Name : Tanushree More

Email ID : [tanushreemore39@gmail.com](mailto:tanushreemore39@gmail.com)

Batch : 1st September, 2023

GitHub : <https://github.com/TanushreeMore>

LinkedIn : <https://www.linkedin.com/in/tanushree-more>

Date : 10-02-2024

**e-Commerce Project**

As per the given assignment of creating an e-commerce website using MongoDB, Express JS, React JS, and Node JS, I gave the necessary comments.

1. Added Node, Express, MongoDB, bcrypt, cors, jsonwebtoken, and razorpay packages for the backend of this assignment.
2. Added CSS, bootstrap, material UI, react-dom, react-router-dom, react-redux, react-paypal-js, jquery, axios packages for the front-end of this assignment.
3. Tried to create a full-featured responsive e-commerce application using React and Node.
4. Created common components such as
   1. Header
   2. Footer
   3. Slider
   4. Navigation
5. Added logo, search box, login button, and cart in the header.
6. Added home, admin dashboard, productList and products on navigation bar.
7. Added background image with the logo and slogan on the home page. Added slider effect on for products on the home page.
8. Added card format to display multiple products on the product list.
   1. Gave hovering effect on cards.
   2. Fetched data from the database.
   3. Used pagination for showing multiple pages on the bottom of the page.
   4. Moved to product detailing when click on the shop now button.
9. Made the details of the products visible. The user can choose the size of the product before adding it to the cart. Also added rating and reviews in a static form.
10. Added quantity increment decrement button on cart for products. Delete, update and like icon shown here. The delete function is working properly. The price of the product changes as the quantity of the product changes.
11. Added price details with a checkout button at the bottom. It moves to the stepper’s step-3.(if the user has logged in). If not, it moves to the step-1 and suggests the user to either log in or create an account.
12. Implemented stepper’s step-2 is for delivery address. Here user can add address where they want the product to be delivered.
13. Step-3 is for the order summary. Added payment button afterwards.
14. Designed payment button which redirects to the RazorPay’s gateway, which is currently in test mode. It shows payment done successfully for now as a static information.
15. Inserted an admin dashboard link on the navbar such that in the admin dashboard, products and orders can be added, deleted, or their status changed.
16. Created dashboard menu using drawer.
17. Shown order and product tables on the dashboard.
18. Listed products.
19. Added product images.
20. Rendered and styled products.
21. Added page routing.
22. Created a Node js server.
23. Fetched products from backend.
24. Managed state using the reducer hook.
25. Added the bootstrap UI framework.
26. Created product and rating components.
27. Created product details screen.
28. Created loading and messages.
29. Added item to the cart.
30. Created cart screen.
31. Created sign-in screen.
32. Connected to the MongoDB database.
33. Created seed samples for products and users.
34. Created sign-in, sign-up as well as other API’s.
35. Implemented a shipping screen.
36. Created a place order screen.
37. Created order screen.
38. Paid orders using RazorPay.
39. Created a profile screen.
40. Added a sidebar for admin dashboard and a common search box.
41. Created an admin menu and dashboard.
42. Created product and order management.
43. Tried to create account on Heroku for publishing the application. But account is not generating.