DAY 6 - DEPLOYMENT PREPARATION AND STAGING ENVIRONMENT SETUP

1. Overview

The Furniture E-commerce website is developed using Next.js 14, with data fetched from Sanity CMS and styled using Tailwind CSS. The project is deployed on Vercel for seamless hosting and continuous deployment.

2. Deployment Preparation

Before deploying the website to production, several steps must be taken to ensure smooth performance, security, and reliability. Below are the key preparations:

Code Optimization:

- Minimize unnecessary dependencies and optimize the Next.js build.
- Use `next/image` for optimized image rendering.
- Enable caching strategies to improve performance.
- Remove unused code and console logs.

Environment Variables:

- Store sensitive credentials like Sanity project ID, dataset, and tokens in `.env.local`.
- Ensure that these variables are properly set in Vercel's environment settings.
- Use `NEXT_PUBLIC_` prefix for variables that need to be accessible on the client side.

Database and CMS Configuration:

- Ensure Sanity CMS is configured correctly with the right dataset (e.g., production dataset for live deployment).
- Test all content fetching and mutations to avoid API failures.
- Set up webhooks in Sanity to trigger revalidation when content is updated.

Testing and Debugging:

- Conduct unit testing for React components.
- Perform integration testing for CMS data fetching.
- Ensure no errors or warnings appear in the console.
- Test the website across multiple devices and screen sizes.

3. Staging Environment

Before launching into production, a staging environment helps test the website in a near-production setting.

Staging Deployment on Vercel:

- Create a separate Vercel project or deploy a branch as a preview.
- Use Vercel's preview deployments to test changes before merging into the main branch.
- Set up environment variables for the staging environment.

Data Synchronization:

- Use a separate Sanity dataset for staging to avoid modifying live data.
- Ensure that any content updates in staging do not affect production.

Performance and Load Testing:

- Analyze site performance using Lighthouse.
- Check Core Web Vitals for loading speed, interactivity, and stability.

User Acceptance Testing (UAT):

- Conduct end-to-end testing by simulating real user interactions.
- Validate checkout process, product filtering, and cart functionalities.
- Ensure responsiveness and accessibility compliance.

4. Production Deployment on Vercel

Once the staging environment is tested successfully, the website is ready for production deployment:

Final Checks:

- Merge the staging branch into the main branch.
- Double-check environment variables and secrets.
- Ensure database and CMS configurations point to production data.

Deploying to Production:

- Push the latest code to the `main` branch.
- Vercel will automatically trigger a production deployment.
- Monitor the deployment logs for errors.

Post-Deployment Monitoring:

- Set up real-time monitoring with Vercel Analytics.
- Regularly check for SEO and performance improvements.

Deployment Link:

https://hackathone-e-commerce-website.vercel.app/

5. Conclusion

Following these deployment preparations and staging environment steps ensures that the Furniture E-commerce website is stable, secure, and high-performing in production. With Vercel's seamless deployment workflow and Sanity's powerful CMS integration, managing and scaling the website becomes efficient and hassle-free.