Day 5 Progress

Test Project

Test Page 1

I tested my project using Postman. When I hit the URL http://localhost:3000/Shop and fetch all data from the API, the products are displayed within 2 seconds. If no products are found, a message appears prompting me to add products.

I also tested the search functionality, and it works correctly. The filters are functioning as expected, allowing me to narrow down the displayed products based on the criteria I set.

Additionally, I verified that the page displays exactly 9 products per page as intended. Everything is working properly.

Testing Summary Table

Test Case ID	Description	Input/Action	Expected Result	Actual Result	Status
TC001	Test API response time	Hit http://localhost:3000/Shop to fetch products	Products should display within 2 seconds	Products displayed in 2 seconds	Passed
TC002	Handle no products	Fetch products when none exist	Show "No products found" with "Add Product" option	Proper fallback message displayed	Passed
TC003	Test search functionality	Search for a specific product ("Pizza")	Display products matching the search term	Relevant products displayed	Passed
TC004	Test filters	Apply category and price filters	Display products based on the selected filters	Products displayed correctly	Passed
TC005	Verify pagination	Check that only 9 products are displayed per page	9 products displayed on the page	9 products displayed	Passed

Test page 2

I tested the **Page 2** functionality, including the "Add to Cart" page. Everything works perfectly as expected. When I click on any food image, it navigates to a new page that dynamically displays the details of the selected food item.

I verified the following:

- The dynamic routes function correctly, showing the specific product page corresponding to the selected food image.
- The product details page includes the food's description and two buttons (+ and -) to adjust the quantity of the item.
- Clicking the **"Add to Cart"** button triggers an alert confirming that the selected food item has been added to the cart.

Testing Summary Table

Test Case ID	Description	Input/Action	Expected Result	Actual Result	Status
TC001	Test dynamic routing	Click on a food image	Navigate to a new page with product details	Navigation and details work fine	Passed
TC002	Verify product details	View the selected product's description and details	Show product name, price, description, and image	All details displayed correctly	Passed
TC003	Test quantity adjustment	Use the (+) and (-) buttons	Adjust the quantity of the food item dynamically	Quantity updates dynamically	Passed
TC004	Test "Add to Cart" button	Click the **Add to Cart** button	Show an alert confirming the item has been added	Alert displayed as expected	Passed

Test page 3

I tested **Page 3** functionality by using the URL http://localhost:3000/Carts in Postman to ensure everything works perfectly. The following were verified:

- The cart displays all food items that I added, along with their respective prices.
- The "Remove" or "Delete" option functions correctly, allowing items to be removed from the cart, and the total price updates dynamically.
- The total price is updated accurately when items are added or removed from the cart.
- Using browser DevTools, I confirmed that all UI updates occur as expected based on the selected actions.
- The HTTP response status is **200 OK** in Postman, indicating the API is functioning correctly.
- The products added to the cart are displayed accurately on the cart page with their actual prices.

This confirms that the cart page's UI and backend are working together seamlessly, dynamically updating according to the user actions.

Testing Summary Table

Test Case ID	Description	Input/Action	Expected Result	Actual Result	Status
TC001	Verify products in the cart	Navigate to /Carts after adding items	Display all products added to the cart	All items displayed correctly	Passed
TC002	Test product removal	Remove a product from the cart	The product is removed, and the total price is updated	Product removed successfully; total updated	Passed
TC003	Test total price update	Add or remove multiple products	Total price adjusts dynamically based on cart contents	Total updates accurately	Passed
TC004	Verify API response	Hit /Carts URL in Postman	Status code **200 OK** with correct cart data	Status 200 OK, data correct	Passed
TC005	Check UI updates	Perform actions (add/remove items) and observe UI	UI updates dynamically based on the selected actions	UI updates properly	Passed

Test page 4

I tested page 4 which includes the "Checkout" button functionality. When I click the "Checkout" button, it navigates to the checkout page (http://localhost:3000/Checkout) and displays the following features:

- The address form works correctly, requiring all fields to be filled in before proceeding.
- The page shows the food items added to the cart, including their quantities and the total price.
- Discounts are applied correctly, and the discounted price is displayed alongside the original price.
- When the discount is removed, the total price reverts to the actual price.
- The UI updates dynamically to reflect the selected options and displays everything accurately in the checkout template.
- I also tested the checkout URL in Postman to ensure the backend is working properly, receiving a status code of **200 OK** and accurate response data.

Testing Summary Table

Test Case ID	Description	Input/Action	Expected Result	Actual Result	Status
TC001	Verify navigation to the checkout page	Click the "Checkout" button	Navigate to /Checkout and display checkout details	Navigation and details correct	Passed
TC002	Test address form validation	Leave address fields blank and submit	Show error for required fields	Validation works as expected	Passed
TC003	Verify cart details in checkout	View the cart details on the checkout page	Display all items with quantities and total price	Cart details displayed properly	Passed
TC004	Test discount application	Apply a discount code	Adjust the total price to reflect the discount	Discount applied successfully	Passed
TC005	Test discount removal	Remove the discount code	Revert the total price to the original amount	Total price updated correctly	Passed

Test Case ID	Description	Input/Action	Expected Result	Actual Result	Status
TC006	Verify API response	Hit /Checkout URL in Postman	Status code **200 OK** with correct cart and price details	Status 200 OK, data correct	Passed
TC007	Check UI updates	Perform actions like adding/removing discount and editing address	UI dynamically updates based on the actions performed	UI updates correctly	Passed

Lastly, I verified that the backend is working properly, and the UI is responsive on both mobile and desktop devices.