Prepared By: Shahzeena Samad
Date: January 22, 2025

Day 6 - Staging Environment Setup & Deployment

Objective:

The goal for Day 6 is to ensure the marketplace is ready for launch by configuring the staging environment, selecting the hosting platform, and ensuring smooth functionality through rigorous testing. This process builds upon previous optimization and testing to make sure the app is fully prepared for production use.

Step 1: Hosting Platform Configuration:

1. ### Choosing Vercel for Hosting:
   - Vercel was selected due to its strong integration with GitHub, scalability, and its optimization for Next.js applications—perfect for an e-commerce website that requires efficient performance.

2. ### Linking GitHub Repository to Vercel:
   - The GitHub repository was connected to Vercel to automate the deployment process, ensuring that every push to the main branch results in an immediate deployment to the staging environment.

Step 2: Environment Variables Setup:

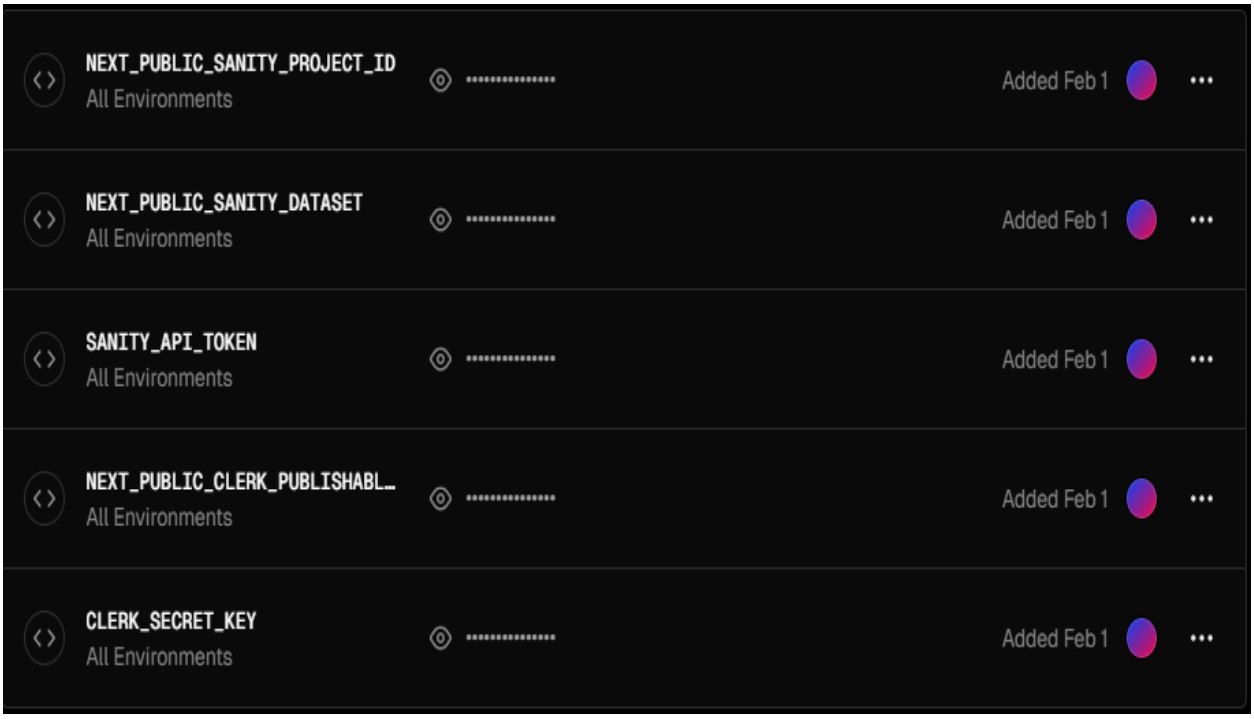
1. ### Creating and Managing .env File:
   - A `.env` file was created to manage important environment-specific variables, such as:

```
NEXT_PUBLIC_SANITY_PROJECT_ID=your_project_id
NEXT_PUBLIC_SANITY_DATASET=production
```

```
API_KEY=your_api_key
```

2. Uploading Variables to Vercel Dashboard:

- Secure environment variables were added to Vercel's dashboard for production, ensuring sensitive information is not exposed in the codebase.



Step 3: Staging Deployment:

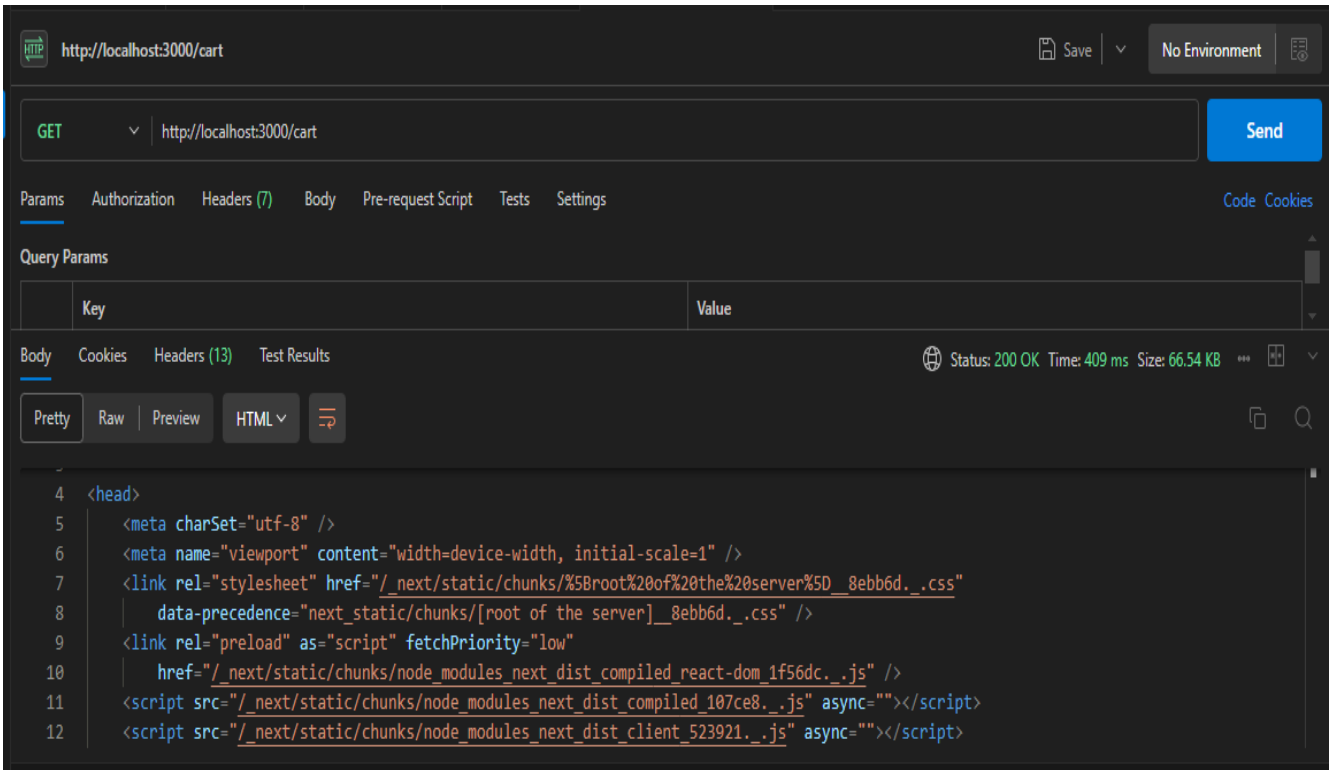
1. Deploying to Staging:

- The application was deployed to the staging environment via Vercel. This was done by pushing the latest changes to the GitHub repository, which triggered the deployment process.

2. **Verification and Basic Functionality Check:**
    - After deployment, the marketplace was verified to ensure essential functions—such as product listing, cart updates, and checkout—were working correctly.

Step 4: Testing in Staging Environment:

1. **Test Types Conducted:**
    - **Functional Testing:** Ensured that critical features like product listing, cart management, and the checkout process functioned as expected.
    - **Performance Testing:** Used Lighthouse to analyze key performance metrics, including loading speed and site responsiveness.
    - **Security Testing:** Confirmed that the website was using HTTPS for secure communications and that API interactions were properly secured.



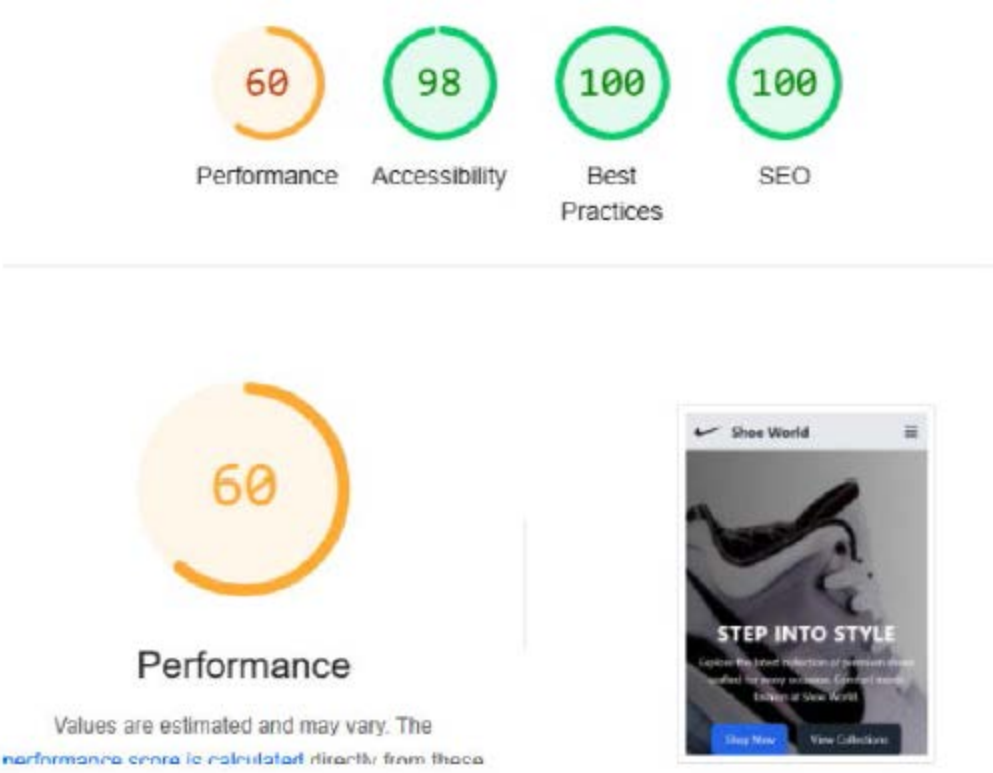
2. Test Case Documentation:

- Test cases were carefully documented in a CSV file, detailing the results of each test to ensure complete functionality across the website.

Test Case ID	Objective	Expected Result	Actual Result	Status
TC-001	Validate product detail display.	Accurate product details displayed.	Data matched API response.	Passed
TC-002	Test real-time search functionality.	Search results update dynamically.	Accurate results; speed improved.	Passed
TC-003	Verify cart operations.	Cart operations work seamlessly.	All operations performed as expected.	Passed
TC-004	Handle invalid API requests.	Error message displayed gracefully.	Error message shown; system stable.	Passed
TC-005	Validate product listing accuracy.	Products listed accurately.	Products displayed as expected.	Passed
TC-006	Test category filtering.	Category filter works correctly.	Filter worked as intended.	Passed

3. Performance Testing Insights:

- After analyzing the performance with Lighthouse, specific areas requiring optimization were identified, particularly around loading speeds.



Step 5: Documentation and File Organization:

1. **Updating the README.md:**
   - The README.md file was updated to include detailed deployment instructions, environment variable setup, and test results for easy reference.

2. **File Structure Organization:**
   - The project files were neatly organized into a structured folder hierarchy to ensure efficient management and smoother future updates.

Conclusion:

The marketplace was successfully deployed to Vercel, with all core features thoroughly tested in the staging environment. Both performance and security were validated, with a few areas for improvement noted. Documentation was meticulously updated for future reference, and the project is now ready to move into the production phase.