

Marketplace Deployment

Day 06:

The goal of Day 06 is to finalize the deployment of our marketplace. After completing all functionalities in Day 05, this task focuses on securing sensitive information, configuring the deployment environment, and ensuring the marketplace is ready for real-time traffic.

Key Features to Highlight

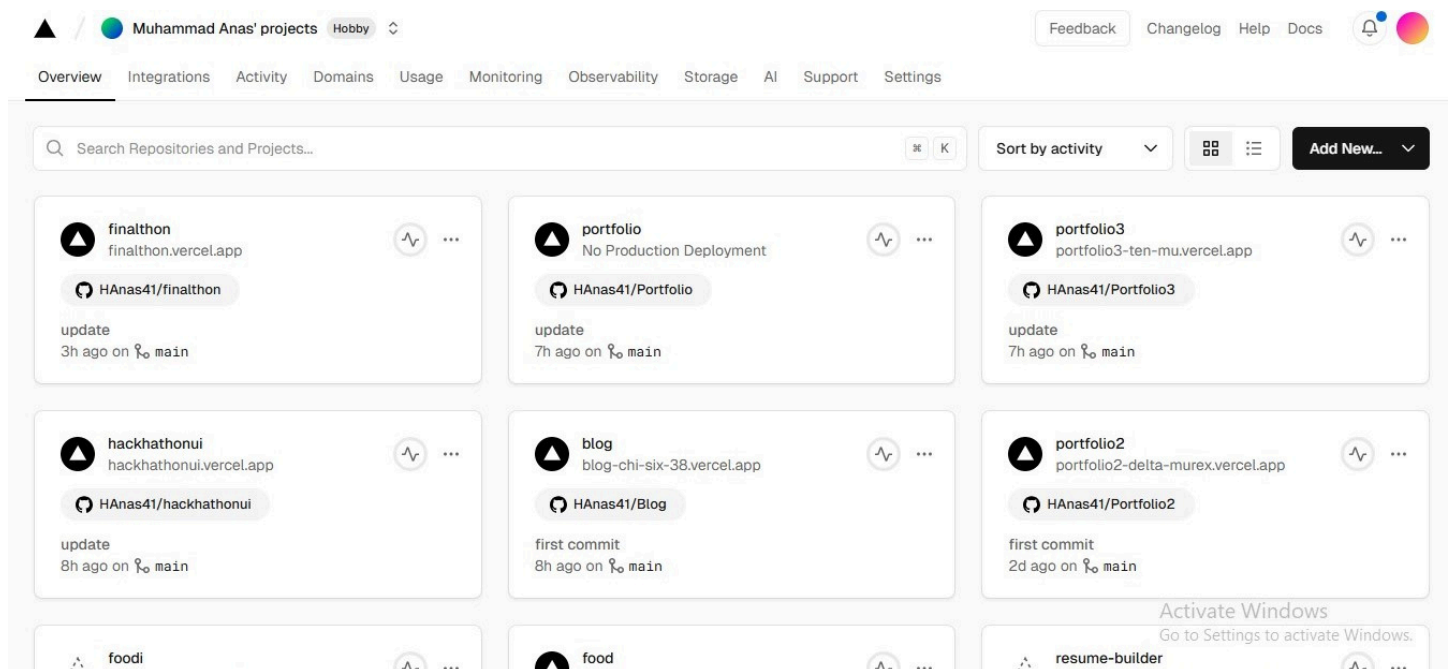
1. Platform for Hosting

For hosting our marketplace, we have chosen Vercel. Vercel is a powerful platform known for its seamless deployment and fast performance, with many services available in the free plan.

Steps to Deploy:

- **Step 1: Create an account on Vercel or link it directly to GitHub.**
- **Step 2: Deploy our marketplace by connecting the project to Vercel.**

Vercel handles most of the configuration automatically, making the deployment process straightforward and efficient.



Configure Environment Variables

A crucial step in securing the marketplace is to protect sensitive data, such as API keys and other private information. We use the .env.local file to store these private details securely.

```
111 .env
1  NEXT_PUBLIC_SANITY_PROJECT_ID="46b5v3pj"
2  NEXT_PUBLIC_SANITY_DATASET="production"
3  NEXT_PUBLIC MOCK_API="https://67a5f4fc5107
4  NEXT_PUBLIC_SANITY_AUTH_TOKEN="skZrodaZJi3
5  NEXT_PUBLIC_WEARISTA="https://finalthon.ve
```

Staging for Deployment

Once the environment variables are configured, we move on to the deployment stage. The process includes:

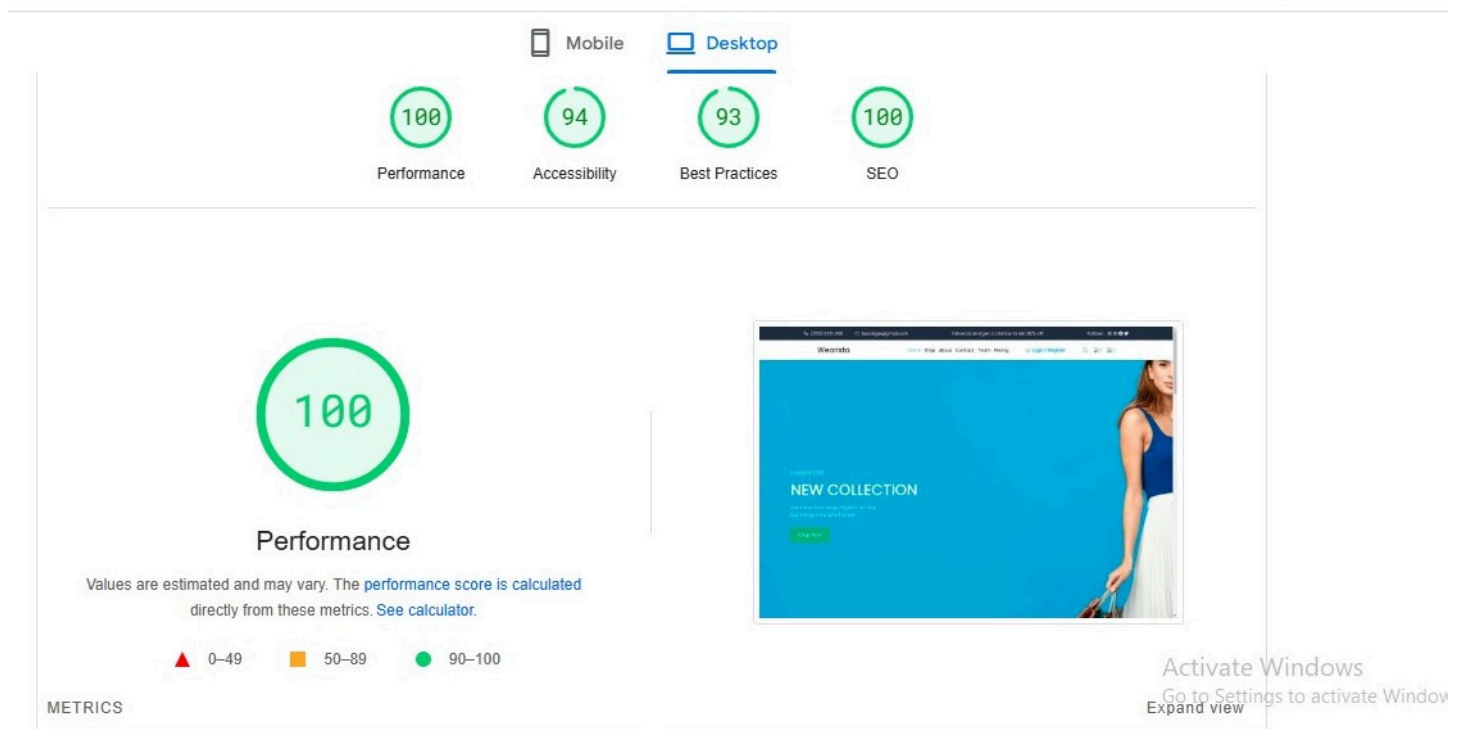
1. **Connecting the Marketplace to Vercel:** We link our marketplace project to Vercel for deployment.
2. **Vercel Deployment Process:** Vercel automatically checks the marketplace for potential issues using tools like ESLint.
3. **Identifying and Resolving Deployment Issues:** If Vercel detects any issues, it provides clear guidance on how to resolve them.
4. **Successful Deployment:** After resolving any issues, the marketplace is deployed live and ready for users.

Testing After Staging

After deployment, we conducted thorough testing to ensure the marketplace operates smoothly.

This included:

- **Functional Testing:** Verified key features such as pagination, search bar, filter operations, cart functionality, and the checkout process.
- **Performance Testing:** Assessed the marketplace's performance using Lighthouse, focusing on load times, accessibility, SEO, and overall user experience.












https://pagespeed.web.dev/analysis/https-finalthon-vercel-app/1ba9ndlpyb?form_factor=desktop&category=performance&category=accessibility&category=best-practices&category=seo&hl=en-US&utm_source=lh-chrome-ext

README

CSV Testing Report for Wearista

This document provides a structured **test report** for Fabric Haven, ensuring functionality, error handling, responsiveness, and user experience.

Test Cases

Test Case ID	Description	Test Steps	Expected Result	Actual Result	Status	Remarks
TC001	 Validate Product Listing	Open product page > Verify product	Product should be displayed	 Product displayed	 Passed	No issues found
TC002	 Test API Error Handling	Trigger Discount API error > Refresh page	Show fallback UI with a user-friendly message	 Fallback UI shown	 Passed	Handled gracefully
TC003	 Cart	Add item to cart > Verify	Cart should	 Cart works		Works as

