Testing Documentation

Project Overview: This document outlines the testing process and results for the project. The goal of the testing was to ensure the correctness, responsiveness, and functionality of the application's key features, including dynamic routing, cart functionality, shipment API integration, and product details. Testing was conducted using industry-standard tools to verify the application's stability and usability.

Test Summary: The following table summarizes the test cases executed during this testing cycle:

Test Case ID	Test Case Description	Test Steps	Expected Result	Actual Result	Status	Severity Level	Remarks
TC001	Validate product listing page	Open product page > Verify products	Products displayed correctly	Products displayed correctly	Passed	Low	No issues found
TC002	Test API error handling	Disconnect API > Refresh page	Show fallback UI with error message	Error message shown	Passed	Medium	Handled gracefully
TC003	Check cart functionality	Add product to cart > Verify cart contents	Cart updates with added product	Cart updates as expected	Passed	High	Works as expected
TC004	Ensure responsiveness on mobile	Resize browser window > Check layout	Layout adjusts properly to screen size	Responsive layout working as intended	Passed	Medium	Test successful

Testing Tools Used:

1. **React Testing Library:** For unit testing React components. We verified the functionality and rendering of individual components to ensure their behavior aligns with the expected outcomes.
2. **Cypress:** For end-to-end testing. We simulated real user interactions to test the overall flow and integration of components within the application.

Areas Tested:

1. **Dynamic Routing:** Verified the correctness of routes and ensured that the application correctly displays dynamic product pages based on the URL structure. Dynamic routing functionality is working as expected.
2. **Cart Functionality:** Ensured the cart updates dynamically when products are added, removed, or updated. The add-to-cart feature is functioning correctly.
3. **Search and Filter Functionality:** Tested the search bar and filter buttons to confirm they retrieve and display relevant products efficiently. Both features are working excellently.
4. **Shipment API:** Validated the API integration by simulating different responses (e.g., successful, failed, and delayed) and ensured the fallback UI handles errors gracefully. API fetches are handled well using try-catch blocks, with appropriate error messages such as "No products found" or "Loading..." displayed when needed.
5. **Responsiveness:** Checked responsiveness on different screen sizes and devices to guarantee a consistent user experience.
6. **Product Details Page:** Confirmed all product details are displayed correctly, including images, descriptions, and pricing. Product details are shown accurately and align with the expected design.

Work in Progress: This document is a draft and will be updated as new features and functionalities are added to the project. Testing for additional components and workflows is currently underway. The final version of this document will be released once all testing phases are completed successfully.

Conclusion: All key features and components of the application were tested thoroughly. The system passed all test cases, demonstrating that it is stable, functional, and ready for deployment. If additional features are introduced, testing will be updated to include them. This ensures continuous quality assurance for the project.

Remarks: No critical issues were identified during the testing process. All tests were completed successfully. However, since additional features are planned, this document will remain under review until all updates are finalized.

➢ **Prepared by: Areeba Zafar (Student Leader)**