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 platfons, 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: Step 1

1. **Choose a Hosting Platform**: Vercel (Recommended) Vercel was selected as the hosting platform due to its seamless integration with Next.js and its ability to automatically deploy projects from GitHuh Key benefits include:

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

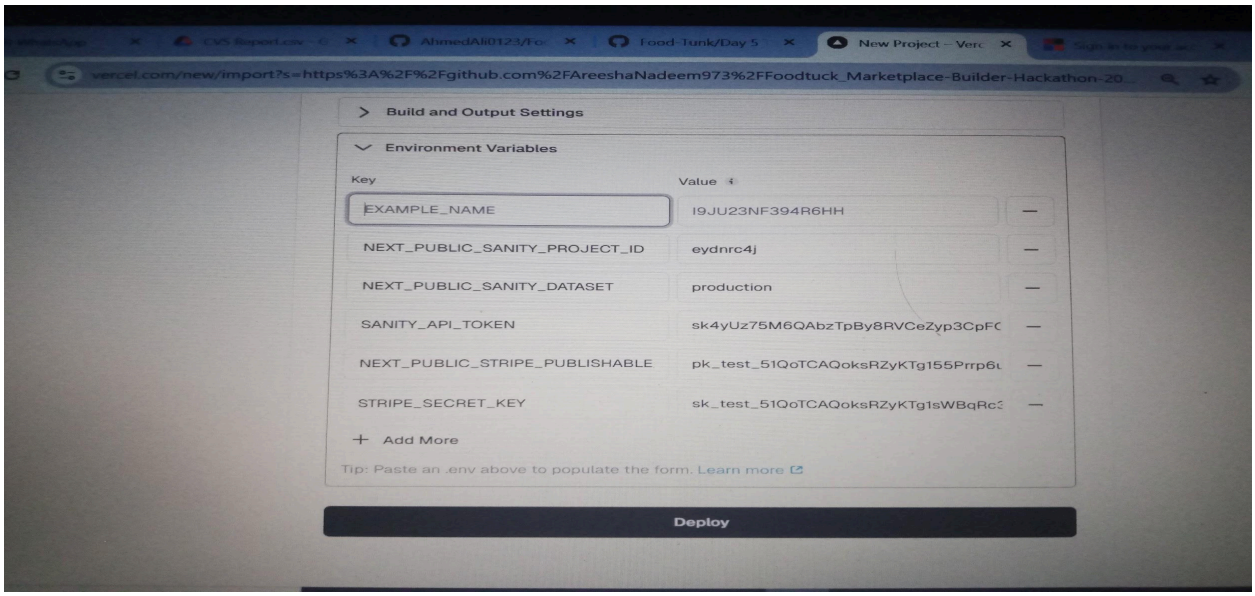
2. **Finalize Application's Interaction with Backend Services** Before deploying, it is essential to ensure that the application interacts correctly with backend services, such as Sanity CMS and third-party APIs.

- **Sanity CMS:** Confirm proper fetching of content with correct configuration of the Sanity client and API keys.
- **Third-party APIs:** Ensure API calls work securely and that sensitive dara like API keys are stored in environment variables.

Environment Variable Configuration: Step 2

1. **Secure API Keys, Database Credentials, and Sensitive Data Using.env Files** Sensitive data like API keys and credentials should be stored securely in. eny files. This prevents exposing secrets in the codebase.

2. **Configure Environment Variables in the Hosting Platform for Secure Deployment** After setting up. cav 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: Step 3

1. Deploy the Application to Staging

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

2. Validate Deployment

- Check Vercel's build logs for a successful deployment.
- Access the staging URL and test the site to ensure all features work as expected.

- Verify the functionality of backend integrations (e.g., Sanity CMS and APIs).

3. Troubleshoot

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

Staging Environment Testing: Step 4

1. **Conduct Functional Testing**

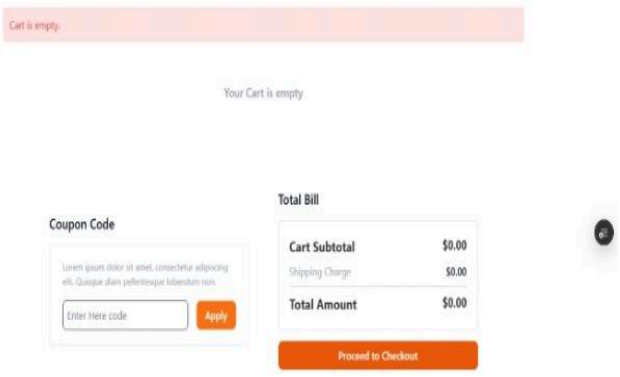
- **Cypress**: Test workflows like product listing and checkout correctly.
- **Postman**: Validate API responses to ensure the application fetches data

2. **Perform Performance Testing**

- Lighthouse or GTmetrix Analyze site performance, including load times and speed. Optimize as necessary.

3. **Verify Responsiveness & Error Handling**

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



4. **Document Test Results**

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

Test Case Reporting

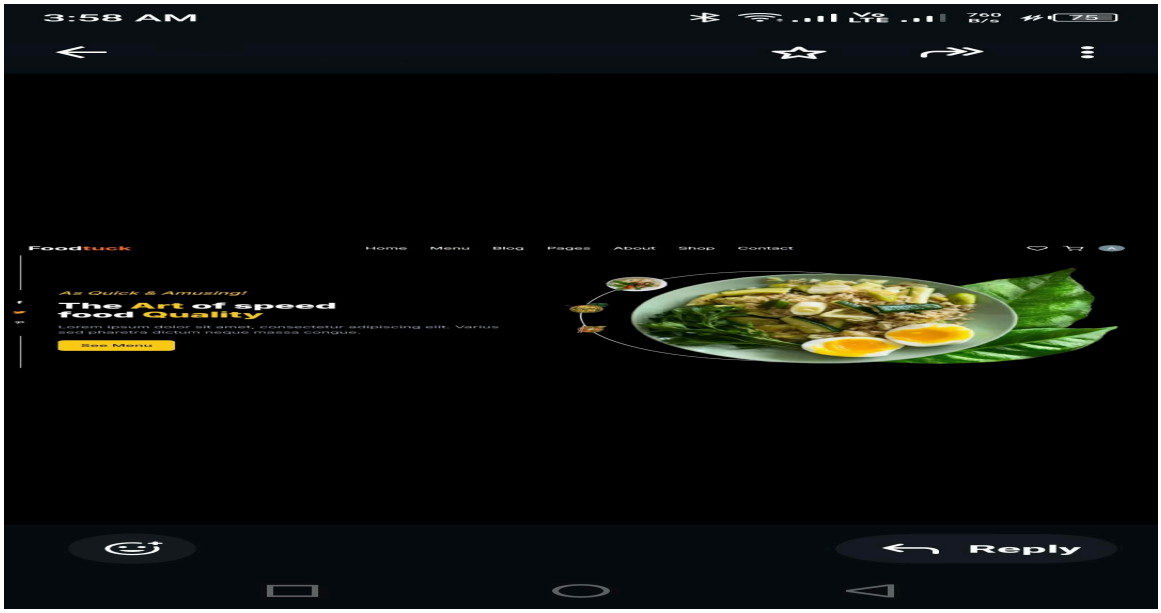
- Record all test outcomes, including performance benchmarks and functional tests.
- List unresolved issues and prioritize fixes before production deployment.

Test Case Reporting

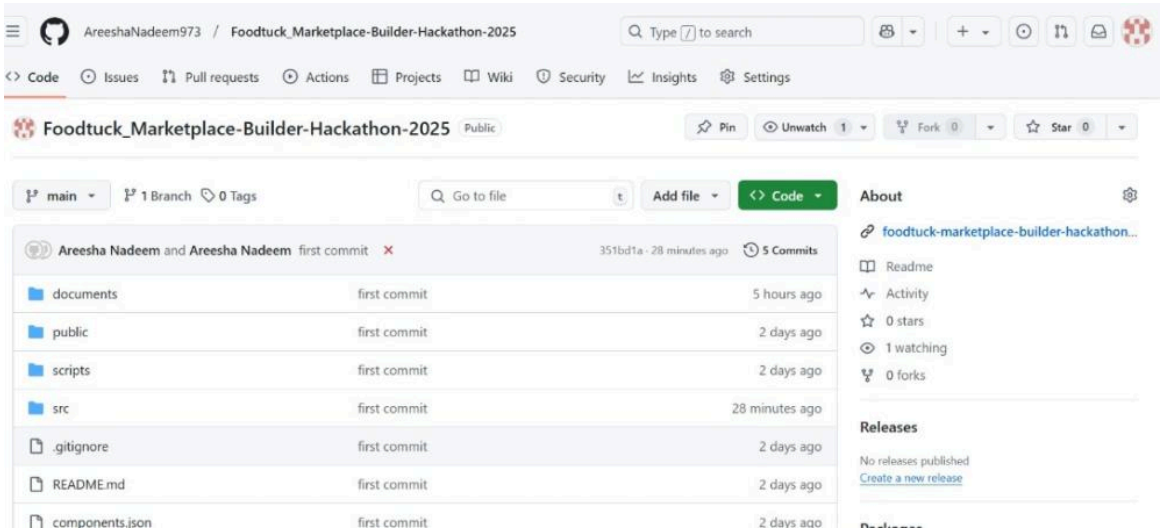
Documenting test cases in a structured format ensures clear tracking of testing progress and issues. Here's how you can document test cases in a CSV file:

Test Case ID	Description	Steps	Expected Result	Actual Result	Status	Remarks
TC001	Validate product listing	Open product page > Verify products	Products displayed	Products displayed	Passed	No issues found
TC002	Test API error handling	Disconnect API > Refresh page	Show fallback message	Fallback message shown	Passed	Handled gracefully
TC003	Check cart functionality	Add item to cart > Verify cart	Cart updates correctly	Cart updates correctly	Passed	Works as expected
TC004	Test form validation	Submit form with empty fields	Display error message	Error message displayed	Failed	Missing validation check
TC005	Verify HTTPS connection	Open site > Check HTTPS status	HTTPS enabled	HTTPS enabled	Passed	Secure connection

Vercel



Github



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. Test cases and 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.