**General Instructions for the Online Store Project**

**Objective:**  
The objective of this project is to build an online store with four main pages. You will define the products yourself, implement navigation between the pages, and manage the shopping cart and checkout process. You are allowed to use Bootstrap for styling.

**Requirements:**

1. **Home Page:**
   * The home page should display details about the store, such as the store name, tagline, and general information about the store.
   * Include a navigation bar (Navbar) at the top of the page, with links to other pages (Home, Products, Cart, Checkout).
2. **Products Page:**
   * Display the products in different categories, with an image, product name, price, and a short description.
   * The products will be defined in a React state, including at least the following details: image, name, price, and description.
   * Each product should have a button to add it to the shopping cart.
   * When a product is clicked, a modal should open showing full details about the product (such as a larger image, full description, and additional price details).
3. **Shopping Cart Page:**
   * Show the products that the user has added to their cart.
   * Display the name of each product, its price, and the quantity selected.
   * Calculate the total price for the items in the cart.
   * Allow the user to remove items from the cart and change the quantity of items.
4. **Checkout Page:**
   * Include a form for payment details such as name, address, and credit card information.
   * There should be a button to confirm the payment, and upon a successful transaction, a message should appear (this can be a modal or an alert).
5. **Navbar:**
   * The Navbar should be present on all pages and include links/buttons for navigation to Home, Products, Cart, and Checkout.

**Design Requirements:**

* **Bootstrap**: Use **Bootstrap** to style the pages and components effectively and responsively.
* The design should be clean, modern, and user-friendly.

### Important Notes:

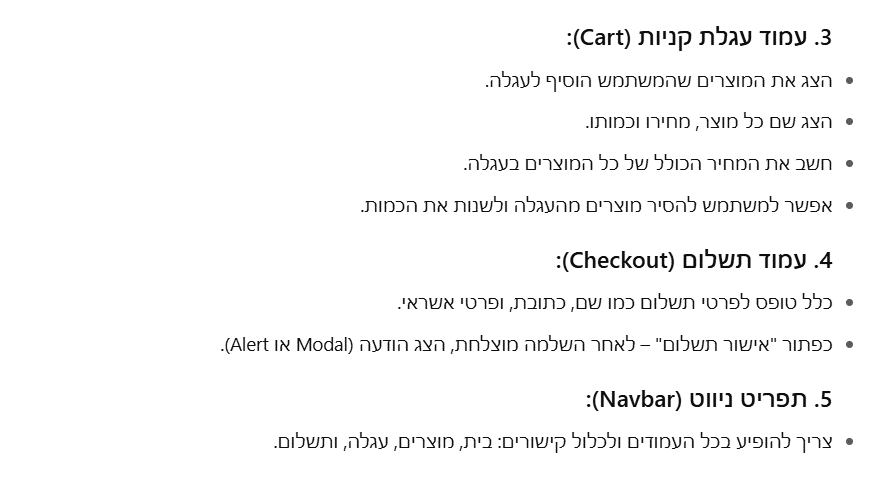
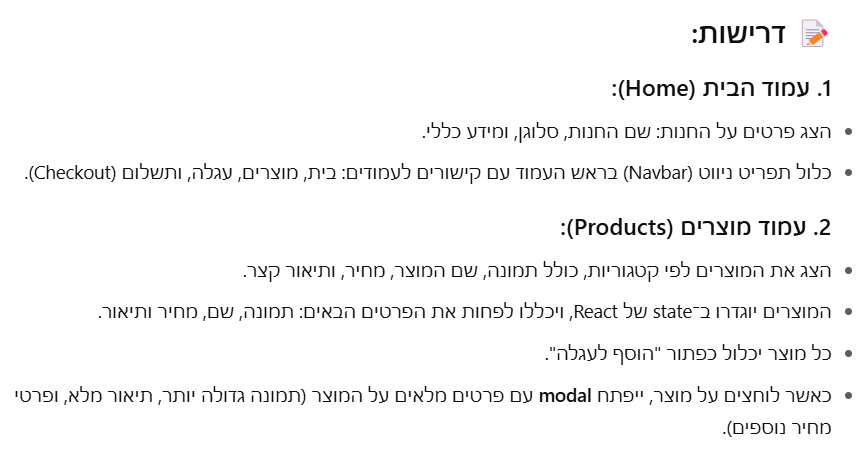
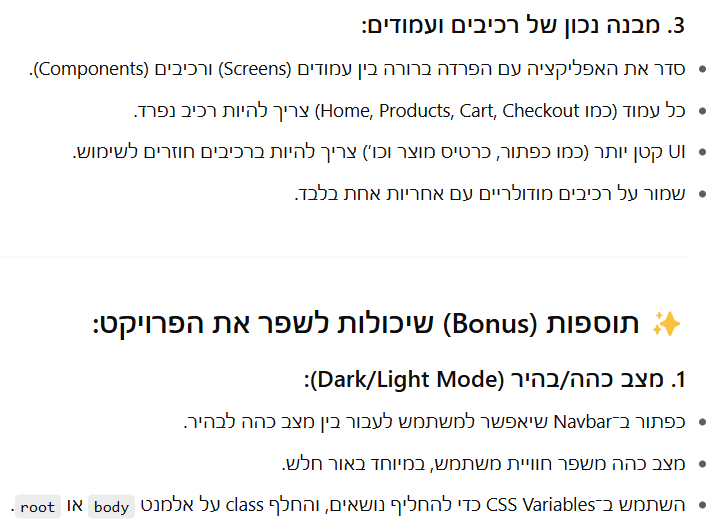
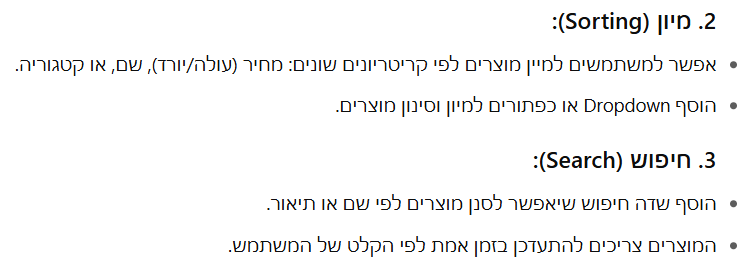
1. **Use Redux for Global State Management:**
   * You are required to use **Redux** to manage the global state of your application, including the shopping cart, user information, and other shared data between components.
   * Make sure to set up a Redux store, actions, and reducers correctly to handle the state flow.
2. **Code Consistency and Readability:**
   * Follow consistent naming conventions and coding practices throughout the project.
   * Ensure that your code is readable, well-structured, and well-documented.
   * Write clear and concise comments where necessary to explain complex or important sections of your code.
3. **Proper Component and Screen Structure:**
   * Make sure your application is well-organized with clear separation between screens and components.
   * Each screen (e.g., Home, Products, Cart, Checkout) should be represented by its own component, and individual UI elements should be placed in reusable components when appropriate.
   * Keep your components modular and focused on a single responsibility.

### Bonus Features for Your Project:

1. **Dark/Light Mode Toggle:**
   * Add a button in the Navbar that allows users to toggle between **dark mode** and **light mode**.
   * Implementing dark mode can improve the user experience, especially in low-light environments.
   * Use CSS variables to easily switch between themes, and toggle the class on the root element (e.g., body).
2. **Sorting:**
   * Allow users to sort products by different criteria, such as price (ascending/descending), name, or category.
   * Add a dropdown or buttons to filter and sort the products displayed.
3. **Search Functionality:**
   * Add a search bar that allows users to filter products by name or description.
   * As users type in the search field, products should be filtered in real-time.



Good luck with your project!

/src

├── /components # קומפוננטות קטנות שחוזרות במסכים

│ ├── Navbar.jsx

│ ├── ProductCard.jsx

│ ├── CartItem.jsx

│ └── CheckoutForm.jsx

│

├── /pages # מסכים שלמים (Pages)

│ ├── HomePage.jsx

│ ├── ProductsPage.jsx

│ ├── CartPage.jsx

│ └── CheckoutPage.jsx

│

├── /slices # קבצי Redux Slice

│ └── cartSlice.js

│

├── /store # הגדרת ה-Store של Redux

│ └── store.js

│

├── App.jsx # ניהול כל הראוטים (דפים) והניווט

├── main.jsx # כניסה לפרויקט (מייצר את ה-App)

└── index.css # עיצוב כללי