Day 6 -Hackathon

Deployment Preparation And Staging Environment Setup

For E-Commerce Website

Introduction:

As part of the project to develop and deploy the e-commerce website, my responsibilities primarily focused on ensuring that the deployment process was efficient, the staging environment was properly configured and everything was set up for seamless testing and deployment to production.

Deployment Strategy Planning

I first worked on planning the deployment strategy to ensure a smooth transition from development to production. This included:

1. Writing Code In VS Code :

I used VS Code to write and modify the code for the e-commerce website . VS Code provided me with the necessary tools and extensions to work efficiently.

2. Version Control With Git :

- I used Git to manage and track the changes made to the codebase.
- After making changes , I would open the terminal and follow these commands:

git add . **:** This command adds all the modified files to the staging area , preparing them for a commit.

git commit -m "message" : I then commit the changes with a message describing what was done.

git push origin main : Finally I pushed the changes to the Github repository, ensuring that the latest updates were saved.

3. Deployment To Versal:

The hosting platform I used Versal.

- For the deployment process, I used versal, a cloud platform that integrates well with GitHub for seamless continuous deployment.
- After pushing the code to GitHub, Versal automatically detected the changes and started the deployment process.
- Versal built and deployed the latest version of the website to the staging or production environment based on the branch that was pushed.

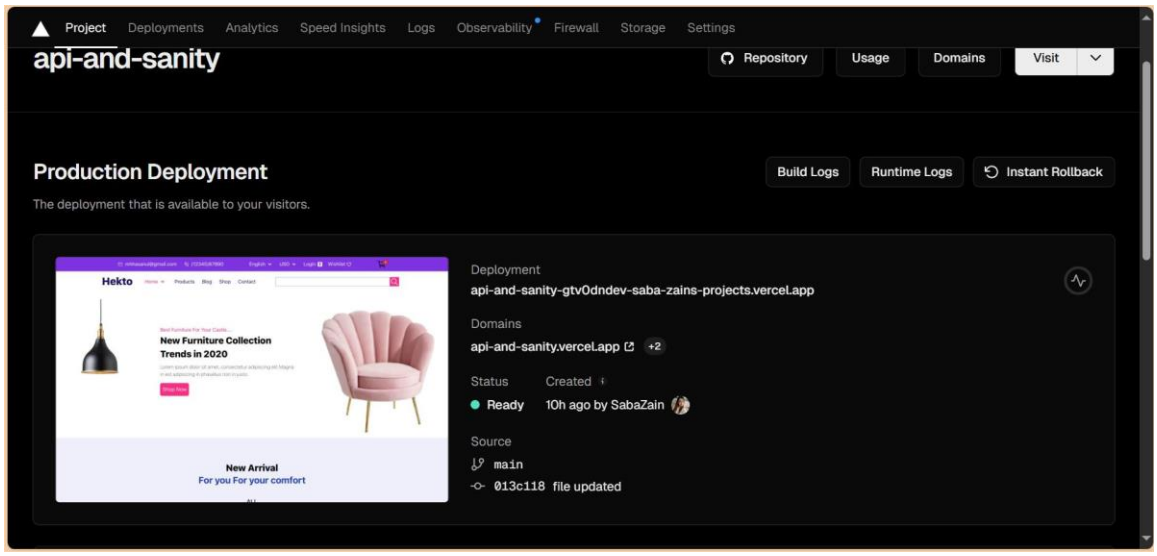
4. Environment Variables Configuration:

- Environment Variables in versal are used to securely store and manage sensitive information, such as API Keys, database credentials or configuration settings that application needs during runtime.
- These variables are set through the versal dashboard and are automatically injected in application during the build and deployment process.
- By using Environment Variables we can avoid hardcoding sensitive data into codebase, ensuring a more secure and flexible deployment process. Additionally it can be customized for different deployment environments(example: staging, production) allowing for tailored configurations without changing the code.

By integrating GitHub with Versal , I ensured that any updates made, versal handles the build and deployment tasks without manual intervention. Additionally, versal supports preview environments , which allow developers to preview changes before pushing them to the production environment, providing an additional layer of safety and confidence before updates go live.

Conclusion:

In conclusion, using Versal in combination with GitHub streamlines the deployment workflow, reduces the risk of human error, and speeds up the release cycle, ensuring that the website or application is always upto date and running smoothly.


Project
Deployments
Analytics
Speed Insights
Logs
Observability
Firewall
Storage
Settings
api-and-sanity
Repository
Usage
Domains
Visit
Production Deployment
The deployment that is available to your visitors.
Build Logs
Runtime Logs
Instant Rollback
Hekto
New Furniture Collection
Trends in 2020
New Arrival
For you For your comfort
Deployment
api-and-sanity-gtv0dndev-saba-zains-projects.vercel.app
Domains
api-and-sanity.vercel.app
+2
Status
Created
Ready
10h ago by SabaZain
Source
main
013c118 file updated