Website Report: Testing, Optimization, and Security Measures

1. Test Cases Executed and Results

- **Product Fetching (API/Sanity Integration):**

  - **Test Description**: Ensured products are fetched correctly from the API or Sanity CMS.

  - **Expected Result**: Products should be displayed with accurate details from the source.

  - **Actual Result**: Test successful, with all product data fetching correctly.

- **Cart Functionality:**

  - **Test Description**: Verified the functionality of adding/removing items to/from the cart.

  - **Expected Result**: Cart should update correctly with added or removed products.

  - **Actual Result**: Test successful, cart functionality working as expected.

- **Checkout Functionality:**

  - **Test Description**: Checked if the checkout process runs smoothly.

  - **Expected Result**: Users should be able to proceed through checkout without errors.

  - **Actual Result**: Test successful, checkout functionality completed without issues.

- **Search Bar Functionality:**

  - **Test Description**: Ensured the search bar returns relevant product results based on user input.

  - **Expected Result**: Search results should accurately reflect the search query.

  - **Actual Result**: Test successful, search bar returns correct results.

- **Dynamic Routing:**

  - **Test Description**: Verified if dynamic routing works for product pages and category pages.

  - **Expected Result**: URLs should update based on user selections.

  - **Actual Result**: Test successful, dynamic routing functioning correctly.

- **Responsiveness:**

  - **Test Description**: Checked if the website adjusts properly across various screen sizes.

  - **Expected Result**: Website should be fully responsive and adapt to different devices.

  - **Actual Result**: Test successful, site is fully responsive.

- **Lighthouse Testing (SEO, Performance, Best Practices):**

  - **Test Description**: Performed Lighthouse tests to evaluate SEO, performance, and best practices.

  - **Expected Result**: High scores in SEO, performance, and best practices.

  - **Actual Result**: Test successful, with good scores across SEO, performance, and best practices.

- **User Acceptance Testing (UAT)**:
    - **Test Description**: Ensured that the website met the user requirements and expectations.
    - **Expected Result**: All functionalities should align with the requirements and work as intended.
    - **Actual Result**: UAT tests passed successfully.

2. Performance Optimization Steps

- **Image Compression**: Used TinyImage to compress images, reducing the load time significantly.
- **Lazy Loading**: Implemented lazy loading for images to enhance performance.
- **Next.js Image Tag**: Utilized Next.js Image tag for efficient image handling, ensuring better performance.
- **Performance Tools**: Used Postman for API testing, Lighthouse for general performance, and TinyImage for image compression.
- **Outcome**: These optimizations led to a noticeable decrease in load time, improving user experience.

3. Security Measures Implemented

- **Token-based Authentication**: Added ID tokens to secure user data and prevent unauthorized access.
- **Sensitive Information**: Ensured sensitive information such as API keys and tokens were stored securely in the .env.local file.
- **Outcome**: Security measures were successfully implemented and sensitive data is protected.

4. Challenges Faced and Resolutions

- **Challenges**: No significant challenges were faced during the development process.
- **Resolutions**: The project proceeded smoothly without any major roadblocks.

5. Features Not Implemented

- **User Authentication**: The user authentication functionality was not implemented in the website.
- **Order Tracking**: The order tracking functionality was also not added to the website.