DAY 6 - DEPLOYMENT PREPARATION AND STAGING ENVIRONMENT SETUP

Objective:

Day 6 focuses on preparing your marketplace for deployment by setting up a staging environment, configuring hosting platforms, and ensuring readiness for a customer facing application. Building on the testing and optimization work from Day 5, this stage emphasizes ensuring the marketplace operates seamlessly in a production-like environment. Students will also learn about industry-standard practices for managing different environments like non-production (TRN, DEV, SIT) and production (UAT, PROD, DR).

Step 1: Hosting Platform Setup

1. Why Vercel for Deployment?

Vercel is the official hosting platform for Next.js, offering:

- Easy Integration – Connect your GitHub/GitLab repository for automatic deployments.
- Fast Global CDN – Ensures quick loading times with edge caching.
- Automatic Builds & Previews – Deploys changes automatically from different branches.
- Environment Variable Management – Securely manage secrets like API keys.
- Zero Configuration – No need for complex server setup, just push your code, and Vercel handles the rest.

2. Why Vercel for Deployment?

Since I've chosen Vercel, linking my GitHub repository will enable automatic deployments every time I push changes. Here's how to do it:

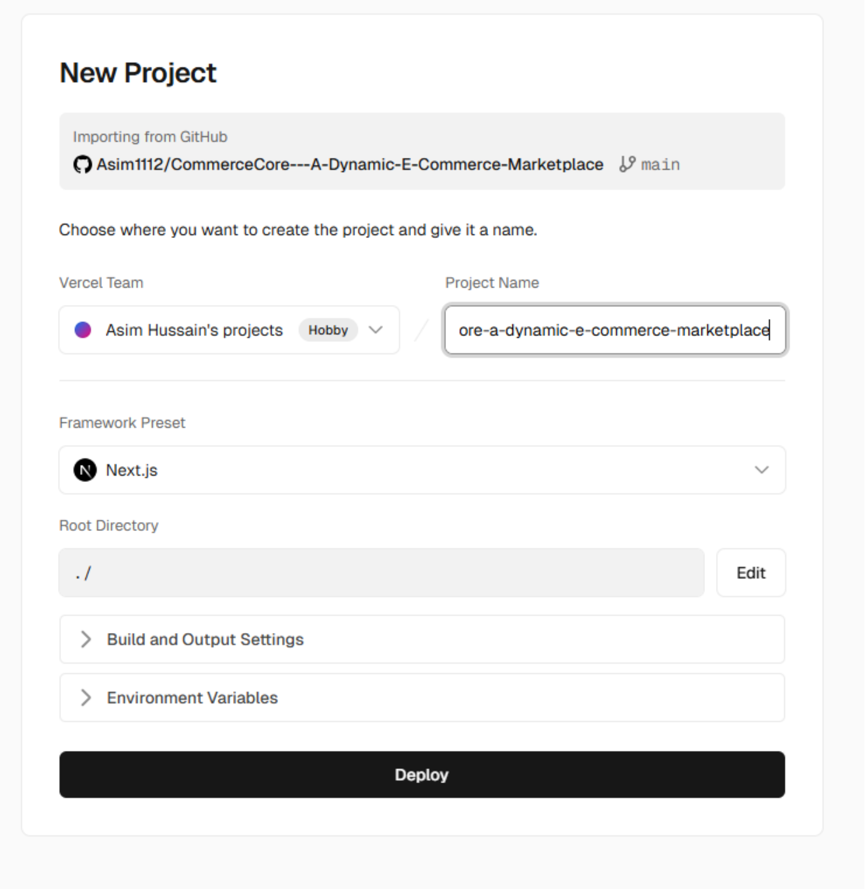
Connecting GitHub to Vercel

Sign in to Vercel & Import Project

Step 2: Configure Environment Variables

1. Create a .env File:

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



New Project

Importing from GitHub
Asim1112/CommerceCore---A-Dynamic-E-Commerce-Marketplace main

Choose where you want to create the project and give it a name.

Vercel Team: Asim Hussain's projects Hobby

Project Name: ore-a-dynamic-e-commerce-marketplace

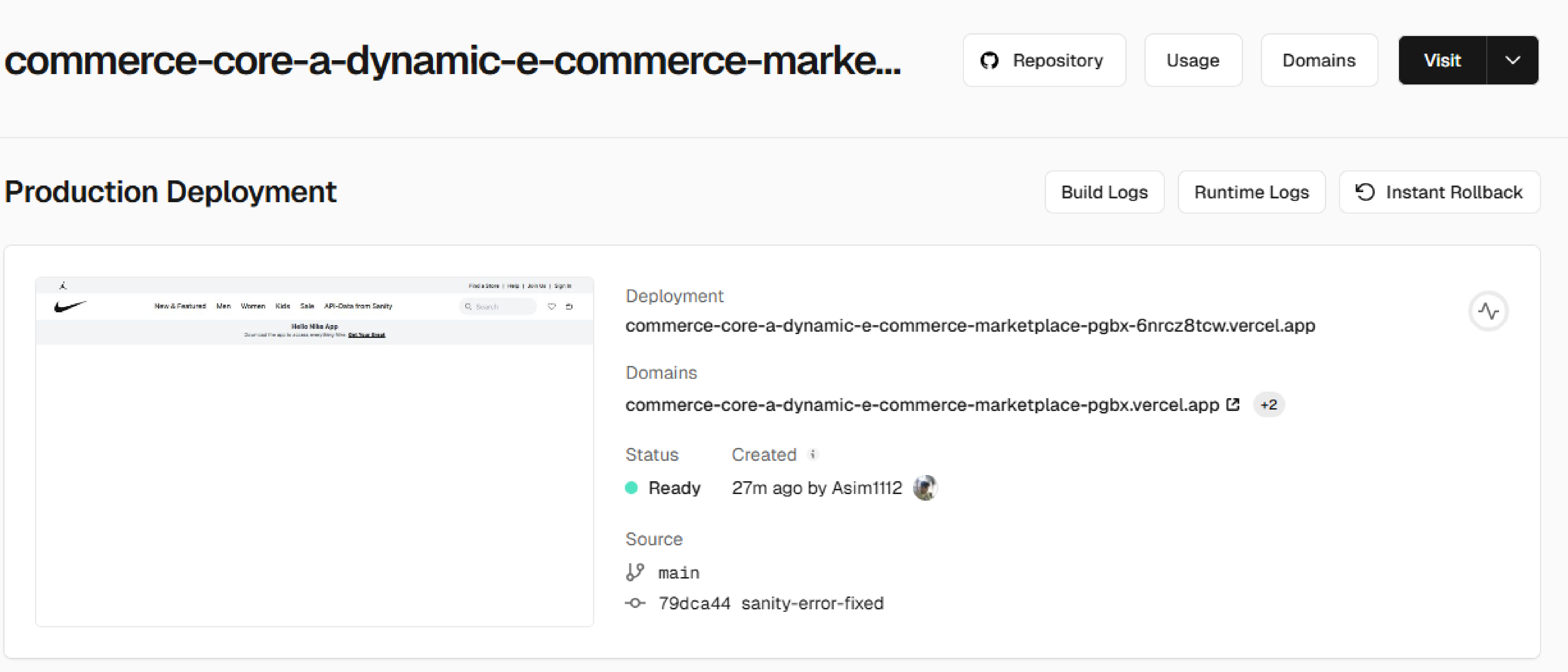
Framework Preset: Next.js

Root Directory: ./

2. **Upload Variables to Hosting Platform:**
   Use the hosting platform's dashboard to securely add environment
   Variables.

Step 3: Deploy Staging

1. **Deploy Application:**
   - Deploy the application to a staging environment through the hosting platform.



commerce-core-a-dynamic-e-commerce-marke...

Production Deployment

Deployment
commerce-core-a-dynamic-e-commerce-marketplace-pgbx-6nrcz8tcw.vercel.app

Domains
commerce-core-a-dynamic-e-commerce-marketplace-pgbx.vercel.app +2

Status: Ready

Created: 27m ago by Asim1112

Source
main
79dca44 sanity-error-fixed

2. **Validate Deployment:**
   - Ensure the build process completes without errors.
   - Verify basic functionality in the staging environment.

Step 4: Staging Environment Testing

1. **Testing Types:**
   - **Functional Testing:** Verify all features, such as product listing, search, and cart operations.
   - **Performance Testing:** Use Lighthouse or GTmetrix to analyze speed and responsiveness.
   - **Security Testing:** Validate input fields, HTTPS usage, and secure API communications

2. **Test Case Reporting:**
   - Document all test cases in a CSV file with fields like Test Case ID, Description, Steps, Expected Result, Actual Result, Status, and Remarks. For example:

Step 5: Documentation Updates

1. **Create README.md:**
   - Summarize all project activities, including deployment steps and test case results.

2. **Organize Project Files:**
   - Ensure all files from Days 1 to 6 are in a structured folder hierarchy.

Expected Output

1. A fully deployed staging environment for the marketplace.
2. Environment variables securely configured.
3. Test case and performance reports documenting staging tests.
4. All project files and documentation organized in a GitHub repository.
5. A professional README.md file summarizing project activities and results.