Project Report and output

Project Team Members:

Gunasekar.B(Leader)

Simeon.A

Ranaprathab.G

Raja.R

Sugan rajkumar.A

Project Title: E-Commerce Application

Project Summary

This project is a frontend implementation of an e-commerce platform, built using **React** and **React Router** for routing, along with several additional libraries to enhance functionality and improve the user experience. This platform allows customers to browse products, manage their shopping carts, view individual product details, and manage their profiles. It also includes an **Admin** interface for product, user, and order management.

Table of Contents

1. Project Setup

Ensure that you have **Node.js** installed (version 14+ is recommended).

Steps to Install

1. Clone the repository: git clone
2. Navigate to the project folder: cd frontend
3. Install all dependencies: npm install
4. Start the development server: npm start

This will start the application on http://localhost:3000.

2. Libraries Used

This project leverages the following libraries, along with their installation commands:

1. **React** - Core framework. npx create-react-app
2. **React Router** - For routing and navigation. npm install react-router-dom
3. **Axios** - For API requests. npm install axios
4. **Context API** - Used for global state management, no additional installation needed as it is included in React.
5. **React Testing Library and Jest** - For testing components. npm install --save-dev @testing-library/react @testing-library/jest-dom
6. **Web Vitals** - For measuring performance. npm install web-vitals
7. **dotenv** - Used for managing environment variables. npm install dotenv

3. Folder Structure and Overview

```
frontend/ ├── src/ │ ├── components/ # Reusable components │ │ └── Navbar.jsx │ ├── context/
# Context for global state │ │ └── GeneralContext.js │ ├── pages/ # Main page components │ │
├── Home.jsx │ │ ├── Authentication.jsx │ │ ├── customer/ # Customer-specific pages │ │ │ ├──
Profile.jsx │ │ │ ├── Cart.jsx │ │ └── admin/ # Admin-specific pages │ │ ├── Admin.jsx │ │ ├──
AllProducts.jsx │ ├── App.js # Main app component │ ├── index.js # Entry point │ └── index.css #
Global styles └── public/ └── index.html
```

- **components/** - Contains reusable components like Navbar.
- **context/** - Contains global state management using React’s Context API.
- **pages/** - Contains main pages for the app, divided into **customer** and **admin** functionalities.

4. Component Documentation

Navbar.jsx

This component is a reusable navigation bar. It utilizes useNavigate from React Router for navigation to different routes such as /profile, /cart, and /admin.

Home.jsx

Displays the homepage with general product categories or featured products. Links to different product categories using the router.

Authentication.jsx

Handles user authentication (login and registration).

Profile.jsx (Customer)

Displays the customer’s profile, including order history and personal information. Integrates with useContext for accessing global user data.

Cart.jsx (Customer)

Allows users to view items in their cart and proceed to checkout. Uses Context API for cart state management.

Admin.jsx

Admin dashboard that links to all administrative functionalities.

AllProducts.jsx (Admin)

Displays all products in the database with options to edit or delete each product.

5. Context and State Management

GeneralContext.js

This file in the context folder creates a React Context to manage global states, such as:

- User authentication state
- Cart items
- Any other global settings or data

Usage

Wrap the App component in GeneralContextProvider (in index.js) to make the global state accessible across the application.

6. API Integration

APIs are managed using axios. The base URL is defined in each component where it’s used, making it possible to switch between development and production URLs if needed.

Example

In Product.js:

```
useEffect(() => { axios.get('http://localhost:6001/api/products') .then(response =>
{ setProducts(response.data); }) .catch(error => { console.error('Error fetching products:',
error); }); }, []);
```

7. Testing

Tests are written using **Jest** and **React Testing Library**.

- **App.test.js** - A basic test to check that the app renders.
- **setupTests.js** - Configures Jest with DOM matchers from @testing-library/jest-dom.

To run tests: npm test

8. Potential Improvements

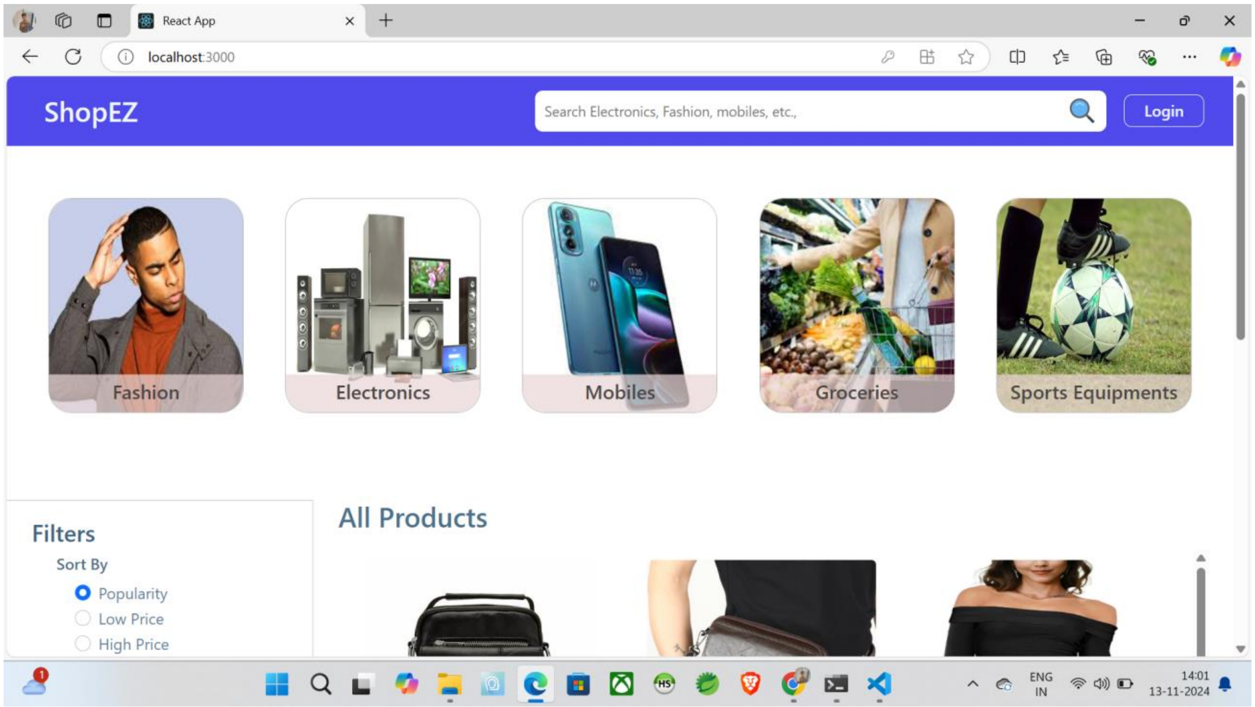
- **Error handling**: Add comprehensive error handling to provide feedback to users during API failures.
- **User roles and permissions**: Improve the admin and user separation, with more robust access controls.
- **Styling**: Use a CSS preprocessor like SASS or a library like Tailwind CSS for better style management and design consistency.

- **Environment Variables**: Use .env files to manage API URLs and other configurations more effectively.

9. Conclusion

This report should serve as a useful guide to setting up, understanding, and maintaining the frontend of this e-commerce web application. Each library and component is listed with its purpose and installation, creating a strong foundation for further development and enhancement.

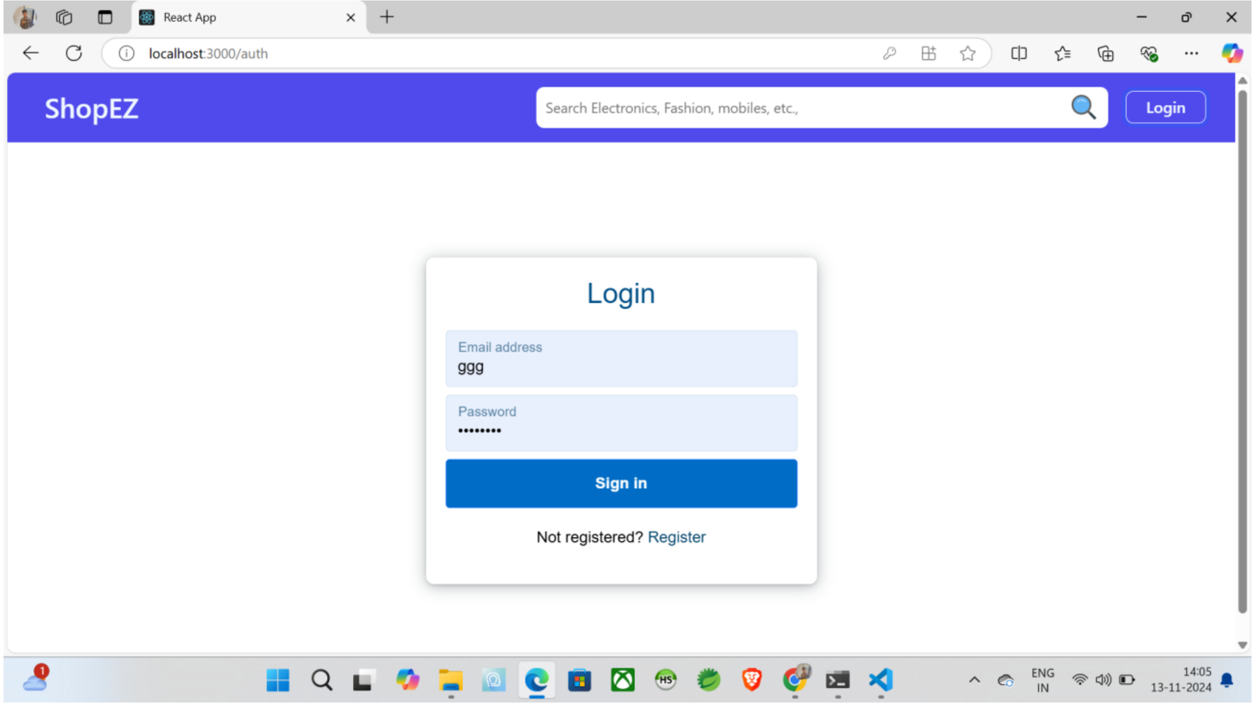
Project Output:



Filters

Sort By

● Popularity
○ Low Price
○ High Price

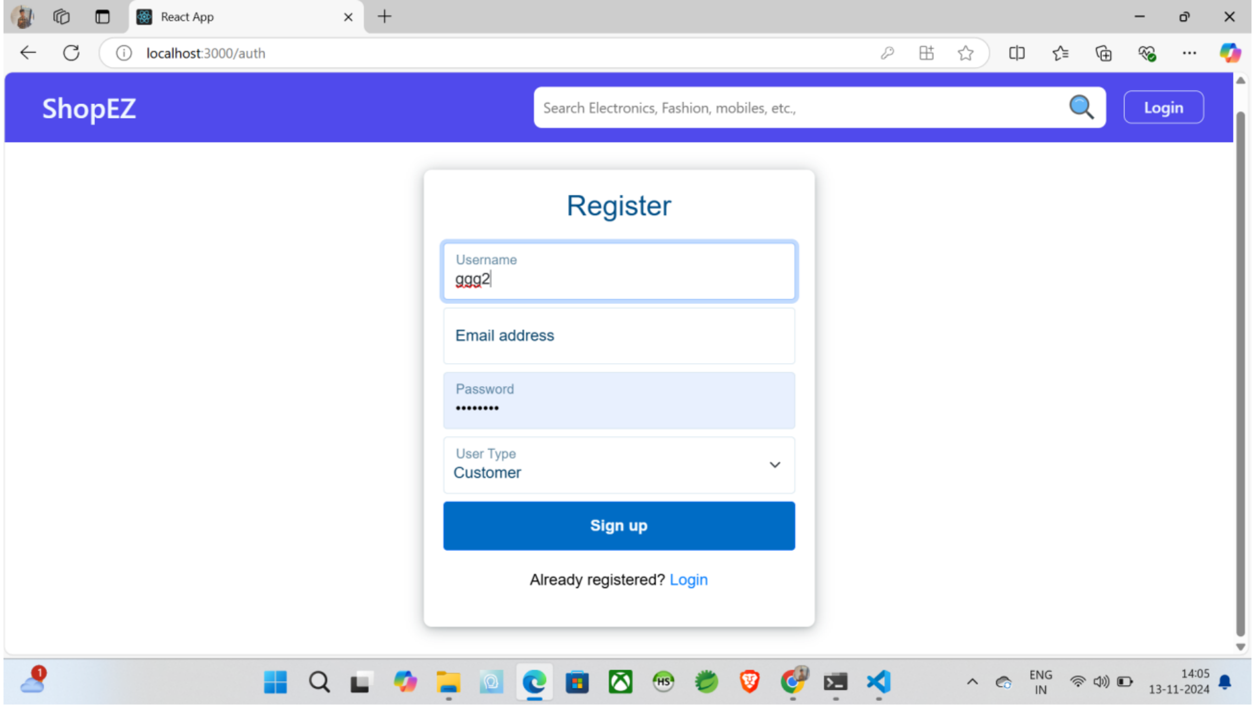


Login

Email address
ggg

Password
••••••••

Not registered? Register



Register

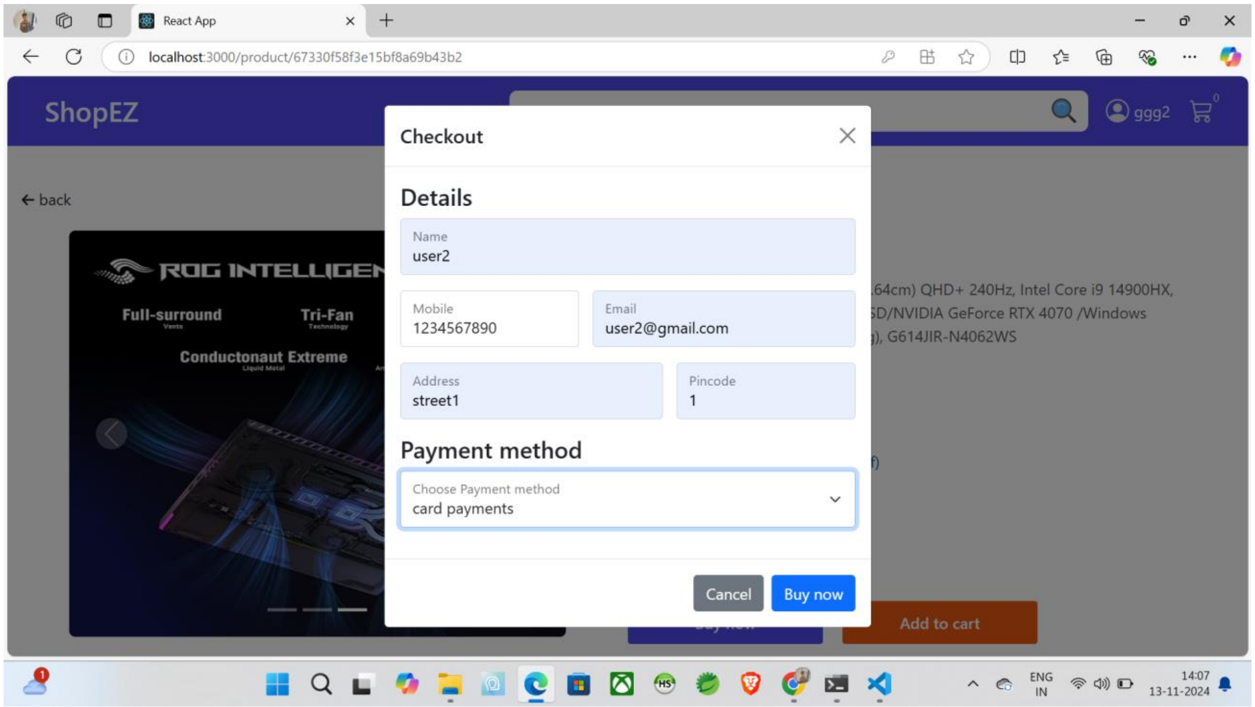
Username
ggg2

Email address

Password
••••••••

User Type
Customer

Already registered? Login



Checkout

Details

Name
user2

Mobile
1234567890

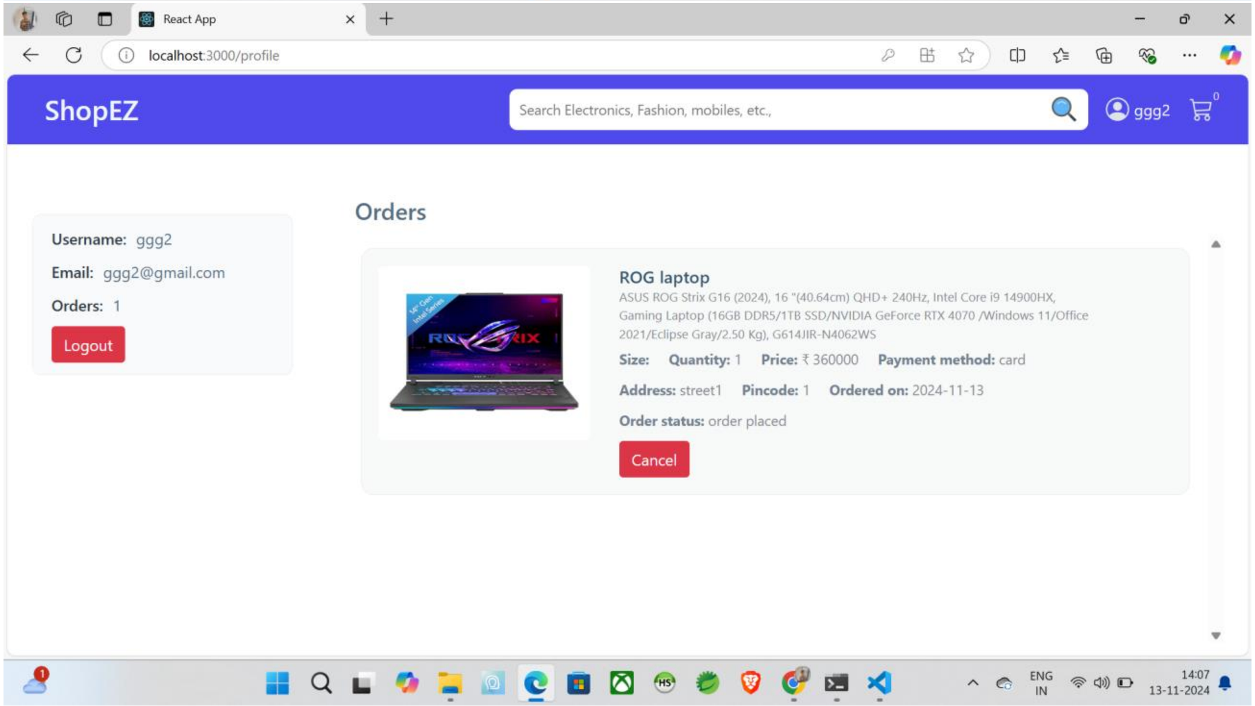
Email
user2@gmail.com

Address
street1

Pincode
1

Payment method

Choose Payment method
card payments



Username: ggg2

Email: ggg2@gmail.com

Orders: 1

Orders

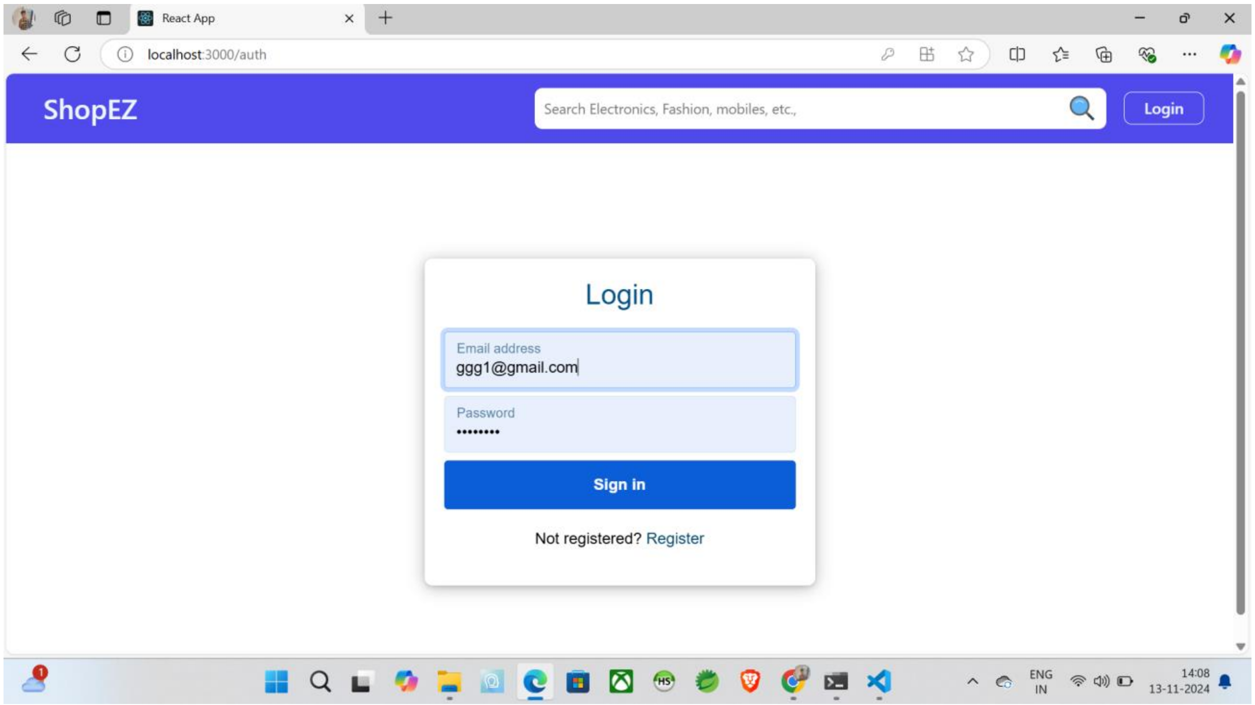
ROG laptop

ASUS ROG Strix G16 (2024), 16 "(40.64cm) QHD+ 240Hz, Intel Core i9 14900HX, Gaming Laptop (16GB DDR5/1TB SSD/NVIDIA GeForce RTX 4070 /Windows 11/Office 2021/Eclipse Gray/2.50 Kg), G614JIR-N4062WS

Size: **Quantity:** 1 **Price:** ₹ 360000 **Payment method:** card

Address: street1 **Pincode:** 1 **Ordered on:** 2024-11-13

Order status: order placed

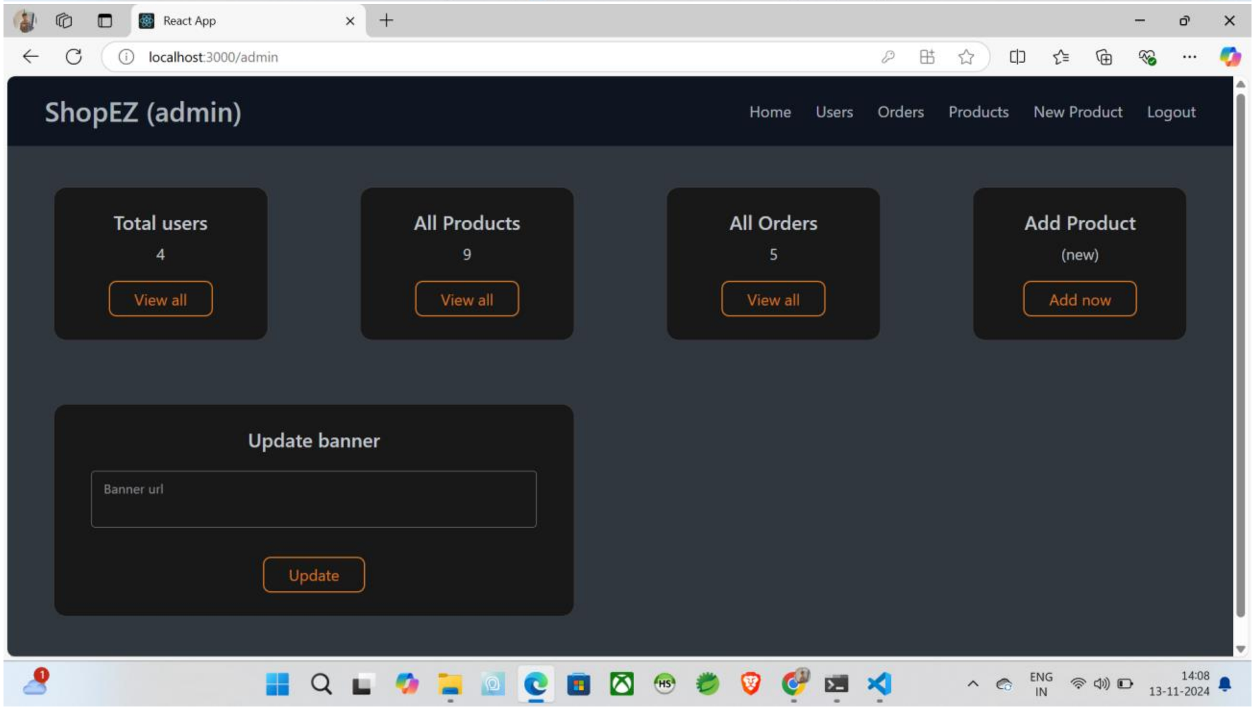


Login

Email address
ggg1@gmail.com

Password
••••••••

Not registered? Register



Total users
4

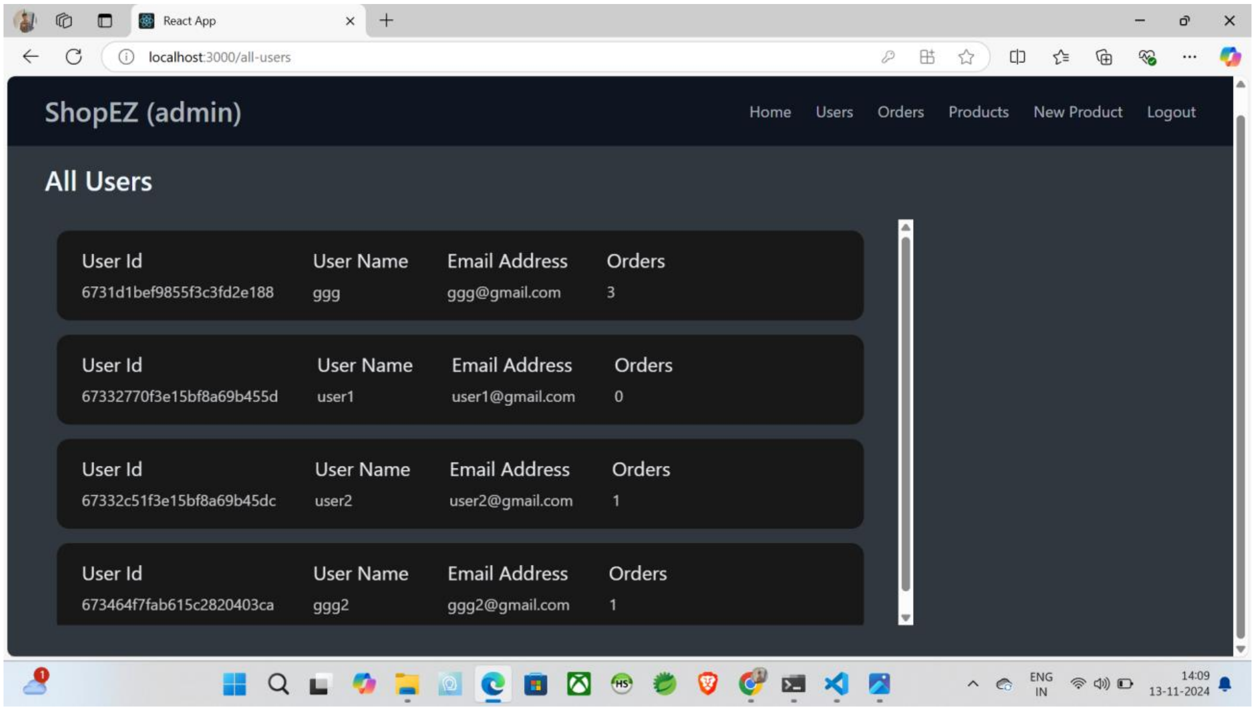
All Products
9

All Orders
5

Add Product
(new)

Update banner

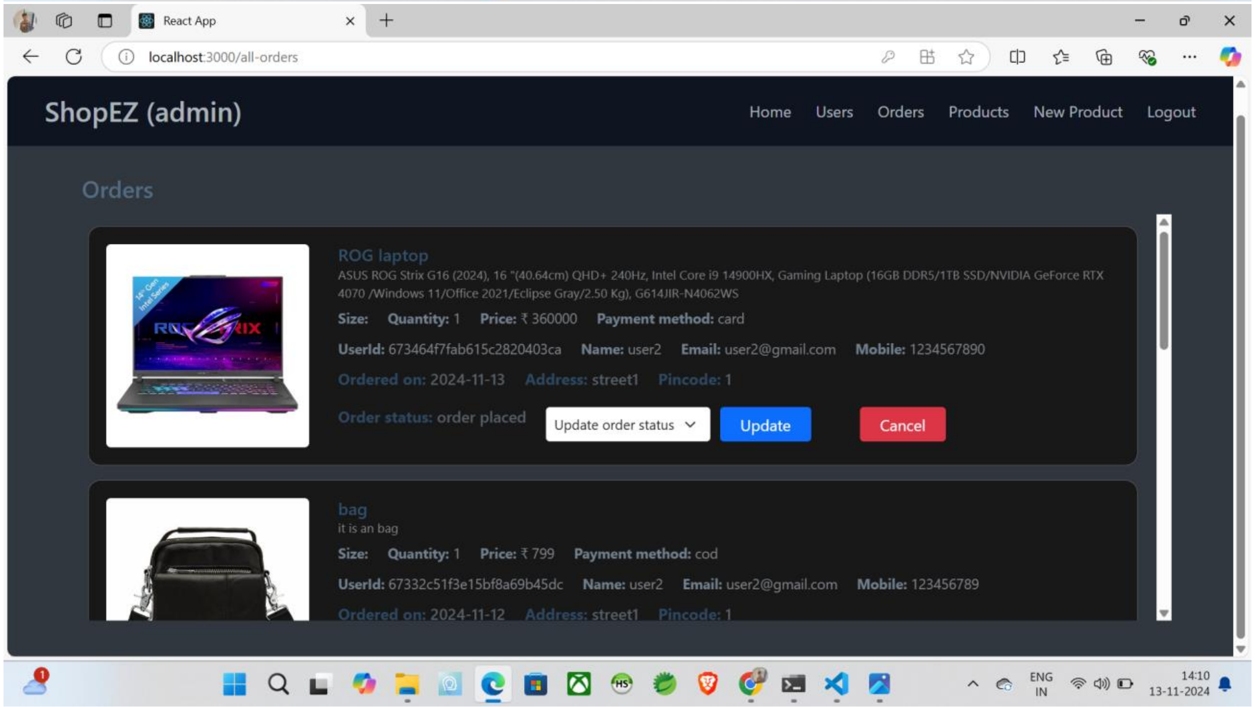
Banner url



ShopEZ (admin)

All Users

User Id	User Name	Email Address	Orders
6731d1bef9855f3c3fd2e188	ggg	ggg@gmail.com	3
67332770f3e15bf8a69b455d	user1	user1@gmail.com	0
67332c51f3e15bf8a69b45dc	user2	user2@gmail.com	1
673464f7fab615c2820403ca	ggg2	ggg2@gmail.com	1



ShopEZ (admin)

Orders

ROG laptop

ASUS ROG Strix G16 (2024), 16 "(40.64cm) QHD+ 240Hz, Intel Core i9 14900HX, Gaming Laptop (16GB DDR5/1TB SSD/NVIDIA GeForce RTX 4070 /Windows 11/Office 2021/Eclipse Gray/2.50 Kg), G614JIR-N4062WS

Size: **Quantity:** 1 **Price:** ₹ 360000 **Payment method:** card

UserId: 673464f7fab615c2820403ca **Name:** user2 **Email:** user2@gmail.com **Mobile:** 1234567890

Ordered on: 2024-11-13 **Address:** street1 **Pincode:** 1

Order status: order placed

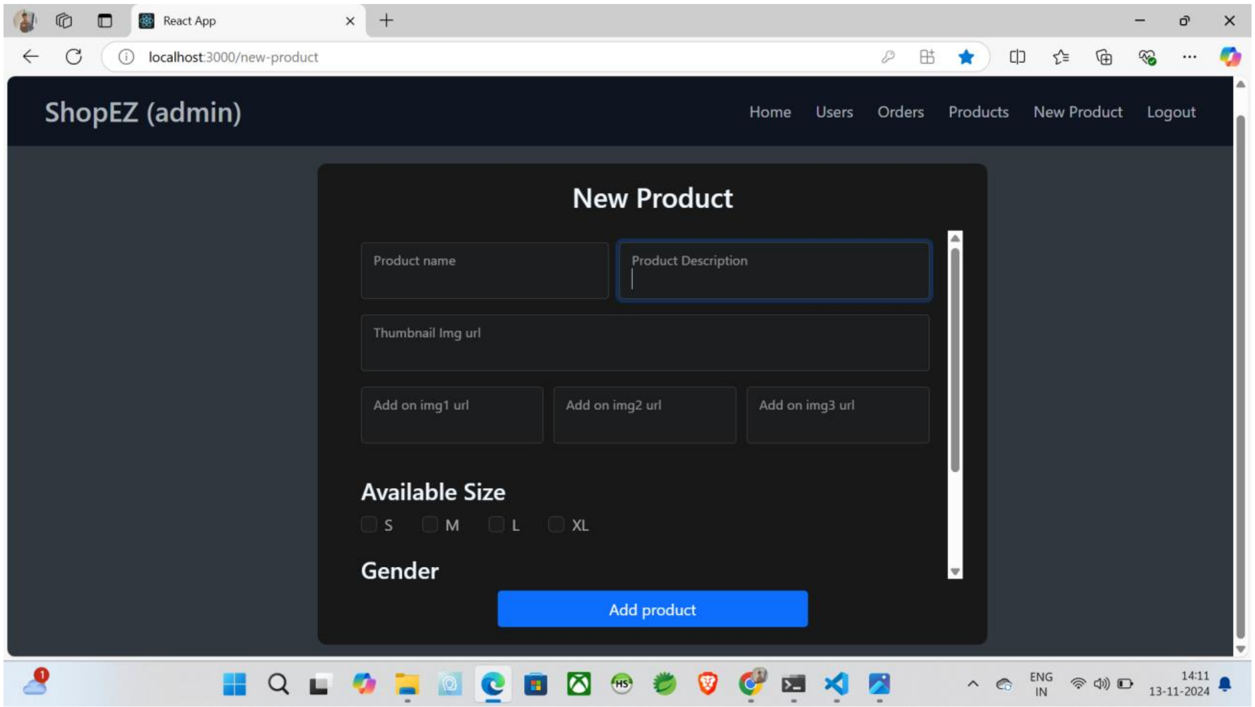
bag

it is an bag

Size: **Quantity:** 1 **Price:** ₹ 799 **Payment method:** cod

UserId: 67332c51f3e15bf8a69b45dc **Name:** user2 **Email:** user2@gmail.com **Mobile:** 123456789

Ordered on: 2024-11-12 **Address:** street1 **Pincode:** 1



New Product

Product name

Product Description

Thumbnail Img url

Add on img1 url

Add on img2 url

Add on img3 url

Available Size

□ S □ M □ L □ XL

Gender