

Day-6, Report – 2

Deployment & Environmental Variables Strategy for Next.js E-Commerce

Setting Up the Deployment Process

Version Control with GitHub:

- Ensure your project is fully version-controlled using GitHub.
- Create a repository for the project and push your codebase to GitHub.
- Use branches (e.g., main development, or feature branches) to manage changes effectively.



Integrating with Vercel:

- Connect your GitHub repository to Vercel.
- Enable automatic deployment, so any changes pushed to the `main` branch trigger a new deployment.

Vercel Features and Benefits

Automatic Builds:

- Vercel automatically detects your Next.js configuration and builds the website without manual setup.
- Built-in optimizations for Next.js, such as server-side rendering (SSR) and static site generation (SSG).

Custom Domains:

- Easily link custom domains to your eCommerce site.

- Vercel provides free SSL certificates for secure browsing.

Serverless Functions:

- Deploy API routes and serverless functions directly using Vercel.
- Great for handling eCommerce-specific functionalities like payment processing or user authentication.

Performance Insights:

- Vercel provides analytics on website performance, including load time and Core Web Vitals.

Global CDN:

- Content is automatically distributed via a global Content Delivery Network (CDN) for fast load times worldwide.

Environmental Variable Configuration

- Environment variables are used to securely store sensitive information such as API keys, database URLs, and other configurations that should not be exposed in your codebase. Vercel provides a secure way to manage and use these variables in your Next.js application.

Environment Variables		
Key	Value	
bilqeesshahid02@gmail.com	bilqeesshahid02@gmail.com	—
NEXT_PUBLIC_SANITY_PROJECT_ID	fzicsaok	—
NEXT_PUBLIC_SANITY_DATASET	production	—
SANITY_API_TOKEN	skierthlr7XdgfpKwH2czezvvXoJvlgGe3l	—
SHIPENGINE_API_KEY	TEST_ojNctACmTA2K7nYD5PiGwKUd5K	—
SHIPENGINE_FIRST_COURIER	se-1706423	—
SHIPENGINE_SECOND_COURIER	se-1706424	—
SHIPENGINE_THIRD_COURIER	se-1706425	—
SHIPENGINE_FOURTH_COURIER	se-1706454	—
NEXT_PUBLIC_CLERK_PUBLISHABLE	pk_test_aW5maW5pdGUtc3VuZmlzaCC	—
CLERK_SECRET_KEY	sk_test_kxaK9KO5dx00kOgN1ie5eGlxU	—

Setting Up a Staging Environment on Vercel

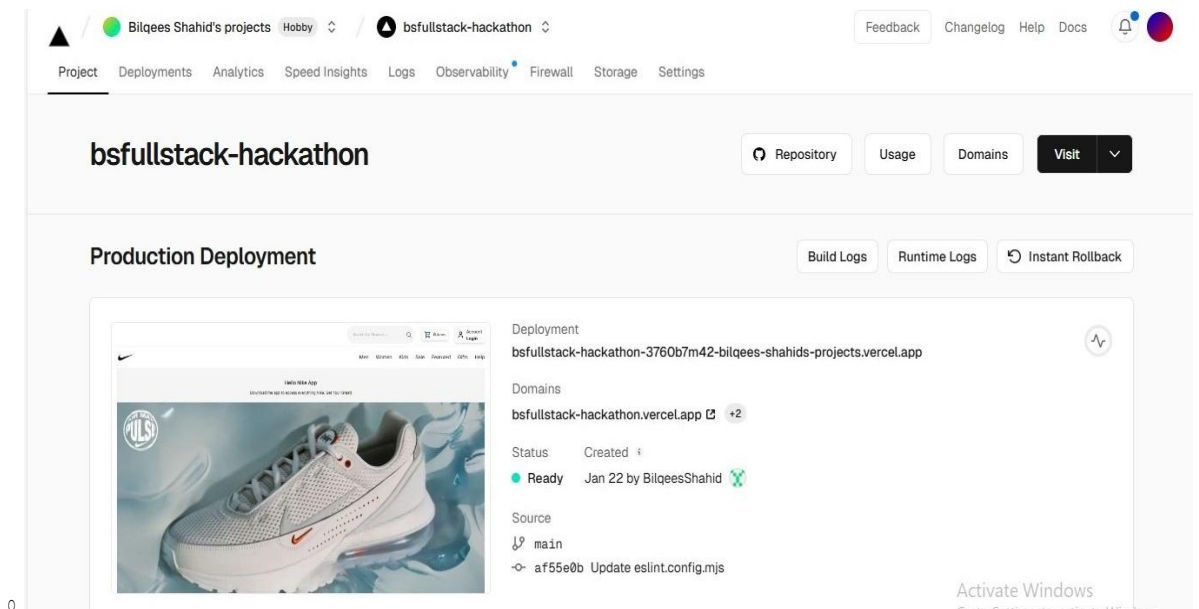
A staging environment is a replica of your production environment used for testing new features, bug fixes, and updates before deploying them live. Here's how to set up a staging environment on Vercel

Benefits of a Staging Environment

- **Safe Testing:** Test new features, updates, or bug fixes without affecting the live site.
- **Replicates Production:** Simulates real-world behavior by mimicking the production environment.
- **Identifies Issues Early:** Detects bugs, UI issues, or performance problems before they reach end-users.

Testing on Staging:

- Access the staging environment via the Vercel-generated URL.
- Test all new features, integrations, and bug fixes thoroughly.



Check List for Day 6

Deployment Preparation: Done

Staging Environment: Done

Documentation: Done

Form Submission: Done Final

Review: Done

Conclusion:

On Day 6 of the hackathon, the focus was on deployment and ensuring the application was production-ready. The deployment process involved setting up environment variables to securely manage sensitive information such as API keys and database URLs. By leveraging Vercel's capabilities, the application was deployed efficiently, allowing us to benefit from features like automatic builds, global CDN, and preview deployments.

Post-deployment, we thoroughly checked the website's efficiency, including performance, responsiveness, and functionality, to ensure a smooth user experience. The task emphasized the importance of a well-configured deployment process and proper environment management, which are crucial for delivering a robust and secure web application.