Phase 3: Shopping Cart and Checkout

5. Shopping Cart Functionality

   - Create a shopping cart model and API.

   - Allow users to add, update, and remove products from the cart.

   - Store cart data in the database and sync with the user's session.

   Details:

   - **Cart Schema:**

     - userId: ObjectId, required, ref('User')

     - products: [{

         productId: ObjectId, required, ref('Product')

         quantity: Number, required

       }]

   - **API Routes:**

     - POST /api/cart: Add product to user's cart

     - PUT /api/cart/:id: Update product quantity in cart

     - DELETE /api/cart/:id: Remove product from user's cart

     - GET /api/cart: Retrieve user's cart items

   - **Frontend:**

     - Create a shopping cart page in Next.js to display cart items.

     - Implement add to cart, update quantity, and remove item functionalities using React state and hooks.

- Sync cart state with backend using API calls.

6. Checkout Process

   - Integrate Stripe for secure payment processing.

   - Implement a checkout page where users can review their cart and enter payment details.

   - Handle order creation and payment confirmation.

   Details:

   - **Order Schema:**

     - userId: ObjectId, required, ref('User')

     - products: [{

        productId: ObjectId, required, ref('Product')

        quantity: Number, required

      }]

     - totalAmount: Number, required

     - paymentStatus: String, default: 'Pending'

   - **API Routes:**

     - POST /api/orders: Process payment and create order

     - GET /api/orders: Retrieve user's order history

   - **Frontend:**

     - Create a checkout page in Next.js to handle payment details.

     - Integrate Stripe's checkout component for payment processing.

     - Confirm payment and display order summary.

Phase 4: Reviews and Ratings

7. Reviews and Ratings

   - Implement a review model and API.

   - Allow users to leave reviews and ratings for products.

   - Display average ratings and user reviews on product pages.

   Details:

   - **Review Schema:**

     - userId: ObjectId, required, ref('User')

     - productId: ObjectId, required, ref('Product')

     - rating: Number, required, min: 1, max: 5

     - comment: String

   - **API Routes:**

     - POST /api/reviews: Add a review for a product

     - GET /api/reviews/:productId: Retrieve reviews for a product

     - PUT /api/reviews/:id: Update a review

     - DELETE /api/reviews/:id: Delete a review

   - **Frontend:**

     - Create a review section on the product detail page in Next.js.

     - Implement form for users to submit reviews and ratings.

     - Display reviews and average rating for each product.

Phase 5: User and Seller Dashboards

8. User Dashboard

  - Develop a dashboard for users to view their purchase history and order details.

  - Implement a wishlist feature for users to save products for later.

  Details:

  - **Dashboard Components:**

    - Order history: Display list of past orders with order details.

    - Wishlist: Allow users to add/remove products from their wishlist.

  - **API Routes:**

    - GET /api/users/orders: Retrieve user's order history

    - GET /api/users/wishlist: Retrieve user's wishlist

    - POST /api/users/wishlist: Add product to wishlist

    - DELETE /api/users/wishlist/:productId: Remove product from wishlist

  - **Frontend:**

    - Create a user dashboard page in Next.js.

    - Implement components to display order history and wishlist.

    - Use React state and hooks to manage wishlist and order data.

9. Seller Dashboard

  - Allow sellers to view and manage their product listings.

  - Implement sales analytics to track product performance.

EcoShop - Sustainable Product Marketplace

Details:

- **Dashboard Components:**

  - Product management: Create, update, and delete product listings.

  - Sales analytics: Display sales data and product performance metrics.

- **API Routes:**

  - GET /api/sellers/products: Retrieve seller's product listings

  - POST /api/sellers/products: Create a new product listing

  - PUT /api/sellers/products/:id: Update a product listing

  - DELETE /api/sellers/products/:id: Delete a product listing

- **Frontend:**

  - Create a seller dashboard page in Next.js.

  - Implement components for product management and sales analytics.

  - Use charts and graphs to visualize sales data.