Governor Sindh Initiative AI Course

Name:Mehwish Sheikh

Roll number:223558

HACKATHON DAY 4

Building_Dynamic_Frontend_Components_for_Your_Marketplace

GENERAL E-COMMERCE COMFORTY WEBSITE

Project Overview

The **Marketplace Builder** is an e-commerce platform designed to allow users to create and manage their own online marketplaces. It provides essential features for product management, user interactions, and order processing, including product listings, inventory management, a shopping cart, wishlist functionality, order confirmation, and notifications.

Key Features

1. Adding Products to the Wishlist

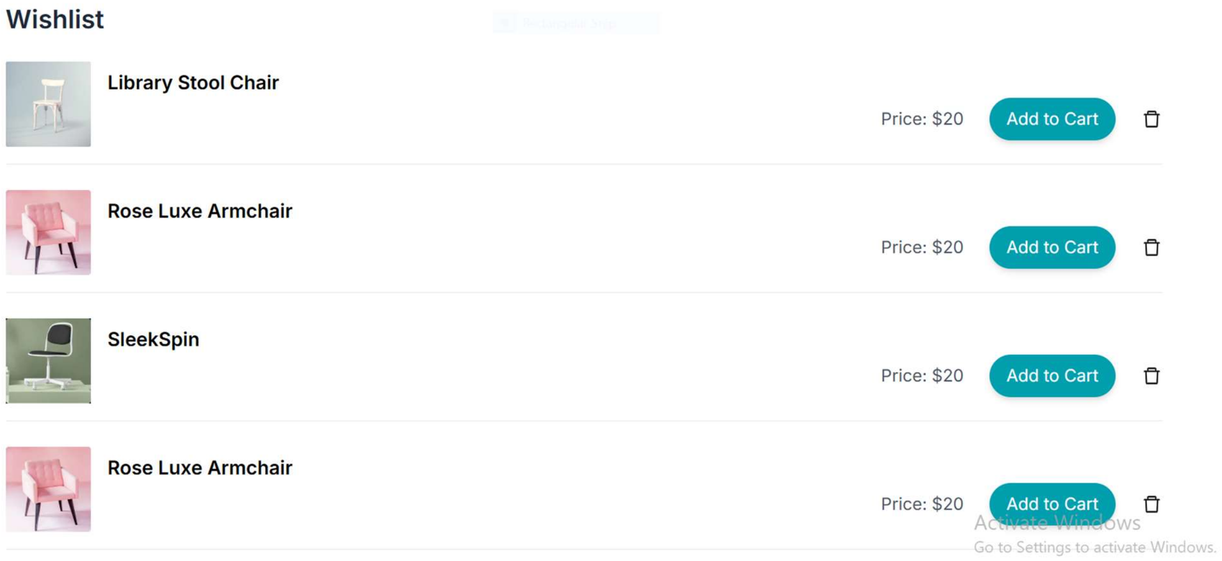
- Users can add products to their wishlist for future purchase.
- This feature allows users to save products they are interested in, helping them easily find and purchase these products later.

Implementation Steps:

- A "Wishlist" button is added to each product in the product listing.
- When a user clicks the button, the product is saved to their wishlist (typically stored in the database or local storage).
- A notification or a visual indicator (e.g., a heart icon) appears when the product is added to the wishlist.
- Users can view their wishlist from a dedicated page or sidebar.

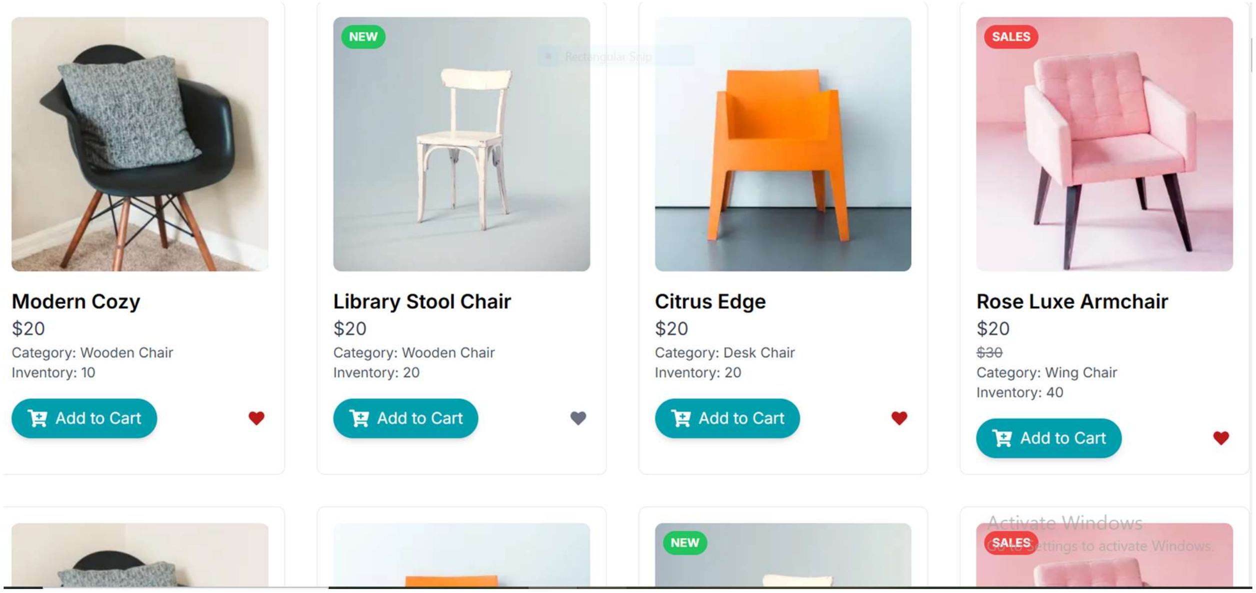
UI/UX Considerations:

- A heart icon or "Add to Wishlist" button is placed near the product image or description.
- A dedicated page or modal displays the wishlist with the option to remove items or add them to the cart.



Wishlist

Library Stool Chair	Price: $20	Add to Cart	🗑
Rose Luxe Armchair	Price: $20	Add to Cart	🗑
SleekSpin	Price: $20	Add to Cart	🗑
Rose Luxe Armchair	Price: $20	Add to Cart	🗑



Modern Cozy
$20
Category: Wooden Chair
Inventory: 10
Add to Cart ♥

NEW
Library Stool Chair
$20
Category: Wooden Chair
Inventory: 20
Add to Cart ♥

Citrus Edge
$20
Category: Desk Chair
Inventory: 20
Add to Cart ♥

SALES
Rose Luxe Armchair
$20
~~$30~~
Category: Wing Chair
Inventory: 40
Add to Cart ♥

NEW

SALES

2. Adding Products to the Cart

- Users can add products to their shopping cart to proceed with the checkout process.
- The cart allows users to review their selected products, adjust quantities, and proceed to checkout.

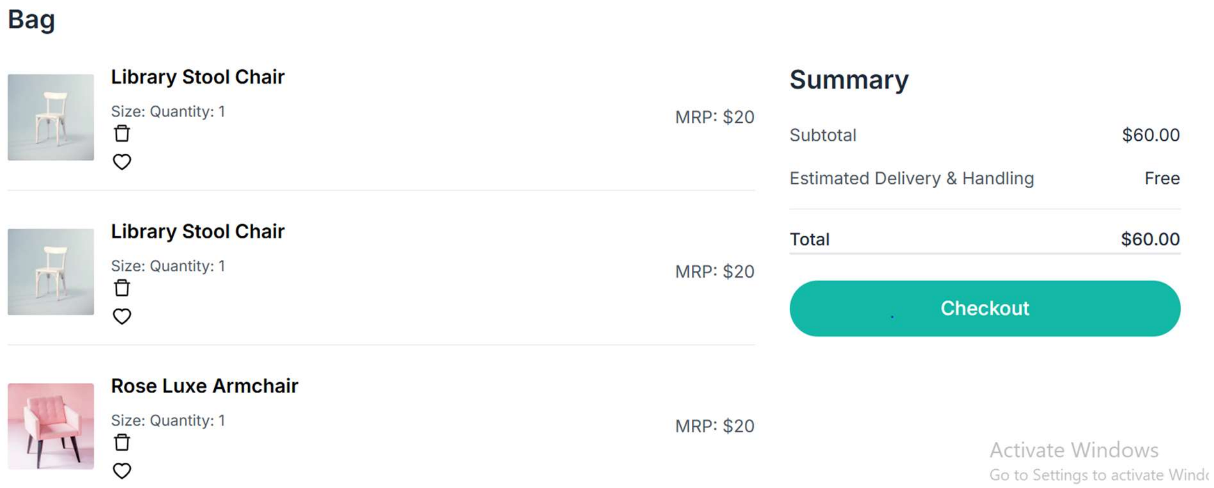
Implementation Steps:

- A "Add to Cart" button is added to each product in the product listing.
- When clicked, the product is added to the user's cart (stored in the database or session storage).
- The cart is updated in real-time, and the user can view the number of items in the cart through an icon or notification.

- The cart page allows users to modify quantities, remove items, or proceed to checkout.

UI/UX Considerations:

- The cart icon is displayed at the top of the page with a counter showing the number of items in the cart.
- A cart page or dropdown displays all added products, their prices, quantities, and a checkout button.



Bag

Library Stool Chair
Size: Quantity: 1
🗑
♡
MRP: $20

Library Stool Chair
Size: Quantity: 1
🗑
♡
MRP: $20

Rose Luxe Armchair
Size: Quantity: 1
🗑
♡
MRP: $20

Summary

Subtotal	$60.00
Estimated Delivery & Handling	Free
Total	$60.00

Checkout

Managing Inventory

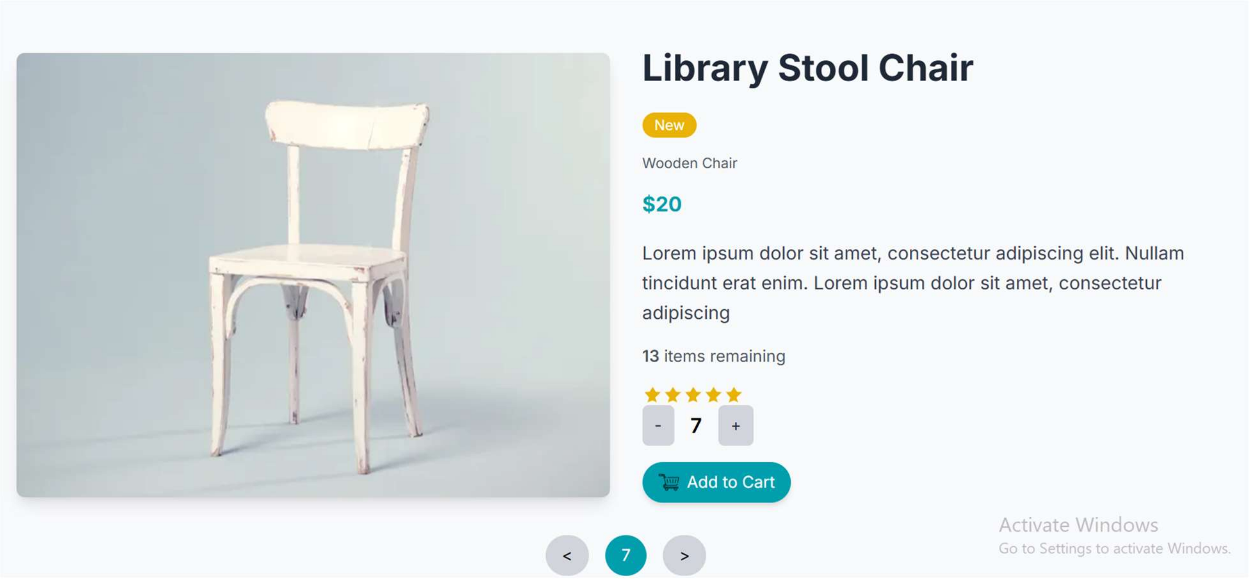
- Sellers can manage their product inventory to ensure they have the correct stock levels for each product.
- Inventory management includes adding new stock, updating product quantities, and removing products when they are out of stock.

Implementation Steps:

- Admins or sellers can access an "Inventory Management" dashboard.
- Sellers can add or update the stock for each product by specifying the quantity available.
- Products with low stock can be highlighted or flagged for restocking.
- When an item is added to the cart, the inventory is updated in real-time.
- If a product's inventory reaches zero, it will be marked as "Out of Stock" and unavailable for purchase.

UI/UX Considerations:

- A stock input field is available on the product management page to update quantities.
- Admins or sellers are notified when stock levels are low or when items are out of stock.



Library Stool Chair

New

Wooden Chair

$20

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam tincidunt erat enim. Lorem ipsum dolor sit amet, consectetur adipiscing

13 items remaining

4. Product Listing

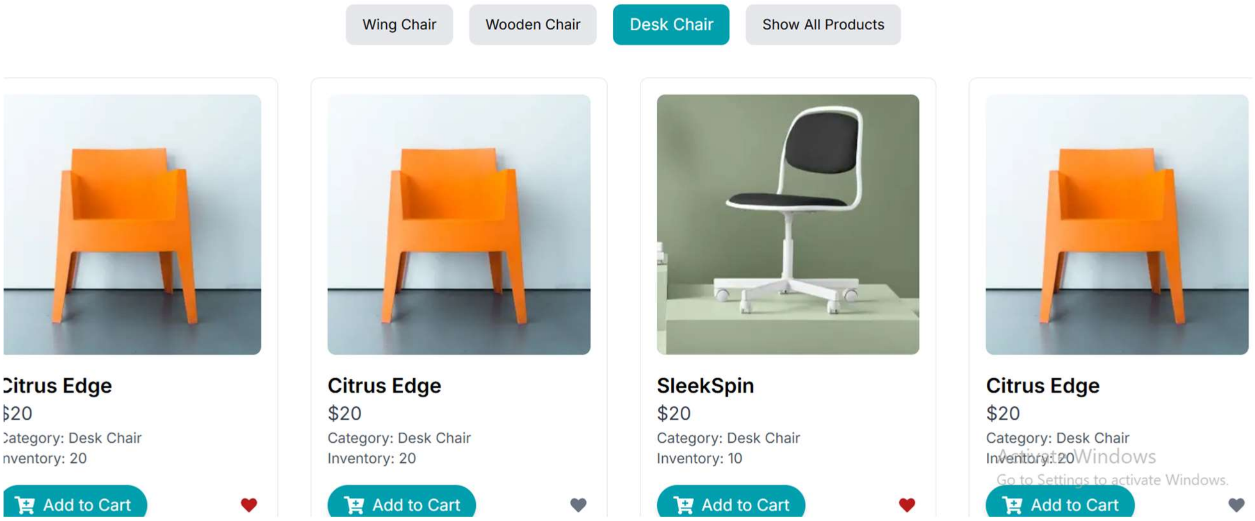
- Sellers can create product listings to showcase their products on the marketplace.
- A product listing includes product details such as name, price, description, images, and category.

Implementation Steps:

- A "Create Product" form is available for sellers to input product details.
- The form includes fields for product name, price, description, images (uploaded via drag-and-drop or file input), and product category.
- Products are categorized and displayed on the marketplace for customers to browse.
- Admins can review and approve product listings before they are published.

UI/UX Considerations:

- A clean and user-friendly product creation form with clear instructions.
- Product listings should include high-quality images, detailed descriptions, and clear pricing.
- Filters and search functionality are implemented to allow customers to find products based on category, price, and other attributes.

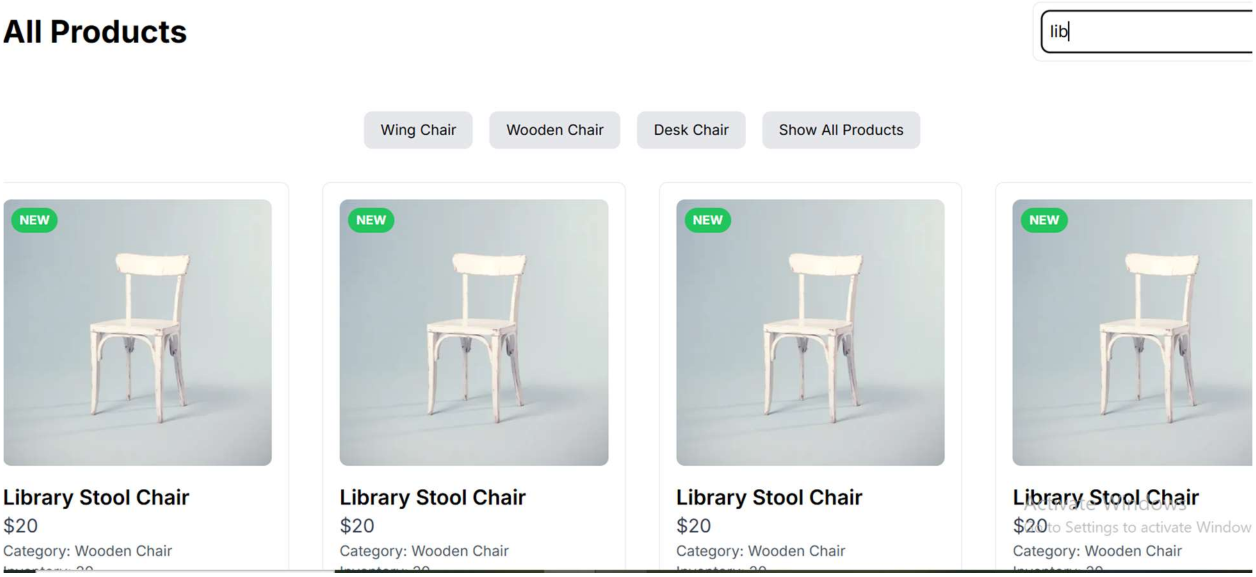


itrus Edge
20
ategory: Desk Chair
nventory: 20

Citrus Edge
$20
Category: Desk Chair
Inventory: 20

SleekSpin
$20
Category: Desk Chair
Inventory: 10

Citrus Edge
$20
Category: Desk Chair
Inventory: 20



All Products

lib

Library Stool Chair
$20
Category: Wooden Chair
Inventory: 20

Library Stool Chair
$20
Category: Wooden Chair
Inventory: 20

Library Stool Chair
$20
Category: Wooden Chair
Inventory: 20

Library Stool Chair
$20
Category: Wooden Chair
Inventory: 20

5. Order Confirmation and Placing Details

- Once a user completes their purchase, they are presented with an order confirmation page.
- The order confirmation includes details such as the products purchased, total price, shipping address, and payment method.

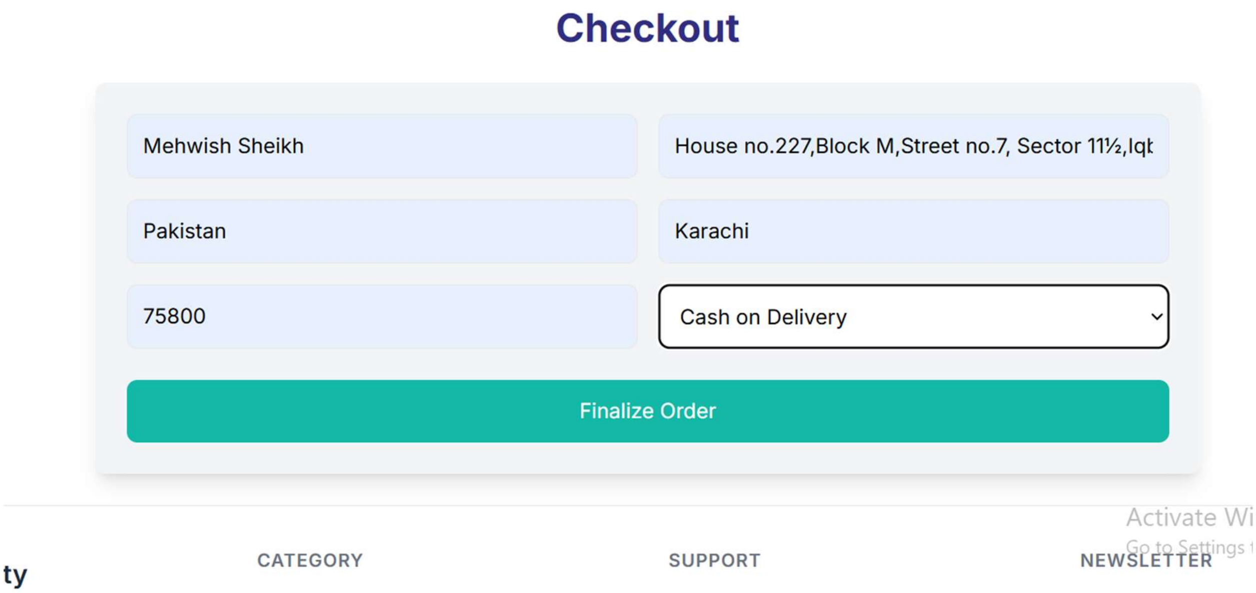
Implementation Steps:

- After checkout, users are directed to an order confirmation page.
- The page displays a summary of the order, including product names, quantities, prices, and the total amount.
- Users are prompted to provide shipping details (address, contact information).
- Payment processing is integrated into the checkout process (e.g., via Stripe, PayPal, etc.).

- Upon successful payment, an order confirmation email is sent to the user, and the order is marked as "Completed."

UI/UX Considerations:

- A clear summary of the order, with the option to print or save the order details.
- A progress bar or steps to guide the user through the checkout process (e.g., Cart → Shipping → Payment → Confirmation).
- A confirmation message thanking the user for their purchase, along with an estimated delivery time.



Checkout

Mehwish Sheikh

House no.227,Block M,Street no.7, Sector 11½,Iqt

Pakistan

Karachi

75800

Cash on Delivery

Notification Feature

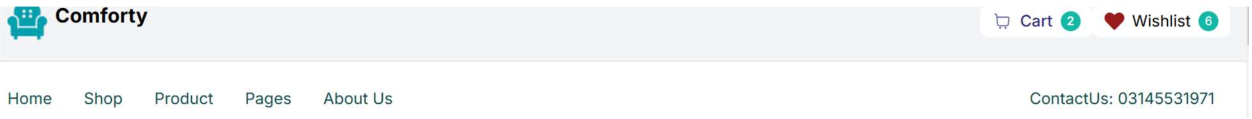
- Notifications are provided to users for various events such as adding products to the cart, wishlist, or completing an order.
- Notifications can be displayed as toast messages or modal popups to keep users informed about their actions.

Implementation Steps:

- Notifications are triggered when a user adds a product to the cart, wishlist, or completes a purchase.
- A notification library (such as **react-toastify** or custom-built) is used to display toast messages at the top or bottom of the screen.
- For important actions (e.g., order confirmation), modal popups are used to provide a more detailed view of the event.
- Notifications can also be used for system messages, such as low inventory warnings or successful product addition.

UI/UX Considerations:

- Toast notifications are displayed for quick actions (e.g., "Added to Cart").
- Modal popups are used for critical actions like order confirmations, with clear call-to-action buttons (e.g., "View Order").
- Notifications are styled using **Tailwind CSS** to match the overall design of the platform.



7. Product Ratings

- Users can rate products they have purchased, providing valuable feedback for other buyers and sellers.
- Ratings are displayed as stars (out of 5) and can be accompanied by a written review.

Implementation Steps:

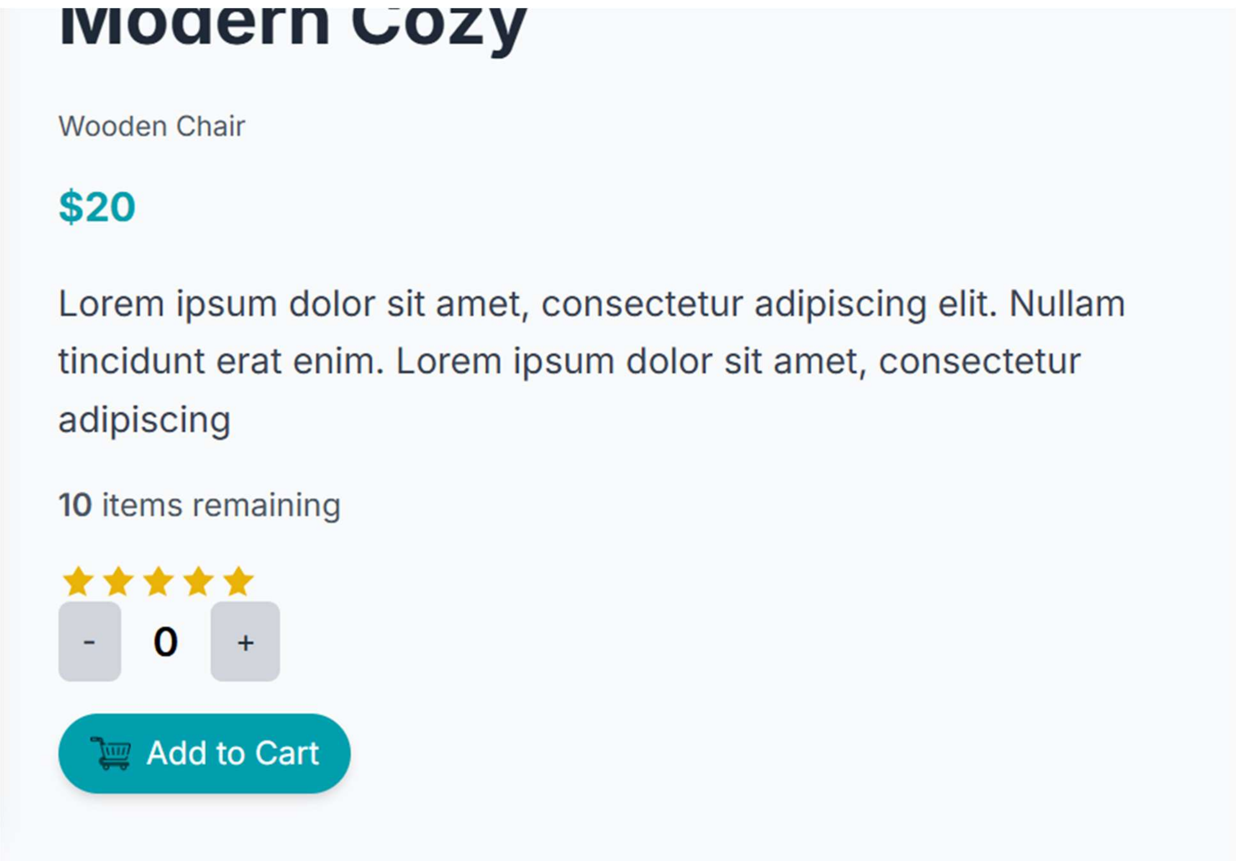
- Users can submit a rating for a product on the product detail page after completing their purchase.
- A "Rate this Product" section is added below the product description where users can select a rating (1-5 stars) and optionally leave a written review.
- The rating is stored in the database and associated with the product.
- The average rating is displayed on the product listing and product detail page.
- Reviews can be moderated or flagged by the admin to ensure quality content.

Backend Implementation:

- A product rating model is created to store ratings and reviews in the database.
- The model includes fields for user ID, product ID, rating (1-5 stars), and review text.
- The average rating is calculated by dividing the total sum of ratings by the number of ratings.
- Admins can manage reviews, including deleting inappropriate reviews or marking them as helpful.

UI/UX Considerations:

- A star rating system is implemented using **Tailwind CSS** for easy styling.
- The average rating is displayed on the product detail page, along with the number of reviews.
- A review section allows users to leave written feedback, with a clear call-to-action (e.g., "Submit Review").
- A "Sort by Rating" option is provided for users to filter products based on ratings.



Modern Cozy

Wooden Chair

$20

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam tincidunt erat enim. Lorem ipsum dolor sit amet, consectetur adipiscing

10 items remaining

★★★★★

Technical Details

Frontend:

- **React** is used for building the user interface, leveraging **Next.js** for server-side rendering and routing.
- **Tailwind CSS** is used for styling, ensuring a responsive and modern design.
- **React Context** is used to manage the cart and wishlist state globally across the app.
- **Sanity CMS** is used to manage product data, including product listings, images, descriptions, inventory, and ratings.

Backend:

- **Sanity** serves as the headless CMS for managing product data, including inventory, product descriptions, images, and ratings.
- **Next.js API routes** are used to handle interactions with the backend, including adding products to the cart, managing inventory, processing orders, and submitting ratings.
- **Node.js** is used to manage server-side logic and API endpoints.

Conclusion

The **Marketplace Builder** is a robust platform that allows sellers to list products, manage inventory, and interact with customers through features like the shopping cart, wishlist, order confirmation, notifications, product ratings, and more. By integrating key features such as inventory management, payment processing, real-time notifications, and product reviews, the platform offers a seamless experience for both sellers and buyers.