

Name: Onkar Anil Waghmode

Div: TY-CS-D

Roll No.: 81

PRN: 12210334

Assignment 5: Design and develop a responsive website for an online Grocery Shop using React, NodeJS and MySQL having 1) Home Page 2) Login Page 3) Catalogue Page: 4) Registration Page: (database)

Frontend:

App.js

```
import React from 'react';
import { BrowserRouter as Router, Route, Routes } from 'react-router-dom';
import Home from './components/Home';
import Login from './components/Login';
import Register from './components/Register';
import OwnerDashboard from './components/OwnerDashboard';
import CustomerDashboard from './components/CustomerDashboard';

function App() {
  return (
    <Router>
      <Routes>
        <Route path="/" element={<Home />} />
        <Route path="/login/:role" element={<Login />} />
        <Route path="/register/:role" element={<Register />} />
        <Route path="/owner-dashboard" element={<OwnerDashboard />} />
        <Route path="/customer-dashboard" element={<CustomerDashboard />} />
      </Routes>
    </Router>
  );
}

export default App;
```

Home.jsx

```
import React from 'react';
import { Link } from 'react-router-dom';
import './styles/Home.css';

const Home = () => {
  return (
    <div className="home-container">
      <h1>Welcome to Grocery Shop</h1>
      <div className="card-container">
        <Link to="/login/owner" className="role-card">Shop Owner</Link>
        <Link to="/login/customer" className="role-card">Customer</Link>
      </div>
    </div>
  );
};

export default Home;
```

Home.css

```
.home-container {
  display: flex;
  flex-direction: column;
  align-items: center;
  margin-top: 60px;
  font-family: 'Segoe UI', sans-serif;
}

.card-container {
  display: flex;
  gap: 30px;
  margin-top: 40px;
}

.role-card {
  padding: 20px 40px;
  background-color: #4CAF50;
  color: white;
  font-size: 18px;
  text-decoration: none;
  border-radius: 10px;
  box-shadow: 0 4px 8px rgba(0,0,0,0.2);
  transition: 0.3s;
}
```

```
.role-card:hover {  
  background-color: #45a049;  
}
```

Register.jsx

```
import React, { useState } from 'react';  
import { useParams, useNavigate } from 'react-router-dom';  
import axios from 'axios';  
import './styles/Form.css';  
  
const Register = () => {  
  const { role } = useParams();  
  const navigate = useNavigate();  
  const [username, setUsername] = useState('');  
  const [password, setPassword] = useState('');  
  
  const register = async () => {  
    try {  
      await axios.post(`http://localhost:5000/register/${role}`, { username, password });  
      alert('Registered successfully!');  
      navigate(`/login/${role}`);  
    } catch (err) {  
      alert('Registration failed!');  
    }  
  };  
  
  return (  
    <div className="form-container">  
      <h2>Register as {role.charAt(0).toUpperCase() + role.slice(1)}</h2>  
      <input placeholder="Username" onChange={e =>  
        setUsername(e.target.value)} />  
      <input type="password" placeholder="Password" onChange={e =>  
        setPassword(e.target.value)} />  
      <button onClick={register}>Register</button>  
    </div>  
  );  
};  
  
export default Register;
```

Login.jsx

```
import React, { useState } from 'react';
import { useParams, useNavigate, Link } from 'react-router-dom';
import axios from 'axios';
import './styles/Form.css';

const Login = () => {
  const { role } = useParams();
  const navigate = useNavigate();
  const [username, setUsername] = useState('');
  const [password, setPassword] = useState('');

  const login = async () => {
    try {
      const res = await axios.post(`http://localhost:5000/login/${role}`, {
        username, password });
      alert('Login successful!');
      navigate(role === 'owner' ? '/owner-dashboard' : '/customer-dashboard',
      { state: res.data });
    } catch (err) {
      alert('Login failed!');
    }
  };

  return (
    <div className="form-container">
      <h2>Login as {role.charAt(0).toUpperCase() + role.slice(1)}</h2>
      <input placeholder="Username" onChange={e =>
        setUsername(e.target.value)} />
      <input type="password" placeholder="Password" onChange={e =>
        setPassword(e.target.value)} />
      <button onClick={login}>Login</button>
      <p className="switch-auth">
        Don't have an account? <Link to={`/register/${role}`}>Sign up
        here</Link>
      </p>
    </div>
  );
};

export default Login;
```

Form.css

```
.form-container {  
    width: 300px;  
    margin: 100px auto;  
    padding: 40px;  
    background: #f9f9f9;  
    border-radius: 10px;  
    box-shadow: 0px 0px 15px rgba(0,0,0,0.1);  
    text-align: center;  
}  
  
.form-container input {  
    width: 90%;  
    margin-bottom: 20px;  
    padding: 10px;  
    border: 1px solid #ccc;  
    border-radius: 5px;  
}  
  
.form-container button {  
    background-color: #28a745;  
    color: white;  
    border: none;  
    padding: 10px 25px;  
    border-radius: 5px;  
    cursor: pointer;  
}  
  
.form-container button:hover {  
    background-color: #218838;  
}  
  
.switch-auth {  
    margin-top: 15px;  
    font-size: 14px;  
}  
  
.switch-auth a {  
    color: #007bff;  
    text-decoration: none;  
}  
  
.switch-auth a:hover {  
    text-decoration: underline;  
}
```

OwnerDashboard.jsx

```
import React, { useState } from 'react';
import axios from 'axios';
import { useLocation } from 'react-router-dom';
import './styles/Dashboard.css';

const OwnerDashboard = () => {
  const { state } = useLocation();
  const [product, setProduct] = useState({ name: '', price: '', quantity: '' });

  const addProduct = async () => {
    await axios.post('http://localhost:5000/add-product', {
      ...product,
      owner_id: state.id,
    });
    alert('Product added!');
  };

  return (
    <div className="dashboard">
      <h2>Owner Dashboard</h2>
      <input placeholder="Product Name" onChange={e => setProduct({
        ...product, name: e.target.value })} />
      <input placeholder="Price" type="number" onChange={e => setProduct({
        ...product, price: e.target.value })} />
      <input placeholder="Quantity" type="number" onChange={e => setProduct({
        ...product, quantity: e.target.value })} />
      <button onClick={addProduct}>Add Product</button>
    </div>
  );
};

export default OwnerDashboard;
```

CustomerDashboard.jsx

```
import React, { useEffect, useState } from 'react';
import axios from 'axios';
import { useLocation } from 'react-router-dom';
import './styles/Dashboard.css';

const CustomerDashboard = () => {
  const { state } = useLocation();
```

```

const [products, setProducts] = useState([]);

useEffect(() => {
  axios.get('http://localhost:5000/products')
    .then(res => setProducts(res.data));
}, []);

const buy = async (id) => {
  const qty = prompt("Enter quantity to buy:");
  await axios.post('http://localhost:5000/buy', {
    customer_id: state.id,
    product_id: id,
    quantity: parseInt(qty),
  });
  alert("Purchase successful!");
};

return (
  <div className="dashboard">
    <h2>Customer Dashboard</h2>
    {products.map(p => (
      <div key={p.id} className="product-card">
        <h3>{p.name}</h3>
        <p>Price: ₹{p.price}</p>
        <p>Available: {p.quantity}</p>
        <button onClick={() => buy(p.id)}>Buy Now</button>
      </div>
    )));
  </div>
);
};

export default CustomerDashboard;

```

Dashboard.css

```

.dashboard {
  padding: 40px;
  text-align: center;
  font-family: 'Segoe UI', sans-serif;
}

.dashboard input {
  margin: 10px;
  padding: 10px;
}

```

```
width: 250px;
}

.dashboard button {
  padding: 10px 20px;
  background-color: #007bff;
  color: white;
  border: none;
  border-radius: 6px;
  cursor: pointer;
}

.dashboard button:hover {
  background-color: #0056b3;
}

.product-card {
  border: 1px solid #ccc;
  margin: 20px auto;
  padding: 20px;
  width: 300px;
  background: #f1f1f1;
  border-radius: 10px;
}
```

Backend:

server.js

```
const express = require('express');
const cors = require('cors');
const bodyParser = require('body-parser');
const db = require('./db');
const app = express();

app.use(cors());
app.use(bodyParser.json());

const PORT = 5000;

// Routes
```

```

app.post('/register/:role', async (req, res) => {
  const { username, password } = req.body;
  const role = req.params.role;

  const table = role === 'owner' ? 'owners' : 'customers';
  const sql = `INSERT INTO ${table} (username, password) VALUES (?, ?)`;
  db.query(sql, [username, password], (err) => {
    if (err) return res.status(400).json({ error: 'User already exists' });
    res.json({ message: 'Registered successfully' });
  });
});

app.post('/login/:role', (req, res) => {
  const { username, password } = req.body;
  const table = req.params.role === 'owner' ? 'owners' : 'customers';

  db.query(`SELECT * FROM ${table} WHERE username = ? AND password = ?`,
  [username, password], (err, result) => {
    if (err || result.length === 0) return res.status(401).json({ error: 'Invalid credentials' });
    res.json({ id: result[0].id, username });
  });
});

// Product registration by owner
app.post('/add-product', (req, res) => {
  const { owner_id, name, price, quantity } = req.body;
  db.query(`INSERT INTO products (owner_id, name, price, quantity) VALUES (?, ?, ?, ?)`, [owner_id, name, price, quantity], (err) => {
    if (err) return res.status(500).json({ error: 'Could not add product' });
    res.json({ message: 'Product added' });
  });
});

// Get all products
app.get('/products', (req, res) => {
  db.query(`SELECT * FROM products`, (err, result) => {
    if (err) return res.status(500).json({ error: 'Error fetching products' });
    res.json(result);
  });
});

// Buy product
app.post('/buy', (req, res) => {
  const { customer_id, product_id, quantity } = req.body;

```

```

db.query(`SELECT * FROM products WHERE id = ?`, [product_id], (err, results)
=> {
  if (err || results.length === 0) return res.status(404).json({ error: 'Product not found' });
  const product = results[0];
  if (product.quantity < quantity) return res.status(400).json({ error: 'Not enough quantity' });

  const newQty = product.quantity - quantity;

  db.query(`UPDATE products SET quantity = ? WHERE id = ?`, [newQty, product_id]);
  db.query(`INSERT INTO purchases (customer_id, product_name, quantity)
VALUES (?, ?, ?)`, [customer_id, product.name, quantity]);
  res.json({ message: 'Purchase successful' });
});

app.listen(PORT, () => console.log(`Server running on port ${PORT}`));

```

db.js

```

const mysql = require('mysql2');

const db = mysql.createConnection({
  host: 'localhost',
  user: 'root',
  password: '*****',
  database: 'wt_grocery_shop',
});

db.connect((err) => {
  if (err) throw err;
  console.log('MySQL Connected');
});

module.exports = db;

```

Output:

```
$ nodemon server.js
[nodemon] 3.0.2
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): ***!
[nodemon] watching extensions: js,mjs,cjs,json
[nodemon] starting `node server.js`
Server running on port 5000
MySQL Connected
New version of nodemon available!
  Current Version: 3.0.2
  Latest Version: 3.1.10
$ npm start
Compiled successfully!

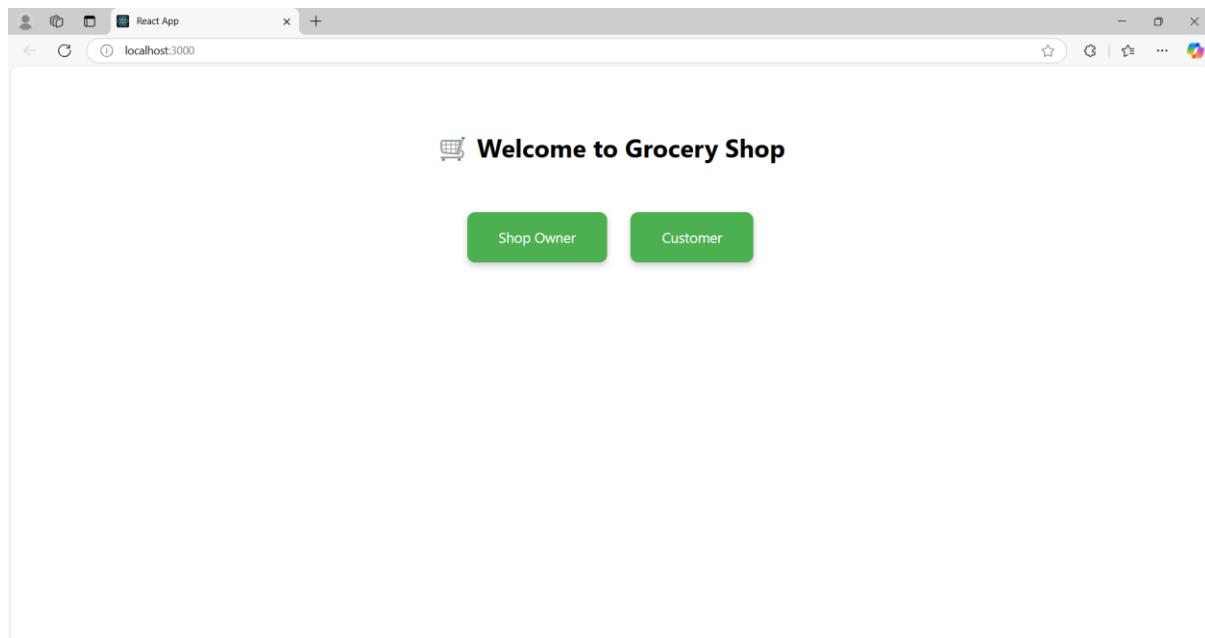
You can now view frontend in the browser.

  Local:          http://localhost:3000
  On Your Network: http://192.168.1.17:3000

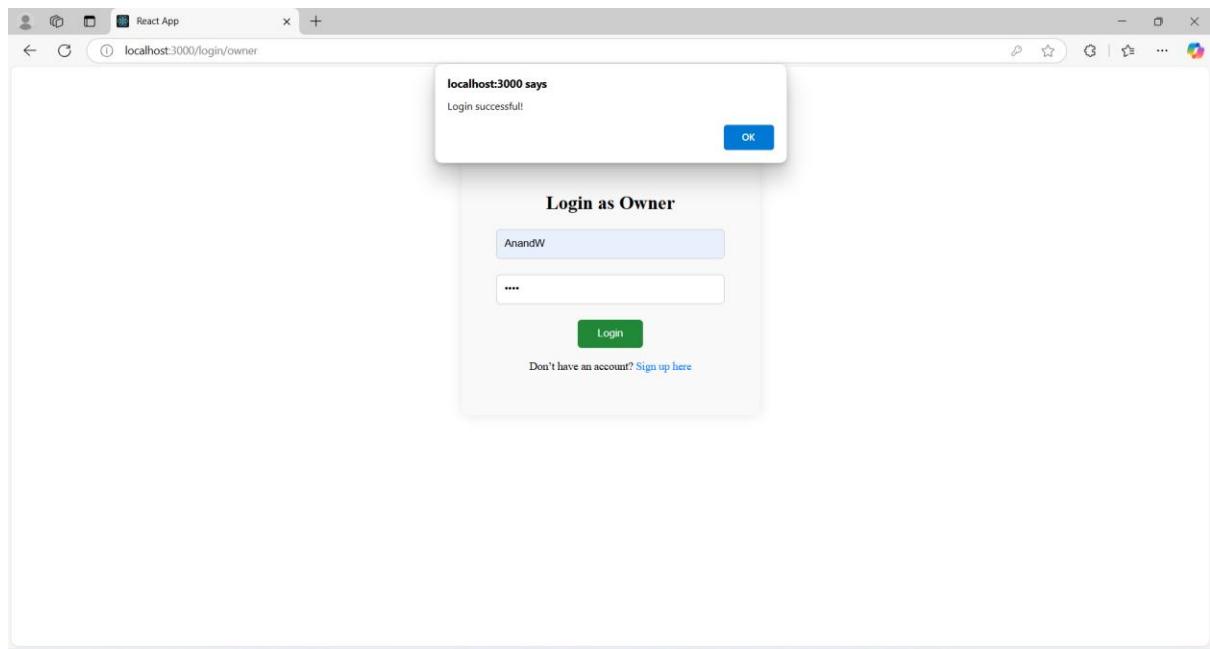
Note that the development build is not optimized.
To create a production build, use npm run build.

webpack compiled successfully
```

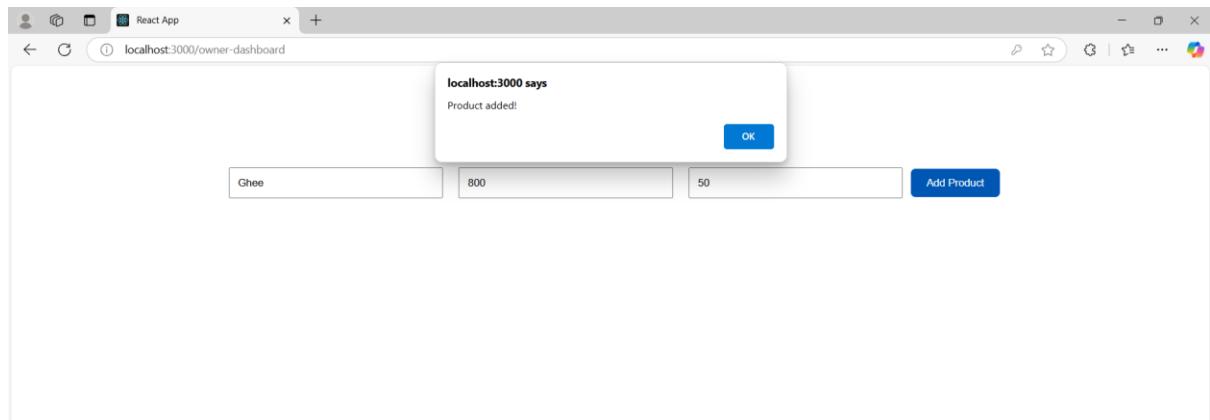
Homepage:



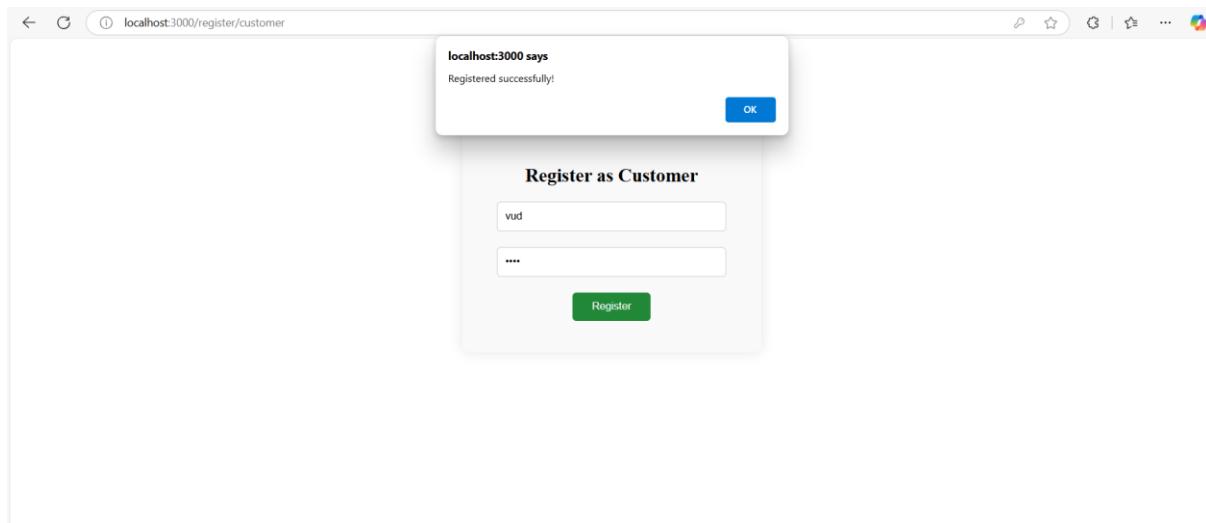
Owner Login Page:



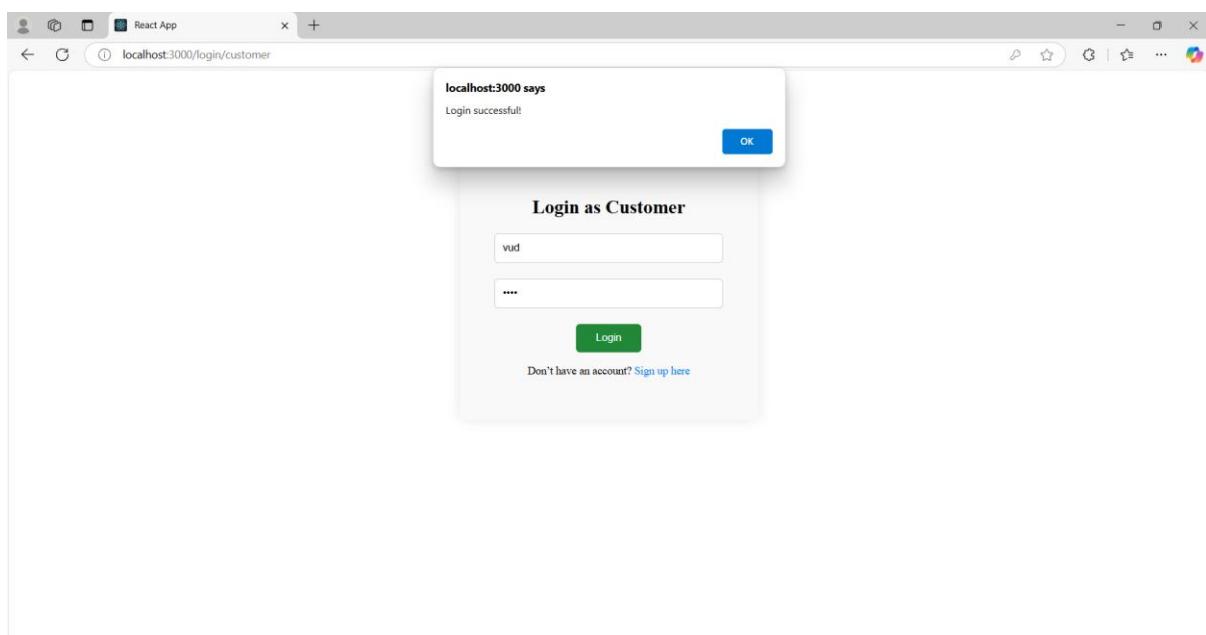
Owner Product Adding Page:



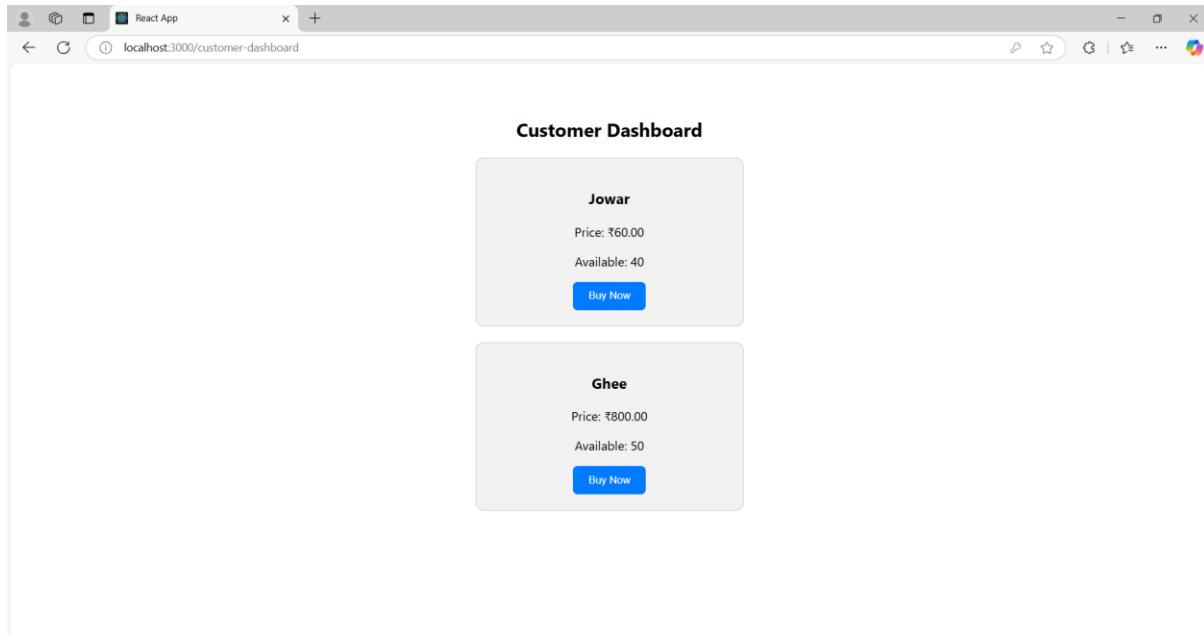
Customer Registration (Sign-up) Page:



Customer Login Page:



Customer Dashboard Page:



Catalogue Page:

