

Day 6: Deployment Preparation and Staging Environment Setup

1. Staging Environment Setup

- **Hosting Platform:** [Insert chosen platform name, e.g., Vercel]
- **Repository Connection:** Connected GitHub repository to [Vercel/Netlify]. The project was linked via the platform's dashboard.
- **Build and Deployment Settings:**
 - Build settings configured as per platform requirements.
 - Ensured that the correct build commands (e.g., `npm run build`) are specified.
 - Deployment pipeline verified to trigger automatically after code push to main branch.

2. Environment Variables

- **Secure Variables:**

`.env` file created with sensitive information:

Ex : `NEXT_PUBLIC_SANITY_PROJECT_ID=your_project_id`
`NEXT_PUBLIC_SANITY_DATASET=production`
`API_KEY=your_api_key`

- **Uploading Variables to Platform:**
 - Environment variables added securely via the platform's dashboard (e.g., Vercel's Environment Variables section).

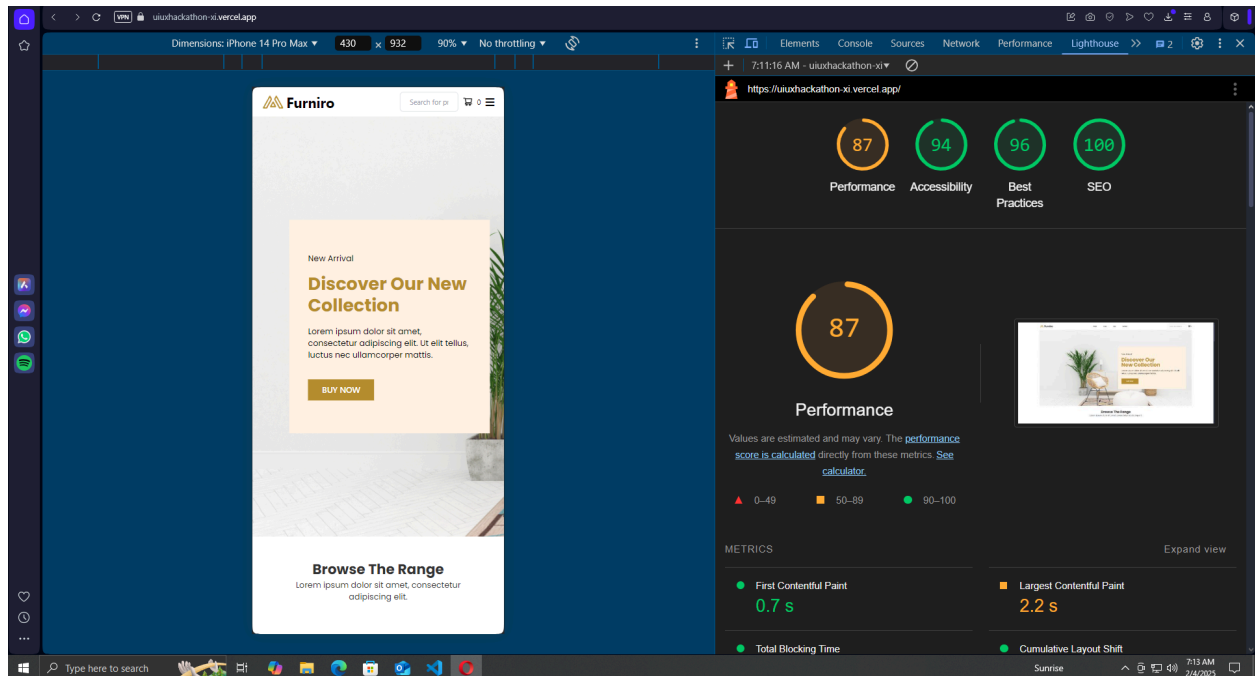
3. Staging Deployment

- **Deployment Confirmation:**
 - Deployment completed successfully without errors.
 - Site loaded correctly in the staging environment.
 - Any issues faced during deployment were resolved (e.g., misconfigured environment variables).

4. Staging Environment Testing

- **Types of Testing Conducted:**

1. **Functional Testing:** Verified features such as:
 - Product listing.
 - Search functionality.
 - Cart operations.
2. **Performance Testing:**
 - Used **Lighthouse** to check site performance.
 - Results showed total performance score of 87% which will be more improved soon.



● Test Case Reporting :

Documented all test cases in the CSV format.

day_6_Testing_Report_Marketplace-General-Ecommerce									
File Edit View Insert Format Data Tools Extensions Help									
Menus									
100% 123 Default...									
A	B	C	D	E	F	G	H	I	
1	Test Case ID	Test Case Description	Test Steps	Expected Result	Actual Result	Status	Severity Level	Assigned To	Remarks
2	TC001	Validate product listing page	Open product page > Verify products	Products displayed correctly	Products displayed correctly	Passed	Low	-	No issues found
3	TC002	Test API error handling	Disconnect API or make invalid query > Refresh page	Fallback UI with error message if query doesn't pull data	Fallback UI displayed as expected	Passed	Medium	-	Handled gracefully
4	TC003	Verify cart functionality	Add product to cart > Verify cart contents	Cart updates with added product	Cart updates as expected	Passed	High	-	Works as expected
5	TC004	Check mobile responsiveness	Resize browser window > Verify layout adjustment	Layout adapts to screen size	Layout adjusts properly	Passed	Medium	-	Test successful
6	TC005	Check product page load	Open product page > Observe page load	Page loads within 3 seconds	Page loads as expected	Passed	High	-	Works as expected
7	TC006	Validate security in production	Run security scan > Check for vulnerabilities	No security issues found	No vulnerabilities detected	Passed	High	-	No issues found
8	TC007	Measure page load time	Reload product page multiple times	Page load time under 2 seconds	Page loads under 2 seconds	Passed	Medium	-	Performance meets expectations
9	TC008	Check application stability	Test functionality post-deployment	Application runs smoothly without errors	Application stable	Passed	Medium	-	No issues found
10									
11									

Conclusion

- The staging environment has been successfully set up and tested.
- Deployment to the staging environment is complete, and all functionalities have been verified.
- Test case reports and performance results have been documented.