

DAY-06

ARJUMAND AFREEN TABINDA



Logo
Name

Objective:

To effectively deploy the project on netlify, configure environment variables, establish a staging environment, and perform multiple tests to verify the functionality, performance, and security of the application.

Here's a rephrased version of your deployment strategy planning:

****Deployment Strategy Planning****

- **Selected Hosting Platform: netlify**
- Following an assessment of different hosting platforms like Netlify, AWS, and Azure, netlify was chosen for its user-friendly interface, seamless integration with Next.js, and automatic optimization for production deployments.
- **Integration of Backend Services:**
The application interacts with Sanity CMS for content management, and relevant API calls are configured to fetch data dynamically from the backend.
- Third-party APIs are integrated for features such as payments, notifications, or other external functionalities.
- Ensured smooth integration and correct handling of API responses during deployment.

Environment Variable Configuration

- **Secure API Keys and Sensitive Data:**
- .env files were used locally to store sensitive credentials such as API keys, database credentials, and other private data.
- During deployment on Netlify, these environment variables were configured within Netlify's environment settings to ensure secure access to sensitive data without exposing it in the codebase.

- **Netlify Configuration:**

- **Set environment variables in Netlify's dashboard for proper integration with backend services and third-party APIs. This ensures secure access to API keys and sensitive data only in the deployed environment.**

3. Staging Environment Setup

- **Deployment to Staging:**

The project was deployed to Netlify's staging environment for testing. This production-like environment replicates the live setting, allowing us to verify the application's behavior before moving it to production.

- **Build Verification:**

- **The deployment build process was monitored to ensure successful completion.**

- **Verified that the site loaded without errors, confirming that all resources, images, and styles were properly rendered.**

3. Staging Environment Setup

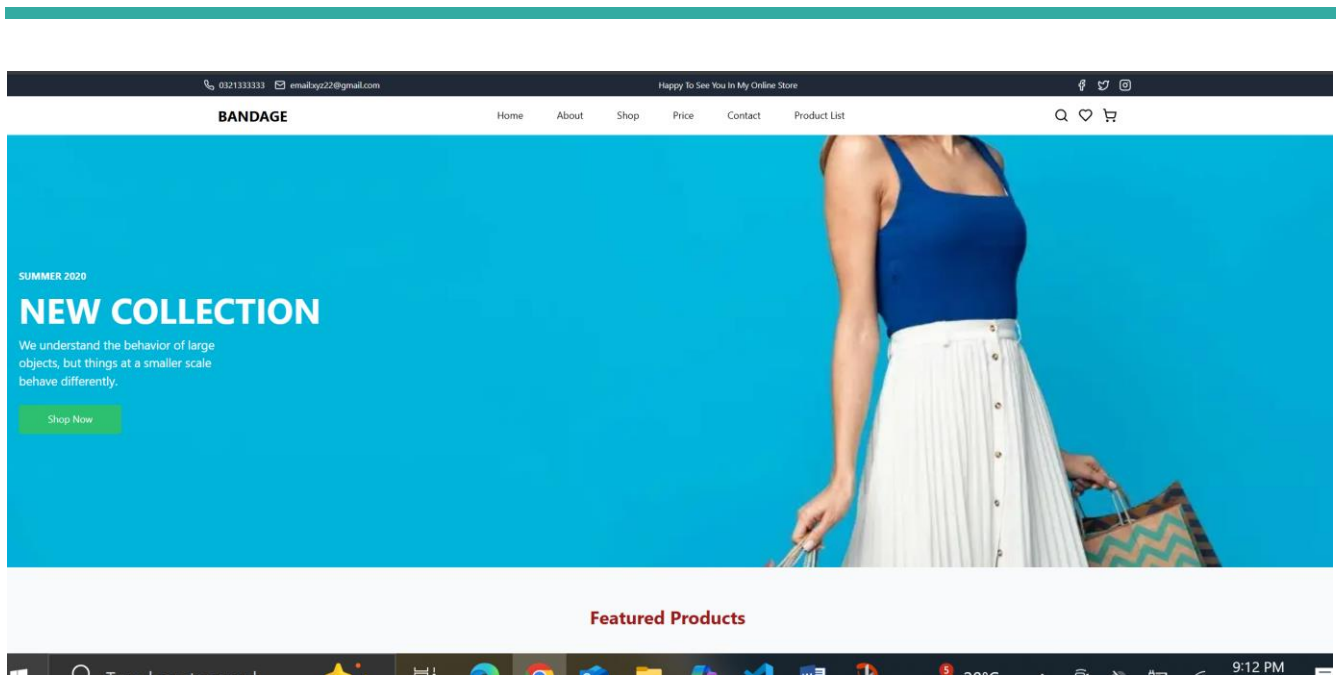
- **Deployment to Staging:**

The project was deployed to Netlify's staging environment for testing. This production-like environment replicates the live setting, allowing us to verify the application's behavior before moving it to production.

- **Build Verification:**

- **The deployment build process was monitored to ensure successful completion.**

- **Verified that the site loaded without errors, confirming that all resources, images, and styles were properly rendered.**



3. Staging Environment Setup

- **Deployment to Staging:**

The project was deployed to Netlify's staging environment for testing. This production-like environment replicates the live setting, allowing us to verify the application's behavior before moving it to production.

- **Build Verification:**

- The deployment build process was monitored to ensure successful completion.

- Verified that the site loaded without errors, confirming that all resources, images, and styles were properly rendered.

5. Documentation and Unresolved Issues

- **Test Results:**

- All tests were completed successfully for functional, performance, and security assessments. The performance score exceeded 85% on Lighthouse, and no significant security vulnerabilities were detected.

-
- Confirmed that the application was responsive, and error handling functioned as intended.
 - Unresolved Issues:
 - No major unresolved issues were identified during the deployment and testing process. Minor UI adjustments were made based on responsiveness tests for mobile devices, but these did not affect functionality or performance.

Conclusion:

The project was successfully deployed on Netlify and validated through a series of tests in the staging environment. All features were tested and confirmed to work as expected. The website passed functional, performance, and security checks and is now prepared for production deployment.