17         <input type="email" className="form-control" id="floatingInput" placeholder="name@example.com"
18           onChange={(e) => setEmail(e.target.value)} />
19         <label htmlFor="floatingInput">Email address</label>
20       </div>
21       <div className="form-floating mb-3 authFormInputs">
22         <input type="password" className="form-control" id="floatingPassword" placeholder="Password"
23           onChange={(e) => setPassword(e.target.value)} />
24         <label htmlFor="floatingPassword">Password</label>
25       </div>
26       <select className="form-select form-select-lg mb-3" aria-label=".form-select-lg example"
27         onChange={(e) => setUserType(e.target.value)}>
28         <option value="">User type</option>
29         <option value="admin">Admin</option>
30         <option value="customer">Customer</option>
31       </select>
32       <button type="submit" className="btn btn-primary" onClick={handleLogin}>Sign in</button>
33
34       <p>Not registered? <span onClick={() => setIsLoginBox(false)}>Register</span></p>
35     </form>
36   )
37 }
38 export default Login

```

- Navbar.jsx



client > src > components > Navbar.jsx > Navbar

```
1 import React, { useContext } from 'react'
2 import '../styles/navbar.css';
3 import { useNavigate } from 'react-router-dom';
4 import About3 from '../assets/siri.png';
5 import { GeneralContext } from '../context/GeneralContext';
6
7 const Navbar = () => {
8
9     const navigate = useNavigate();
10    const usertype = localStorage.getItem('userType');
11
12    const {logout} = useContext(GeneralContext);
13
14    return (
15        <>
16            <div className="navbar">
17
18                {usertype === 'customer' ?
19
20                    <>
21                        <div className='ban'>
22                            <img src={About3} className='siriii' alt="" />
23                            <h2 className='title'>SIRI Bank</h2>
24                        </div>
25
26                        <div className="nav-options" >
27
28                            <p onClick={()=>navigate('/home')}><b>Home</b></p>
29                            <p onClick={()=>navigate('/deposits')}><b>Deposits</b></p>
30                            <p onClick={()=>navigate('/loans')}><b>Loans</b></p>
31                            <p onClick={()=>navigate('/transactions')}><b>Transactions</b></p>
32                            <p onClick={logout}><b>Logout</b></p>
33
34                        </div>
35                    </>
36                : usertype === 'admin' ?
37
38                    <>
39                        <div className='bann'>
40                            <img src={About3} className='siriii' alt="" />
41                            <h2 className='titlee'>SIRI Bank_Admin</h2>
42                        </div>
43                        { /* <h3 ><b>SIRI Bank</b> (Admin)</h3> */ }
44                        <div className="nav-options" >
45
46                            <p onClick={()=>navigate('/admin')}>Home</p>
47                            <p onClick={()=>navigate('/all-users')}>Users</p>
48                            <p onClick={()=>navigate('/all-deposits')}>Deposits</p>
49                            <p onClick={()=>navigate('/all-loans')}>Loans</p>
50                            <p onClick={()=>navigate('/all-transactions')}>Transactions</p>
51                            <p onClick={logout}>Logout</p>
52
53                        </div>
54                    </>
55                :
56
57                ""
58
59            </div>
60        </>
61    )
62 }
```

- Register.jsx

Register

Username

Email address

Password

User type ▼

Sign up

Already registered? Login

```

client > src > components > Register.jsx > Register > handleRegister
1  import React, { useContext } from 'react'
2  import { GeneralContext } from '../context/GeneralContext';
3
4  const Register = ({setIsLoginBox}) => {
5
6    const {setUsername, setEmail, setPassword, usertype, setUserType, register, setHomeBranch} = useContext(GeneralContext);
7
8    const handleRegister = async (e) =>{
9      e.preventDefault();
10     await register()
11   }
12   return (
13     <form className="authForm">
14       <h2>Register</h2>
15       <div className="form-floating mb-3 authFormInputs">
16         <input type="text" className="form-control" id="floatingInput" placeholder="username"
17           onChange={(e) => setUsername(e.target.value)} />
18         <label htmlFor="floatingInput">Username</label>
19       </div>
20       <div className="form-floating mb-3 authFormInputs">
21         <input type="email" className="form-control" id="floatingEmail" placeholder="name@example.com"
22           onChange={(e) => setEmail(e.target.value)} />
23         <label htmlFor="floatingInput">Email address</label>
24       </div>
25       <div className="form-floating mb-3 authFormInputs">
26         <input type="password" className="form-control" id="floatingPassword" placeholder="Password"
27           onChange={(e) => setPassword(e.target.value)} />
28         <label htmlFor="floatingPassword">Password</label>
29       </div>
30       <select className="form-select form-select-lg mb-3 aria-label=".form-select-lg example"
31         onChange={(e) => setUserType(e.target.value)}>
32         <option value="">User type</option>

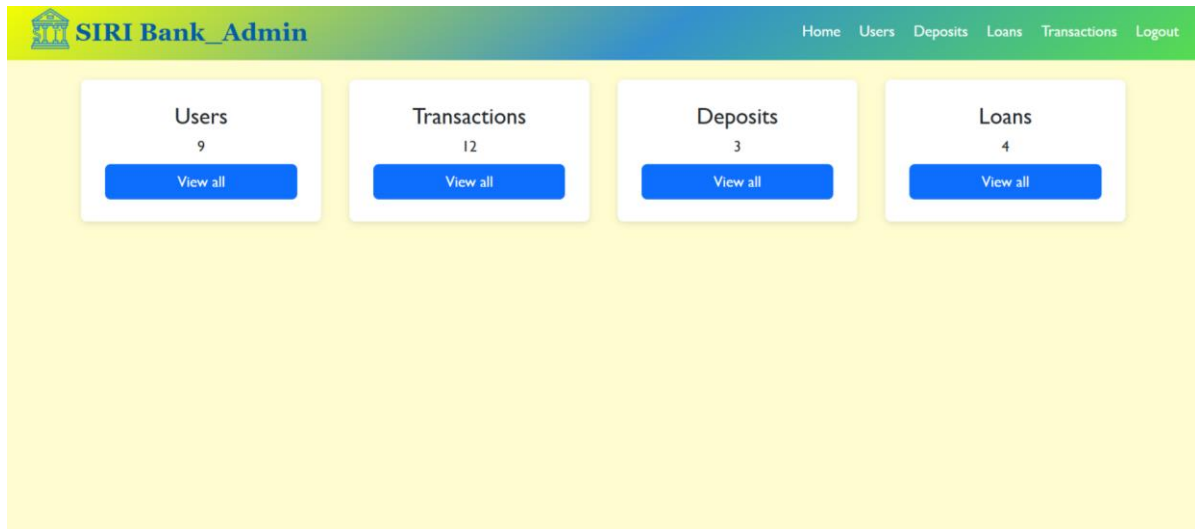
```

3.pages

- Includes different pages for Web App.

Admin Page:


- Admin.jsx



client > src > Pages > Admin.jsx > ...

```
1  import React, { useEffect, useState } from 'react'
2  import Navbar from '../components/Navbar'
3  import '../styles/admin.css'
4  import { useNavigate } from 'react-router-dom'
5  import axios from 'axios'
6
7  const Admin = () => {
8
9      const navigate = useNavigate();
10
11      const [userCount, setUserCount] = useState(0);
12      const [transactionCount, setTransactionCount] = useState(0);
13      const [depositsCount, setDepositsCount] = useState(0);
14      const [loansCount, setLoansCount] = useState(0);
15
16
17      useEffect(()=>{
18
19          fetchData();
20      }, [])
21
22      const fetchData = async () =>{
23          await axios.get('http://localhost:6001/fetch-users').then(
24              (response)=>{
25
26                  setUserCount(response.data.length);
27
28              }
29          );
30          await axios.get('http://localhost:6001/transactions').then(
31              (response)=>{
32                  setTransactionCount(response.data.length);
```

- AdminDeposits.jsx


 SIRI Bank_Admin Home Users Deposits Loans Transactions Logout		
All Deposits		
Deposit name: Sasi Amount: 500 Duration: 18 months	Nominee name: Harsha Customer name: Venky Start Date: 2024-07-11	Nominee age: 210 Customer A/c id: 668ff55c093d1dea372380e1 Mature Date: 11-6-2025
Deposit name: Mukesh Amount: 50 Duration: 12 months	Nominee name: Sasi Customer name: Venky Start Date: 2024-07-19	Nominee age: 21 Customer A/c id: 668ff55c093d1dea372380e1 Mature Date: 19-6-2025
Deposit name: Mukesh Amount: 100 Duration: 12 months	Nominee name: Sasi Customer name: Mern Start Date: 2024-07-19	Nominee age: 20 Customer A/c id: 669a9b6010c0dc302bc7f5f3 Mature Date: 19-6-2025

```

client > src > Pages > AdminDeposits.jsx > ...
1  import React, { useEffect, useState } from 'react'
2  import Navbar from '../components/Navbar'
3  import '../styles/adminDeposits.css'
4  import axios from 'axios';
5
6  const AdminDeposits = () => {
7
8    const [deposits, setDeposits] = useState([]);
9
10   useEffect(()=>{
11     | fetchDeposits();
12   },[]);
13
14   const fetchDeposits = async () =>{
15     | await axios.get('http://localhost:6001/fetch-deposits').then(
16     | (response) =>{
17     |   | setDeposits(response.data);
18     |   }
19     | )
20   }
21
22   return (
23     <>
24     | <Navbar />
25
26     | <div class="deposits">
27     |   |
28     |   | <h2>All Deposits</h2>
29     |   | <div class="deposits-body">
30     |   |   |
31     |   |   | {deposits.map((deposit)=>{
32     |   |   |   | return(

```

- Adminloans.jsx

 SIRI Bank Admin

HomeUsersDepositsLoansTransactionsLogout

All loans

Loan type: home-loan

Balance: 1000

Duration: 12 months

Nominee name: Mukesh

Total amount: 1000

Start Date: 2024-07-11

Customer name: Venky

Customer A/c id: 668ff55c093d1dea372380e1

End Date: 11-6-2025

Loan type: vehicle-loan

Balance: 100

Duration: 6 months

Nominee name: Harsha

Total amount: 100

Start Date: 2024-07-18

Customer name: Venky

Customer A/c id: 668ff55c093d1dea372380e1

End Date: 18-6-2024

Loan type: personal-loan

Balance: 40

Duration: 6 months

Nominee name: Sasi

Total amount: 100

Start Date: 2024-07-19

Customer name: Venky

Customer A/c id: 668ff55c093d1dea372380e1

End Date: 19-6-2024

Loan type: vehicle-loan

Balance: 50

Duration: 12 months

Nominee name: Harsha

Total amount: 100

Start Date: 2024-07-19

Customer name: Mern

Customer A/c id: 669a9b6010c0dc302bc7f5f3

End Date: 19-6-2025

Pending Applications

Loan type: vehicle-loan

Customer name: Mern

Customer A/c id: 669a9b6010c0dc302bc7f5f3

Amount: 300

Duration: 6 months

Approve

Decline

```

1 import React, { useEffect, useState } from 'react'
2 import Navbar from '../components/Navbar'
3 import '../styles/adminLoans.css'
4 import axios from 'axios';
5
6 const AdminLoans = () => {
7
8     const [loans, setLoans] = useState([]);
9
10    useEffect(()=>{
11        fetchLoans();
12    },[]);
13
14    const fetchLoans = async () =>{
15        await axios.get('http://localhost:6001/fetch-loans').then(
16            (response) =>{
17                setLoans(response.data);
18                console.log(response.data);
19            }
20        )
21    }
22
23    const approveLoan = async(id) =>{
24        await axios.put('http://localhost:6001/approve-loan', {id}).then(
25            (response)=>{
26                alert("Loan approved!!");
27                fetchLoans();
28            }
29        )
30    }
31
32    const declineLoan = async(id) =>{
33        console.log(id)
34        await axios.put('http://localhost:6001/decline-loan', {id}).then(
35            (response)=>{
36                alert("Loan Declined!!");
37                fetchLoans();
38            }
39        )
40    }
41
42    return (
43        <>
44        <Navbar />
45        <div className="loans-page">
46            <div className="loans">
47                <h2>All loans</h2>
48            </div>
49        </div>
50    )

```

- Allusers.jsx



All users

Username Lahari	A/c id 668ff352e9af8d6cf1b0a8e5	IFSC SB007BLR30	Email Lahari@gmail.com	Balance 2150
Username Shashank	A/c id 668ff35e1922d59ee5349d5b	IFSC	Email Shashank@gmail.com	Balance 0
Username Venky	A/c id 668ff55c093d1dea372380e1	IFSC SB007HYD25	Email Venky@gmail.com	Balance 2390
Username Mukesh	A/c id 668ff5a3093d1dea372380e8	IFSC SB007VZG229	Email Mukesh@gmail.com	Balance 2050
Username Sasi	A/c id 668ff5ef093d1dea372380ef	IFSC SB007TPTY05	Email Sasi@gmail.com	Balance 2000
Username	A/c id	IFSC	Email	Balance


```

client > src > Pages > AllUsers.jsx > ...
1  import React, { useEffect, useState } from 'react'
2  import Navbar from '../components/Navbar'
3  import '../styles/allUsers.css'
4  import axios from 'axios';
5
6  const AllUsers = () => {
7
8      const [users, setUsers] = useState([]);
9
10     useEffect(()=>{
11         fetchUsers();
12     },[]);
13
14     const fetchUsers = async () =>{
15         await axios.get('http://localhost:6001/fetch-users').then(
16             (response) =>{
17                 setUsers(response.data);
18             }
19         )
20     }
21
22     return (
23         <>
24             <Navbar />
25
26             <div class="all-users-page">
27                 <h2>All users</h2>
28                 <div class="all-users">
29                     {users.map((user)=>{
30                         return(
31                             <div class="user" key={user._id}>
32                                 <p><b>Username </b>{user.username}</p>
33                                 <p><b>A/c id </b>{user._id}</p>
34                                 <p><b>IFSC </b>{user.ifsc}</p>
35                                 <p><b>Email </b>{user.email}</p>
36                                 <p><b>Balance </b>{user.balance}</p>
37                             </div>
38                         )
39                     })}
40
41                 </div>
42             </div>
43         </div>
44     </>
45 )
46
47 }
48
49
50 export default AllUsers

```

- Alltransactions.jsx



All Transactions

Amount: 50	Receiver:	Receiver's a/c id:	Receiver IFSC:
Sender: Sender's a/c id: 669a9b6010c0dc302bc7f5f3	Payment method:	Time: Fri Jul 19 2024 22:41:09 GMT+0530 (India Standard Time)	
Remarks: Loan re-payment			

Amount: 200	Receiver: Charan	Receiver's a/c id: 668ff94f1738a141eb3a6e6d	Receiver IFSC: SB007CNI99
Sender: Mem	Sender's a/c id: 669a9b6010c0dc302bc7f5f3	Payment method: NEFT	Time: 2024-07-19T17:06:38.508Z
Remarks:			

Amount: 100	Receiver:	Receiver's a/c id:	Receiver IFSC:
Sender: Mern	Sender's a/c id: 669a9b6010c0dc302bc7f5f3	Payment method:	Time: Fri Jul 19 2024 22:34:31 GMT+0530 (India Standard Time)
Remarks: Deposit payment			

Amount: 10	Receiver:	Receiver's a/c id:	Receiver IFSC:
------------	-----------	--------------------	----------------

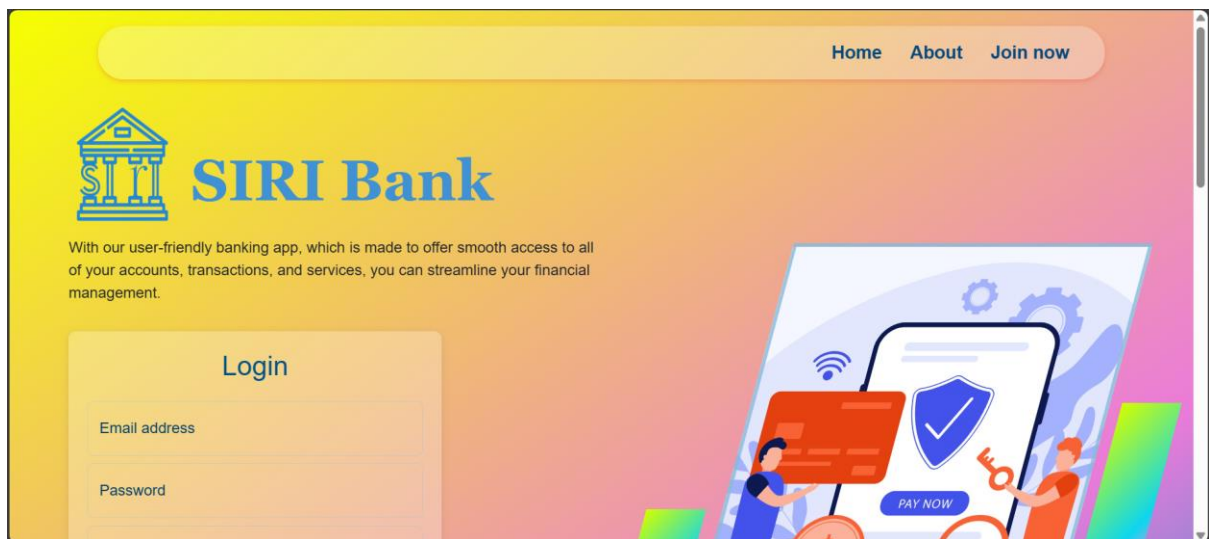
```

client > src > Pages > AllTransactions.jsx > ...
1  import React, { useEffect, useState } from 'react'
2  import Navbar from '../components/Navbar'
3  import '../styles/allTransactions.css'
4  import axios from 'axios';
5
6  const AllTransactions = () => {
7
8      const [transactions, setTransactions] = useState([]);
9
10     useEffect(()=>{
11         fetchDeposits();
12     },[]);
13
14     const fetchDeposits = async () =>{
15         await axios.get('http://localhost:6001/transactions').then(
16             (response) =>{
17                 setTransactions(response.data.reverse());
18             }
19         )
20     }
21
22     return (
23         <>
24             <Navbar />
25
26             <div class="all-transactions-page">
27
28                 <h2>All Transactions</h2>
29
30                 <div class="transactions-body">
31
32                     {transactions.map((transaction) =>{
33                         return(
34                             <div class="transaction" key={transaction._id}>
35                                 <span>
36                                     <p><b>Amount:</b> {transaction.amount} </p>
37                                     <p><b>Receiver:</b> {transaction.receiverName} </p>
38                                     <p><b>Receiver's a/c id:</b> {transaction.receiverId} </p>
39                                     <p><b>Receiver IFSC:</b> {transaction.receiverIFSC} </p>
40                                 </span>
41                                 <span>
42                                     <p><b>Sender:</b> {transaction.senderName} </p>
43                                     <p><b>Sender's a/c id:</b> {transaction.senderId} </p>
44                                     <p><b>Payment method:</b> {transaction.paymentMethod} </p>
45                                     <p><b>Time:</b> {transaction.time} </p>
46                                 </span>
47                                 <p><b>Remarks:</b> {transaction.remarks} </p>
48                             </div>
49                         )
50                     })

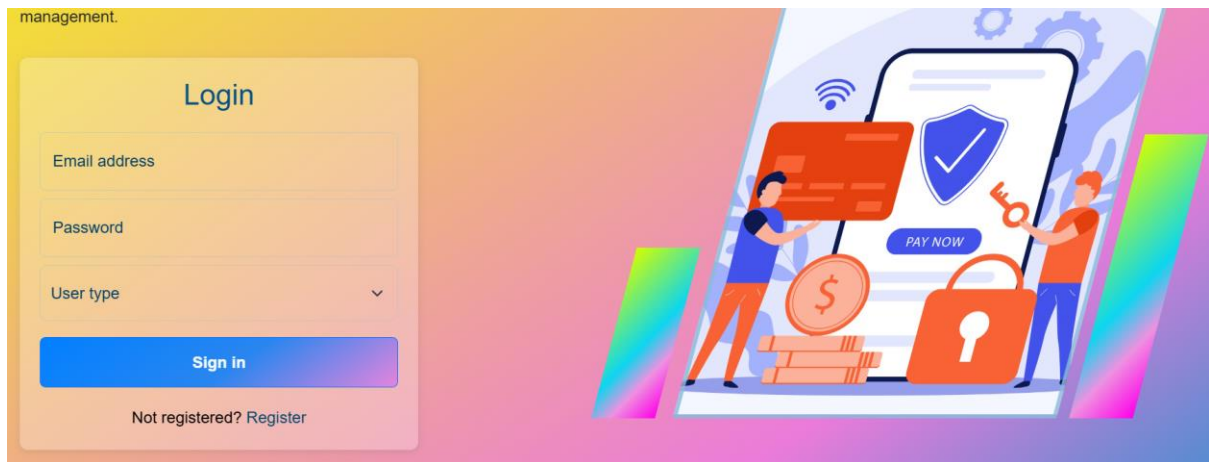
```

Home Page:

- Landing.jsx

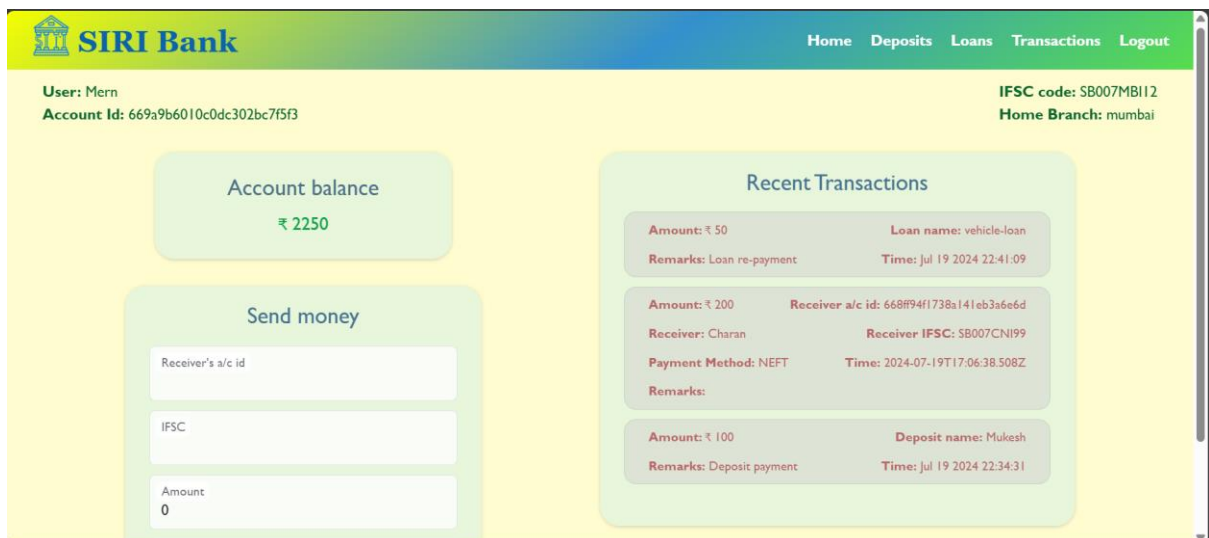


```
client > src > Pages > Landing.jsx > ...
1  import React, { useState } from 'react';
2  import '../styles/landing.css';
3  import About1 from '../assets/bank.jpg';
4  import About2 from '../assets/blue.jpg';
5  import About3 from '../assets/siri.png';
6  import Login from '../components/Login';
7  import Register from '../components/Register';
8
9  const Landing = () => {
10
11     const [isLoginBox, setIsLoginBox] = useState(true);
12
13
14     return (
15       <div className='landingPage'>
16
17         <div className="landing-header">
18           <span className="landing-header-logo"><img src="" alt="" /></span>
19           <ul>
20             <li className='header-li'><a href="#home"><b>Home</b></a></li>
21             <li className='header-li'><a href="#about"><b>About</b></a></li>
22             <li className='header-li'><a href="#home"><b>Join now</b></a></li>
23           </ul>
24         </div>
25
26         <div className="landing-body">
27
28           <div className="landing-hero" id='home'>
29             <div className="landing-hero-content">
30               <div className='banner'>
31                 <img src={About3} className='siri' alt="" />
32                 <h2 className='banktitle'>SIRI Bank</h2>
33               </div>
34
35               <p>With our user-friendly banking app, which is made to offer smooth access to all of your accounts, transactions, &
36
37               <div className="authentication-form">
38                 { isLoginBox ?
39                   <Login setIsLoginBox={setIsLoginBox} />
40                   :
41                   <Register setIsLoginBox={setIsLoginBox} />
42                 }
43               </div>
44             </div>
45           </div>
46         </div>
47       </div>
48     );
49   }
50
```



Customer Page:

- Home.jsx



```

client > src > Pages > Home.jsx > Home
1  import React, { useEffect, useState } from 'react'
2  import '../styles/home.css'
3  import Navbar from '../components/Navbar'
4  import axios from 'axios';
5
6  const Home = () => {
7
8
9      const username = localStorage.getItem('username');
10     const userid = localStorage.getItem('userId');
11     const ifsc = localStorage.getItem('IFSC');
12     const homeBranch = localStorage.getItem('homeBranch');
13
14     const [balance, setBalance] = useState(0);
15     const [sendingAmount, setSendingAmount] = useState(0);
16     const [sendingIFSC, setSendingIFSC] = useState();
17     const [sendingMethod, setSendingMethod] = useState();
18     const [sendingAcId, setSendingAcId] = useState();
19     const [sendingRemarks, setSendingRemarks] = useState('');
20     const [transactions, setTransactions] = useState([]);
21
22
23     useEffect(()=>{
24         fetchUserData();
25     }, [])
26
27     const fetchUserData = async () => {
28         console.log(userid);
29         try{
30             if (userid) {
31                 await axios.get(`http://localhost:6001/user-details/${userid}`).then(
32                     async (response) => {
33                         setBalance(response.data.balance);
34                         console.log(response);
35                     }
36                 ).catch((err)=>{
37                     console.log(err);
38                 });
39
40                 await axios.get(`http://localhost:6001/transactions`).then(
41                     async (response) => {
42                         console.log(response);
43                         setTransactions(response.data.reverse());
44                     }
45                 ).catch((err)=>{
46                     console.log(err);
47                 });
48             }
49         }catch(err){
50             console.log(err);

```

- UserTransactions.jsx

To send money:

User: Mern

Account Id: 669a9b6010c0dc302bc75f3

IFSC code: SB007MB112

Home Branch: mumbai

Account balance

₹ 2250

Send money

Receiver's a/c id

IFSC

Amount

0

Remarks

Send

Recent Transactions

Amount: ₹ 50

Loan name: vehicle loan

Remarks: Loan re-payment

Time: Jul 19 2024 22:41:09

Amount: ₹ 200

Receiver a/c id: 668f94173b141eb3de6d

Receiver: Charan

Receiver IFSC: SB007CN899

Payment Method: NEFT

Time: 2024-07-19T17:06:38.506Z

Remarks:

Amount: ₹ 100

Deposit name: Mukesh

Remarks: Deposit payment

Time: Jul 19 2024 22:34:31

```

client > src > Pages > UserTransactions.jsx > UserTransactions > map() callback
1  import axios from 'axios';
2  import React, { useEffect, useState } from 'react'
3  import Navbar from '../components/Navbar';
4
5  const UserTransactions = () => {
6
7      const userId = localStorage.getItem('userId');
8
9      const [transactions, setTransactions] = useState([]);
10
11      useEffect(()=>{
12          fetchDeposits();
13      },[]);
14
15      const fetchDeposits = async () =>{
16          await axios.get('http://localhost:6001/transactions').then(
17              (response) =>{
18                  setTransactions(response.data.reverse());
19              }
20          )
21      }
22
23
24      return (
25          <>
26          <Navbar />
27
28          <div class="all-transactions-page">
29
30              <h2>All Transactions</h2>
31
32              <div class="transactions-body">
33
34                  {transactions && transactions.filter(transaction => transaction.senderId === userId || transaction.receiverId === userId).map((transaction)=>{
35
36                      return(
37                          <>
38                          {transaction.senderId === userId ?
39
40                              <>
41                              {transaction.deposit || transaction.loan?
42
43                                  <div className=" user-transaction money-sent" key={transaction._id}>
44                                  <span>
45                                  <p><b>Amount: </b>₹8377; {transaction.amount}</p>
46                                  {transaction.deposit ? <p><b>Deposit name:</b> {transaction.deposit}</p>
47                                  : <p><b>Loan name:</b> {transaction.loan}</p>
48
49                                  </span>

```

Account balance

₹ 2000

Send money

Receiver's a/c id
668ff94f1738a141eb3a6e6d

IFSC
SB007CNI99

Choose payment method
IMPS
NEFT

Choose payment method

Recent Transactions

Amount: ₹ 250
Deposit name: For Home

Remarks: Deposit payment
Time: Jul 20 2024 22:03:32

Amount: ₹ 50
Loan name: vehicle-loan

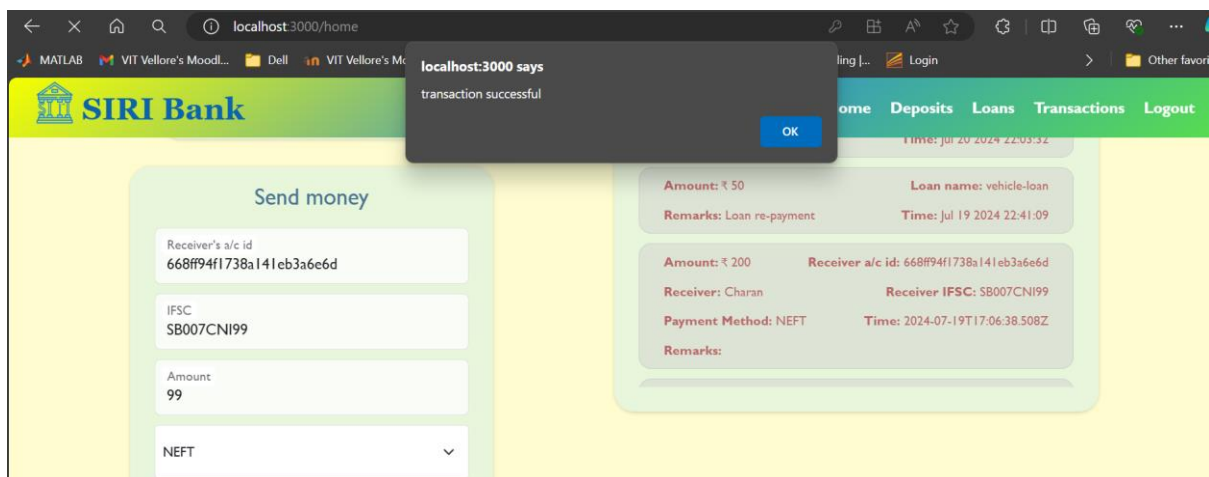
Remarks: Loan re-payment
Time: Jul 19 2024 22:41:09

Amount: ₹ 200
Receiver a/c id: 668ff94f1738a141eb3a6e6d

Receiver: Charan
Receiver IFSC: SB007CNI99

Payment Method: NEFT
Time: 2024-07-19T17:06:38.508Z

Remarks:



After changing the balance

User: Mern

Account Id: 669a9b6010c0dc302bc7f5f3

IFSC code: SB007MBI12

Home Branch: mumbai

Account balance

₹ 1901

Send money

Recent Transactions

Amount: ₹ 99
Receiver a/c id: 668ff94f1738a141eb3a6e6d

Receiver: Charan
Receiver IFSC: SB007CNI99

Payment Method: NEFT
Time: 2024-07-20T16:34:57.239Z

Remarks:

To see all transactions:

 **SIRI Bank**

HomeDepositsLoansTransactionsLogout

All Transactions

Amount: ₹ 50

Remarks: Loan re-payment

Loan name: vehicle-loan

Time: Jul 19 2024 22:41:09

Amount: ₹ 200

Receiver: Charan

Payment Method: NEFT

Remarks:

Receiver a/c id: 668ff94f1738a141eb3a6e6d

Receiver IFSC: SB007CNI99

Time: 2024-07-19T17:06:38.508Z


Amount: ₹ 100

Remarks: Deposit payment

Deposit name: Mukesh

Time: Jul 19 2024 22:34:31

To make a Fixed Deposit (FD):

 **SIRI Bank**

HomeDepositsLoansTransactionsLogout

Fixed Deposits

Deposit name: Mukesh

Nominee name: Sasi

Nominee age: 20

Amount: 100

Duration: 12 months

Maturity Date: 19-6-2025

New Fixed deposit

Deposit name

Nominee name

Age

0

Deposit amount

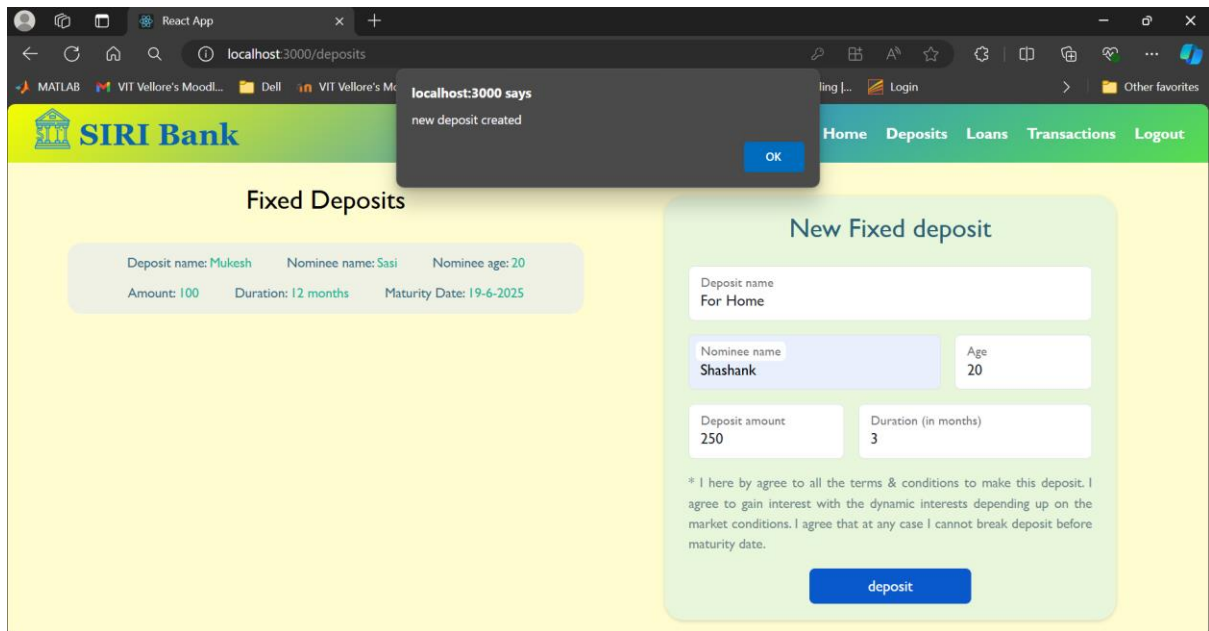
0

Duration (in months)

0

* I here by agree to all the terms & conditions to make this deposit. I agree to gain interest with the dynamic interests depending up on the market conditions. I agree that at any case I cannot break deposit before maturity date.

deposit




```

client > src > Pages > Deposits.jsx > ...
1  import React, { useEffect, useState } from 'react'
2  import Navbar from '../components/Navbar'
3  import '../styles/deposits.css'
4  import axios from 'axios'
5
6  const Deposits = () => {
7
8      const username = localStorage.getItem('username');
9      const userid = localStorage.getItem('userId');
10
11     const [newDepositName, setNewDepositName] = useState('');
12     const [newNomineeName, setNewNomineeName] = useState('');
13     const [newNomineeAge, setNewNomineeAge] = useState(0);
14     const [newDepositAmount, setNewDepositAmount] = useState(0);
15     const [newDepositDuration, setNewDepositDuration] = useState(0);
16
17     const [deposits, setDeposits] = useState([]);
18
19     useEffect(()=>{
20         fetchDepositsData();
21     }, [])
22
23     const fetchDepositsData = async () => {
24
25         try{
26             if (userid) {
27
28                 await axios.get(`http://localhost:6001/fetch-deposits`).then(
29                     async (response) => {
30                         console.log(response);
31                         setDeposits(response.data.reverse());
32                     }
33                 ).catch((err)=>{
34                     console.log(err);
35                 });
36             }
37         }catch(err){
38             console.log(err);
39         }
40     }
41
42     const createNewDeposit = async () =>{
43
44         const newDepositDetails = {depositName: newDepositName, customerId: userid,
45                                     customerName: username, nomineeName: newNomineeName,
46                                     nomineeAge: newNomineeAge, duration: newDepositDuration,
47                                     amount: newDepositAmount, createdAt: new Date()};
48         await axios.post('http://localhost:6001/new-deposit', newDepositDetails).then(
49             async (response) =>{

```

To apply for a LOAN:

 **SIRI Bank**

HomeDepositsLoansTransactionsLogout

Loans

Loan type: vehicle-loanNominee name: ShashankLoan amount: 300

Balance: 300Duration: 6 monthsLoan status: pending

Loan type: vehicle-loanNominee name: HarshaLoan amount: 100

Balance: 50Duration: 12 monthsEnd Date: 19-6-2025

Make payment

Apply for New Loan

Choose loan type

Nominee name

Age

0

Loan amount


0

Duration (in months)

0

* I here by agree to all the terms & conditions to make this loan. I agree to pay interest with the dynamic interests depending up on the market conditions. I agree to repay the loan before the deadline.

Apply

 **SIRI Bank**

HomeDepositsLoansTransactionsLogout

Loans

Loan type: vehicle-loanNominee name: HarshaLoan amount: 100

Balance: 50Duration: 12 monthsEnd Date: 19-6-2025

Make payment

Apply for New Loan

Vehicle loan

Nominee name

Shashank

Age

21

Loan amount

300

Duration (in months)

6

* I here by agree to all the terms & conditions to make this loan. I agree to pay interest with the dynamic interests depending up on the market conditions. I agree to repay the loan before the deadline.

Apply

localhost:3000 says
new Loan created
OK

```

client > src > Pages > Loans.jsx > ...
1 import React, { useEffect, useState } from 'react'
2 import Navbar from '../components/Navbar'
3 import '../styles/loans.css'
4 import axios from 'axios';
5
6 const Loans = () => {
7
8     const username = localStorage.getItem('username');
9     const userid = localStorage.getItem('userId');
10
11     const [newLoanType, setNewLoanType] = useState('');
12     const [newNomineeName, setNewNomineeName] = useState('');
13     const [newNomineeAge, setNewNomineeAge] = useState(0);
14     const [newLoanAmount, setNewLoanAmount] = useState(0);
15     const [newLoanDuration, setNewLoanDuration] = useState(0);
16
17     const [paymentLoanId, setPaymentLoanId] = useState();
18     const [paymentLoanAmount, setPaymentLoanAmount] = useState(0);
19
20     const [loans, setLoans] = useState([]);
21
22     useEffect(()=>{
23         fetchLoansData();
24     }, [])
25
26     const fetchLoansData = async () => {
27         try{
28             if (userid) {
29                 await axios.get(`http://localhost:6001/fetch-loans`).then(
30                     async (response) => {
31                         console.log(response);
32                         setLoans(response.data.reverse());
33                     }
34                 ).catch((err)=>{
35                     console.log(err);
36                 });
37             }
38         }catch(err){
39             console.log(err);
40         }
41     }
42
43     const createNewLoan = async () =>{
44
45         const newLoanDetails = {loanType: newLoanType, customerId: userid, customerName: username,
46                                 nomineeName: newNomineeName, nomineeAge: newNomineeAge, duration: newLoanDuration,
47                                 loanAmount: newLoanAmount, createdAt: new Date()};
48         await axios.post('http://localhost:6001/new-loan', newLoanDetails).then(

```

- ❖ For Everything we get a pop-up window:
 - Like from above pics

Routing

Routing is managed using React Router. Here are the main routes:

- **home** - Landing page of the application.
- **dashboard** - Dashboard displaying user data and statistics.
- **profile** - User profile management.

Integration with Backend

The frontend communicates with the backend APIs hosted on [backend URL]. Key endpoints include:

- **GET /api/data** - Retrieves data for display.

```
app.get('/user-details/:id', async (req, res) => {
  try{
    const user = await User.findOne({_id: req.params.id});
    if(!user){
      return res.status(404).json({ message: 'User not found' });
    }
    res.json(user);
  } catch (error) {
    console.log(error);
    return res.status(500).json({ message: 'Server Error' });
  }
});
```

- **POST /api/user/login** - Handles user authentication.

```

app.post('/login', async (req, res) => {
  const { email, usertype, password } = req.body;
  try {
    if (usertype === 'customer'){
      const user = await User.findOne({ email });

      if (!user) {
        return res.status(401).json({ message: 'Invalid email or password' });
      }
      const isMatch = await bcrypt.compare(password, user.password);
      if (!isMatch) {
        return res.status(401).json({ message: 'Invalid email or password' });
      } else{
        return res.json(user);
      }
    }else if (usertype === 'admin'){
      const user = await Bank.findOne({ email });

      if (!user) {
        return res.status(401).json({ message: 'Invalid email or password' });
      }
      const isMatch = await bcrypt.compare(password, user.password);
      if (!isMatch) {
        return res.status(401).json({ message: 'Invalid email or password' });
      } else{
        return res.json(user);
      }
    }
  } catch (error) {
    console.log(error);
    return res.status(500).json({ message: 'Server Error' });
  }
});

```

User Interface (UI) Design

The user interface design adheres to [explain design principles].

- **Accessibility:** Considering the needs of users with disabilities, including features like keyboard navigation, support for screen readers, and adequate contrast between colours.
- **User-Centric:** The design prioritises the demands of the user, making sure that important activities are simple to reach and that user flows make sense. Quick feedback is given in response to user actions.

- **Responsiveness:** The user interface (UI) is flexible and adjusts to different screen sizes and gadgets with ease, making it accessible and easy to use on computers, tablets, and mobile devices.
- **Consistency:** To create a seamless user experience, the application's layout, colour palettes, typography, and interaction patterns should all be consistent.
- **Simplicity:** The user interface is made to be as easy to use as possible, with a minimal learning curve for the user. Clear presentation of the elements prevents clutter and confusion.

Used [UI framework/library] for implementation.

- **UI Framework/Library:** React, a well-liked JavaScript user interface library, is used to develop the UI. React makes it easier to create reusable components and render the user interface (UI) more quickly.
- **CSS Framework:** Bootstrap is used for styling, offering a complete set of pre-defined classes and components to create modern, responsive user interfaces. For styling requirements that are beyond the scope of the framework, custom CSS is also included.
- **Routing:** React Router manages the application's navigation and routing, allowing for seamless view transitions and preserving the single-page application (SPA) experience.
- **Form managing:** Yup is used for validation and Formic is used for managing user input in forms. This combination makes sure that form submissions are handled effectively and that validations are applied consistently.
- **HTTP Requests:** Axios offers a clear and easy-to-use syntax for communicating with the server, managing errors and responses, and submitting HTTP requests to the backend API.