

Report on Day 6: Deployment Preparation and Staging Environment Setup

Objective:

Day 6 focuses on preparing the marketplace application for deployment by setting up a staging environment, configuring hosting platforms, and ensuring the application is ready for a customer-facing release. This stage builds on the testing and optimization work from Day 5, ensuring that the marketplace operates seamlessly in an environment that closely mirrors production.

Deployment Strategy Planning:

1. Choose a Hosting Platform:

Vercel was selected as the hosting platform due to its seamless integration with Next.js and its ability to automatically deploy projects from GitHub. Key benefits include:

- o **Easy Integration:** Native support for Next.js.
- o **Automatic Deployment:** Deploys automatically from GitHub.
- o **Scalability:** Handles varying traffic loads efficiently.
- o **Serverless Functions:** Enables backend logic without managing servers.

2. Finalize Application's Interaction with Backend Services:

Before deploying, it is essential to ensure that the application interacts correctly with backend services, such as Sanity CMS.

Sanity CMS: Confirm proper fetching of content with correct configuration of the Sanity client and API keys.

Environment Variables:

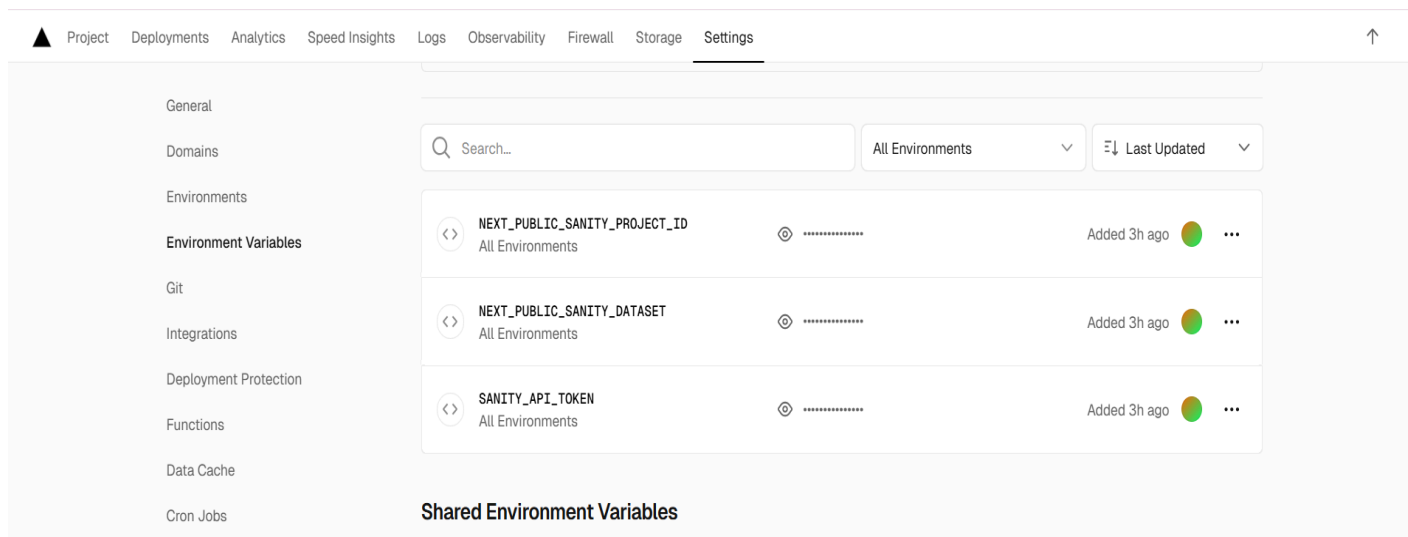
Sensitive Data Using .env Files

Sensitive data like **Sanity Project ID** and **Sanity API Token** should be stored securely in .env files. This prevents exposing secrets in the codebase.

```
NEXT_PUBLIC_SANITY_PROJECT_ID,  
  
SANITY_API_TOKEN;  
  
NEXT_PUBLIC_SANITY_DATASET
```

Configure Environment Variables on Vercel:

After setting up .env files in the development environment, configure these variables in Vercel to ensure secure deployment. This step is crucial for maintaining data security in production.



Staging Environment Setup:

1. Deploy the Application to Staging

- o Push the latest code to GitHub.
- o Set up a Preview Deployment in Vercel (either by linking a staging branch or using the default production branch).
- o Vercel will trigger the build and deployment process automatically.

2. Validate Deployment

- o Check Vercel's build logs for a successful deployment.
- o Access the staging URL and test the site to ensure all features work as expected.
- o Verify the functionality of backend integrations (e.g., Sanity CMS and APIs).

3. Troubleshoot

- o If issues arise, review build logs and verify the correct configuration of environment variables.

Staging Environment Testing:

Performance Testing

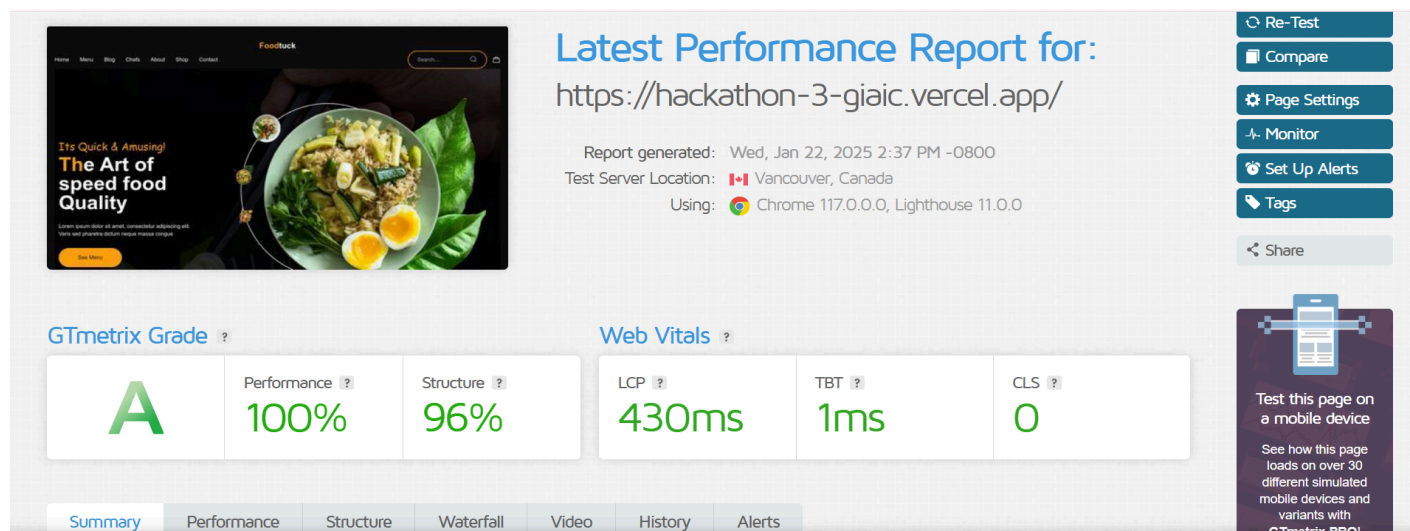
- o **GTmetrix:** Analyze site performance, including load times and speed.
- Optimize as necessary

Verify Responsiveness & Error Handling

- o Test across various devices and screen sizes to ensure the site is mobile friendly.
- o Ensure proper error handling for broken links and failed interactions.

Document Test Results

- o Record all test outcomes, including performance benchmarks, functional tests, and any unresolved issues.



Conclusion:

Day 6's focus on deployment preparation and staging environment setup has ensured that the marketplace application is ready for deployment. The application has been deployed to a staging environment on Vercel, environment variables have been securely configured, and all necessary testing (functional, performance, and security) has been completed. performance benchmarks have been documented, and the project repository is well-organized with a professional README.md file. With these steps completed, the application is prepared for a smooth transition to production