Day 6 :Deployment and Stagging Environment

**1. Deployment to Production**

**Platform Used:**  
I deployed the application to **Vercel**, a platform optimized for **Next.js** applications. Vercel provides automatic deployment from GitHub repositories, making it easy to push changes to production.

**Steps Taken:**

* Linked the project’s GitHub repository to Vercel.
* Configured environment variables for production, including keys for any third-party services (such as **Sanity CMS** ).
* Tested the deployment to confirm that all pages load properly and all features work as expected, from adding items to the cart to checking out.
* Github link: https://github.com/aimanhanif321/Market-place-Hacthone-.git

**Results:**

* The website is now live and accessible for users.
* Real-time testing has been enabled for monitoring and bug fixes.

**2. Setting Up Staging Environment**

A **staging environment** is crucial for testing new features, bug fixes, and updates before making them live. This helps catch any issues in a controlled environment that mimics the production setup.

**Steps Taken:**

* Created a separate branch in the GitHub repository for staging.
* Configured a dedicated **staging environment** on **Vercel** to deploy changes from the staging branch.
* Used staging-specific environment variables to simulate the production environment but with non-sensitive data.
* Ensured that all features are thoroughly tested in the staging environment before pushing them to production.
* Established a workflow where any changes pushed to the staging branch are automatically deployed to the staging environment.

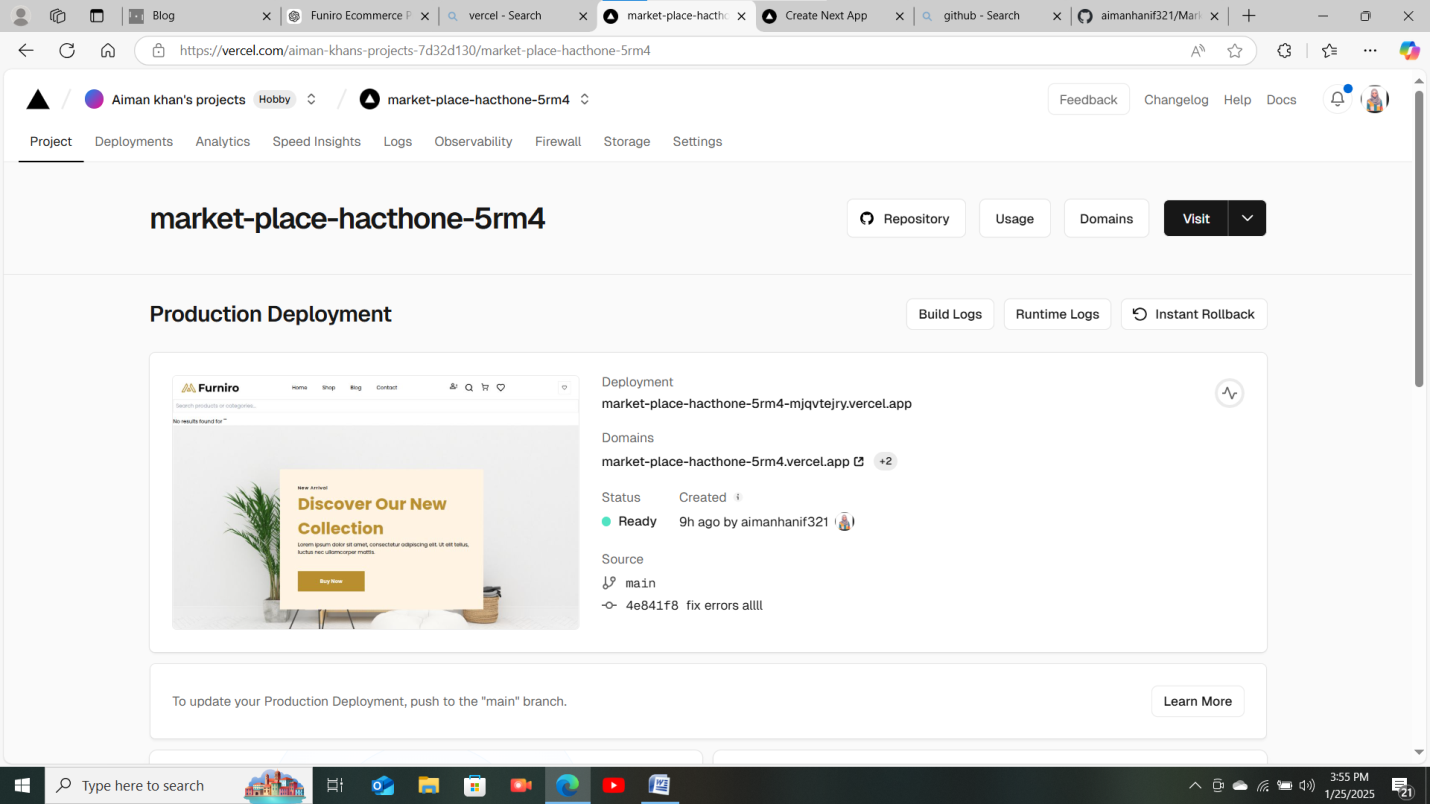
**Results:**

* The staging environment is fully functional, allowing for testing of new features without affecting the production version of the website.
* The deployment process is streamlined, with continuous deployment (CD) set up to handle changes efficiently.

**3. Continuous Integration and Deployment (CI/CD)**

To improve the development workflow, I integrated **CI/CD pipelines** using **GitHub Actions** and **Vercel**. This automation ensures that every time a change is pushed to the staging or production branches, it’s automatically deployed to the corresponding environment.

Vercel link: <https://market-place-hacthone-5rm4.vercel.app/>



**Benefits:**

* Streamlined development and deployment process.
* Reduced risk of human error in deploying code to production.
* Faster feedback loop to catch issues early.

**Conclusion**

The deployment and staging environment setup has been a significant milestone in the development process. The website is now live, and I have a stable staging environment to test future features and updates. With CI/CD pipelines in place, I can ensure that new changes are deployed efficiently and safely, without disrupting the user experience.