DAY 6 - DEPLOYMENT PREPARATION AND STAGING ENVIRONMENT SETUP

Objective:

The primary goal of Day 6 is to finalize the marketplace project for deployment by ensuring it was fully prepared for a production environment. This phase included conducting in-depth testing to identify and resolve potential issues, optimizing performance for real-world usage, and crafting professional deployment documentation. The documentation aim to provide clear and concise guidance, ensuring a smooth deployment process and delivering an intuitive, seamless user experience. This step is crucial in guaranteeing the project's reliability, usability, and readiness for end-users.

Key Learning Outcomes:

1. Deployment Setup

- **Hosting Platform Selection**: Selected **Vercel** for its powerful features and seamless deployment capabilities.
- **GitHub Integration**: Linked the project's GitHub repository with Vercel, automating the deployment process.
- **Configuration**: Set up build settings, environment variables, and secured API keys on Vercel.
- **Environment Variables**: Configured critical variables such as projectId, dataset, and API-token to enhance security.
- **Validation**: Successfully deployed the application to a production-ready environment.

---

2. Comprehensive Testing

- **Functional Testing**:
    - Utilized **Cypress** for validating user workflows.

    - Tested API responses using **Postman**, ensuring accurate cart management and search functionalities.
- **Performance Testing**:
    - Evaluated page speed, responsiveness, and performance metrics using tools like **Lighthouse**.
- **Security Testing**:
    - Implemented input validation.
    - Enforced HTTPS protocols.
    - Protected sensitive data such as API keys from unauthorized access.
- **Cross-Device Compatibility**:
    - Verified consistent performance across multiple devices and browsers, ensuring responsiveness.
- **Error Handling**:
    - Assessed the system's ability to gracefully handle errors and provide meaningful feedback to users.

---

3. Deployment Strategy

- **Hosting and Backend Integration**:
    - Established seamless communication between the frontend and backend services, including **Sanity CMS** and third-party APIs.
    - Secured environment variables to maintain data integrity and user privacy.
- **Staging Environment**:
    - Deployed the project to a staging environment for pre-production testing.
    - Validated successful builds and ensured all features operated as expected.

---

4. Testing Tools Used

- **Postman**: Validated API responses to ensure reliable communication between components.
- **Lighthouse**: Measured and analyzed key performance metrics to optimize the user experience.