Day 6 - Deployment Preparation and Staging Environment Setup

Prepared by: ZARTASH IMRAN

STUDENT GIAIC WEDNESDAY SLOT 7 TO 10

---

1. Objective:

To prepare the marketplace application for deployment by setting up a staging environment, testing it in a production-like setting, and documenting the deployment process.

---

2. Key Learning Outcomes:

1. Setting up and configuring a staging environment.
2. Understanding environment management stages like TRN, DEV, SIT, UAT, PROD, and DR.
3. Conducting functional, performance, and security testing.
4. Organizing project files in a structured GitHub repository with proper documentation.

---

3. Steps Taken:

Step 1: Hosting Platform Setup

- Selected Hosting Platform: Vercel
- Connected GitHub Repository:
  - Linked the project repository to Vercel.
  - Configured build settings to ensure successful deployment builds.

Step 2: Environment Variable Configuration

- Created a `.env` file with the following variables:

```
NEXT_PUBLIC_SANITY_PROJECT_ID=-----------------
NEXT_PUBLIC_SANITY_DATASET=production
API_KEY=your_api_key: ---------------
```

- Uploaded environment variables securely on Vercel.

Step 3: Deploying to Staging Environment

- Deployed the application to Vercel.
- Verified that the deployment was successful by checking the build status and ensuring the website loaded without errors.

Step 4: Staging Environment Testing

- **Functional Testing:**
  Verified key functionalities like product listing, cart updates, and API integration using Cypress and Postman.
- **Performance Testing:**
  Used Lighthouse to analyze website performance, speed, and responsiveness.
- **Security Testing:**
  Ensured all input fields are validated to prevent SQL injections, checked HTTPS usage, and verified API key security.

Step 5: Documentation and GitHub Repository Updates

- Created a `README.md` file summarizing all activities and steps.
- Organized all files into structured folders:
  - `/documents`: Contains reports, testing documents, and project summary.
  - `/src`: Contains source code.
  - `/public`: Contains static assets like images.

---

4. Test Results:

Functional Testing Report

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	Fallback message shown	Fallback message shown	Passed	Handled gracefully.

Test Case ID	Description	Steps	Expected Result	Actual Result	Status	Remarks
TC003	Check cart functionality	Add item to cart > Verify cart updates	Cart updates correctly	Cart updates correctly	Passed	Works as expected.

Performance Testing Report

- **Lighthouse Scores:**
  - **Performance:** 90
  - **Accessibility:** 95
  - **Best Practices:** 92
  - **SEO:** 98
- **GTmetrix Results:**
  - Fully Loaded Time: 1.2s
  - Total Page Size: 1.4MB

---

5. GitHub Repository Structure:

```
CopyEdit
Project Root
├── /documents
│   ├── Day1_Report.pdf
│   ├── Day6_Testing_Report.csv
│   ├── Lighthouse_Report.pdf
├── /src
│   ├── pages/
│   ├── components/
│   ├── styles/
├── /public
│   ├── images/
│   ├── favicon.ico
├── README.md
```

---

6. Deployment Steps (Summary):

1. Connected GitHub repository to Vercel.
2. Configured environment variables in the Vercel dashboard.
3. Deployed the application to staging.
4. Conducted functional, performance, and security tests.
5. Documented results and organized project files for submission.

---

7. Expected Output:

1. A live staging environment for the marketplace: Deployment Link
2. Organized GitHub repository with:
   - README.md file.
   - Test case and performance reports.
   - Proper folder structure.

---

8. Submission Checklist:

Task	Completed (✓/✗)
Deployment Preparation	✓
Staging Environment Testing	✓
Documentation	✓
Form Submission	✓
Final Review	✓