Day 6 - Deployment Preparation and Staging Environment Setup

Objective:

Day 6 focuses on getting the marketplace ready for launch by setting up a staging environment, configuring hosting platforms, and ensuring everything works as expected. It builds on previous testing and optimization, ensuring the app is production-ready and follows industry standards for managing different environments like development, staging, and production.

Step 1: Hosting Platform Setup:

1. Hosting Platform: Vercel

- chose **Vercel** for its easy integration with GitHub, automatic scaling, and optimization for Next.js, ideal for e-commerce sites.

2. Connecting GitHub Repository to Vercel:

- Linked the GitHub repository to Vercel for **automatic deployment** every time code is pushed to the main branch.

Step 2: Configure Environment Variables:

1. .env File Creation:

- Created a .env file to manage sensitive keys like:

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

2. Uploading Variables to Vercel:

- Added environment variables to Vercel's dashboard for secure usage in production.

Step 3: Deploy to Staging:

1. Deploying Application:

- Deployed the app to **staging** via Vercel by pushing changes to the GitHub repository. Vercel automatically built and deployed the website.

2. Validation:

- Verified successful deployment and checked basic functionalities like product listings, cart updates, and checkout.

Step 4: Staging Environment Testing:

1. Testing Types:

- **Functional Testing**: Validated features like product listing and cart functionality.
- **Performance Testing**: Used **Lighthouse** to analyze site speed and responsiveness.
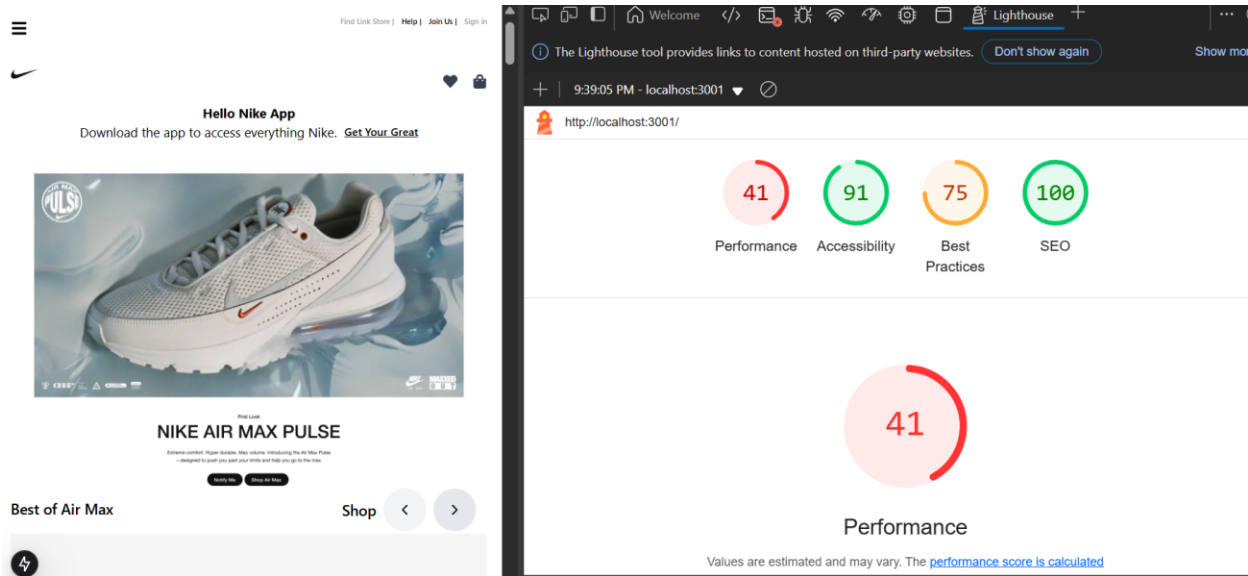- **Security Testing**: Ensured HTTPS and secure API communications.

2. Test Case Reporting:

- Documented test cases in a CSV file, ensuring proper functionality in all critical areas.

Test Case ID	Test Case Description	Test Steps	Expected Result	Actual Result	Status	Severity Level	Remarks
TC-001	Product Detail Page	Open product page. Click on product	Product detail page shows accurate information	Product detail page shows accurate information	Passed	Low	No issues found
TC-002	Dynamic Search Bar	Go to search bar. Enter "Nike Air"	Relevant results for "Nike Air" are displayed	Accurate search results for "Nike Air"	Passed	Medium	Speed can be improved
TC-003	Cart Functionality	Add item to cart. View cart. Remove item	Cart shows correct items, quantities, and prices	Cart worked as expected	Passed	High	Smooth operations
TC-004	API Error Handling	Trigger invalid product ID API. Verify error message	"Product not found" error without crash	"Product not found" message displayed	Passed	High	Robust error handling
TC-005	Product Listing Page	Open product listing. Scroll through page	Products listed with name, price, and image	Products displayed correctly	Passed	Low	No issues found

3. Performance Testing Report:

- Analyzed speed using Lighthouse, and the results showed areas for further optimization.

Step 5: Documentation Updates:

1. README.md:

- Created a README.md file summarizing deployment steps, environment variables, and test results.

2. File Organization:

- Organized all project files in a structured folder hierarchy for easy management.

Conclusion:

The e-commerce website was successfully deployed to **Vercel**, and testing confirmed that core functionalities work as expected. Performance and security were tested, with minor optimizations recommended. Documentation was updated for future reference.