Deployment Documentation: Furniture Marketplace Website

1. Project Overview

This project is a **Furniture Marketplace Website** built with **Next.js 15**, **Sanity CMS**, and **Tailwind CSS**. The website allows users to browse various furniture products, add them to a shopping cart, and proceed with checkout functionalities.

Key Features:

- **Product API Routes:** Dynamic product pages fetched from Sanity CMS.
- **Cart Functionality:** Users can add/remove items from the cart.
- **Search Functionality:** Search products using keywords.
- **Dynamic Routing:** Routes for product details are dynamic and SEO optimized.
- **Responsiveness:** Fully responsive design using Tailwind CSS for all devices.

2. Deployment Process

2.1 Deployment Platform

The project is deployed using **Vercel**, a cloud platform optimized for Next.js applications.

Steps:

1. Push the latest code to the GitHub repository
2. Link the repository to Vercel.
3. Vercel automatically detects the Next.js framework and deploys the project.
4. After deployment, Vercel provides a URL (e.g., https://furniture-marketplace.vercel.app) for the live website.

2.2 Environment Variables

Added the environment variables to .env.local for sensitive information.

2.3 File Structure

Ensure the following key directories and files are included:

- **app/**: Contains the Next.js app directory with routes, components, and pages.
- **sanity/**: Sanity CMS setup and configuration files.
- **.env.local**: Environment variables for API access and sensitive data.

2.4 Sanity Configuration

- Configure Sanity CMS to define product schemas with defineType.
- Data such as product names, images, descriptions, and prices are stored in Sanity.

---

3. Performance Report

The website's performance was tested using **Google Lighthouse** to evaluate key metrics like SEO, performance, and best practices. Below is the summary of the results:

Metric	Score	Comments
Performance	97.5/100	Excellent load time, uses Next.js Image tag for lazy loading.
Accessibility	90/100	No accessibility issues found.
Best Practices	100/100	Followed best practices for web development.
SEO	100/100	SEO-friendly URLs and dynamic metadata.

Key Performance Optimizations:

- **Lazy Loading:** Products and images are loaded lazily using Next.js's built-in functionality.
- **Compressed Images:** Used **TinyImage** for image compression to reduce page load times.
- **Next.js Image Tag:** Optimized images with the next/image component for automatic resizing and performance improvements.

---

4. Test Case Reports

4.1 Functional Tests

1. Product Fetching:

- **Test Case:** Verify products are fetched from Sanity CMS and displayed correctly on the website.
- **Status:** Passed
- **Details:** Successfully fetched product data through API routes and displayed on dynamic product pages.

2. Cart Functionality:

- **Test Case:** Verify that users can add and remove items from the shopping cart.
- **Status:** Passed
- **Details:** Cart functionality works as expected. Items can be added and removed using the "Add to Cart" and "Remove from Cart" buttons.

3. Checkout Process:

- **Test Case:** Verify that the checkout process is functional (without payment gateway).
- **Status:** Passed
- **Details:** Successfully navigated through the checkout page, displaying cart items, and user information. The payment gateway was not integrated for this version.

4. Search Functionality:

- **Test Case:** Verify that the search bar returns relevant products based on the search query.
- **Status:** Passed
- **Details:** Search works correctly, returning products based on keywords from the product names or descriptions.

5. Responsive Design:

- **Test Case:** Verify that the website layout adjusts correctly across different screen sizes (mobile, tablet, desktop).
- **Status:** Passed
- **Details:** The layout adjusts properly for all device sizes using Tailwind CSS classes for responsiveness.

4.2 Security Tests

1. Sensitive Data Protection:

- **Test Case:** Verify that sensitive data such as API tokens and user information are not exposed.
- **Status:** Passed
- **Details:** Sensitive data is secured using the .env.local file, which is not pushed to version control.

---

5. Known Issues & Limitations

- **User Authentication & Order Tracking:** These features were not implemented for this version of the website but are planned for future development.
- **Payment Gateway Integration:** Payment processing was not included, though it can be integrated in the future.

---

6. Conclusion

This furniture marketplace website is fully deployed and optimized for performance, SEO, and usability. It successfully supports product browsing, cart functionality, and checkout features. The website has been tested thoroughly for functionality and performance, and it meets the requirements for a smooth and secure user experience.

For any further development or bug fixes, the source code can be accessed via GitHub.