Day 06

# Deployment Preparation And Staging Environment Setup

By: Kulsoom Imran

# Deployment Report

## 1. Deployment Strategy Planning

### Hosting Platform: Vercel

- Chosen for its seamless integration with modern frameworks like Next.js, fast deployment processes, and advanced features such as serverless functions.
- Repository linked from GitHub for automated deployment workflows.

### Backend Interaction

- Integrated with Sanity CMS for content management.
- Utilized third-party APIs for product listings and other function Verified all API endpoints for seamless interaction with the frontend.

## 2. Environment Variable Configuration

**Local .env File:**

Included sensitive variables:

- NEXT_PUBLIC_SANITY_PROJECT_ID
- SANITY_API_TOKEN
- NEXT_PUBLIC_SANITY_DATASET

**Vercel Dashboard Configuration:**

- Uploaded the above variables via Vercel Project Settings under the "Environment Variables" section.
- Ensured variables were added to the correct scope (e.g., Production, Preview, or Development).

## 3. Staging Environment Setup

### Initial Deployment:

- Deployed the application to the staging environment using Vercel's automated build process.
- Verified that the deployment log showed no errors and the application was accessible at the staging URL.

### Validation Checks:

- Ensured that static assets, dynamic routes, and backend data were loading correctly.

## 4. Staging Environment Testing

### Performance Testing

**Tools Used:** Lighthouse

**Key Metrics:**

- **Page Load Time:** Achieved sub-2 second load times for most pages.
- **Performance Score:** Scored 85+ on Lighthouse.

### Security Testing

**Validations:**

- API keys secured using environment variables.

### Responsiveness Testing

Confirmed cross-device compatibility . Tested across:

- Mobile
- Tablets
- Desktops

### Error Handling

Verified proper handling of 404 Errors with custom error pages.