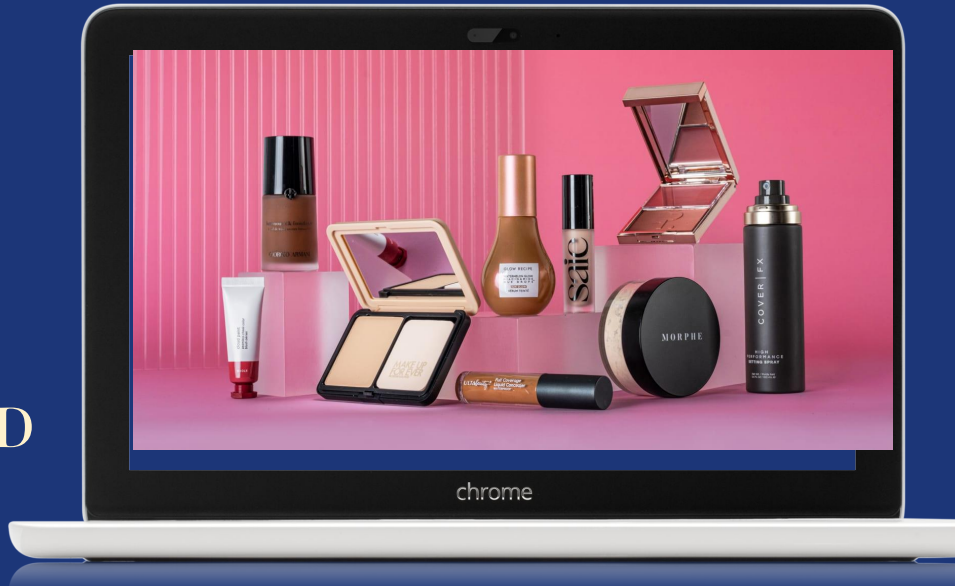Prepared by
Sakeena Majeed

Hackathon Day 6

DEPLOYMENT PREPARATION AND STAGING ENVIRONMENT SETUP

Overview

This document covers the steps required to prepare for deploying my Glow Up Beauty Marketplace website, including setting up a staging environment and conducting detailed tests before the final production deployment. The purpose is to ensure that the application is fully functional, bug-free, and ready for deployment on live servers.

Deployment Preparation and Staging Environment

1. Code Review & Final Adjustments

- Code Review: Ensure clean, well-documented code and remove unnecessary dependencies.
- Version Control: Use Git for version tracking and create a release branch.
- Environment Variables: Verify correct API keys, credentials, and no hardcoded sensitive data.
- Optimizing Assets: Minify files and remove unused assets.
- Cross-Browser Testing: Ensure compatibility across browsers (Chrome, Firefox, Safari).
- Security Checks: Address vulnerabilities like XSS, CSRF, SQL injections, and enable HTTPS.

2. Staging Environment Setup

- Clone Production Environment: Use a staging server with the same setup as production.
- CI/CD Pipeline: Set up CI/CD using tools like Jenkins, GitHub Actions.
- Versioning: Tag releases for version control.

3. Deploying to Staging

- Build the Application: Use npm run build for frontend frameworks.
- Deploy Code to Staging: Push code using Git, Docker, or Kubernetes.
- Database Migration: Apply schema changes to staging database.

Deployment Preparation and Staging Environment

4. Testing Before Final Deployment

- Functional Testing: Verify features like user authentication, product listings, checkout, and admin dashboard.
- Non-Functional Testing: Conduct performance, security, usability, and cross-browser testing.

5. Final Deployment to Production

- Backup Production Data: Backup before deployment.
- Deploy to Production: Use the same process as staging.
- Monitor Deployment: Track errors and performance.
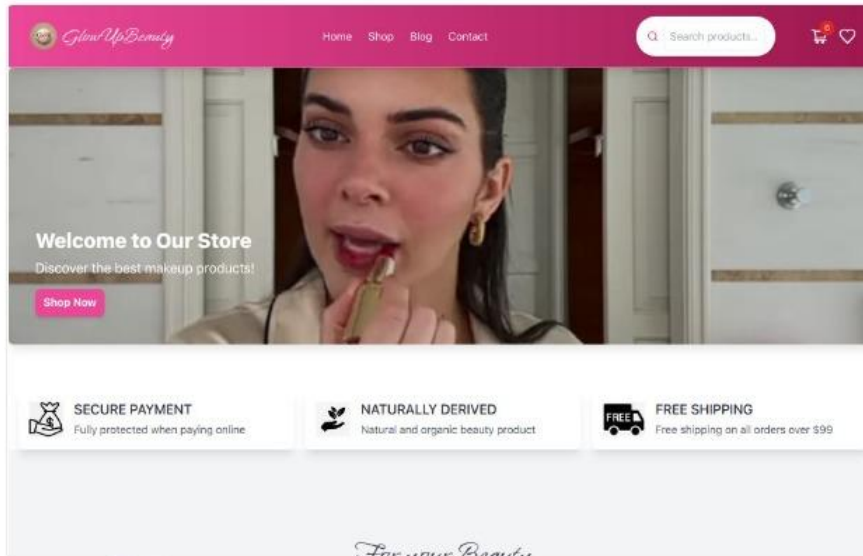- User Communication: Notify users about deployment and release notes.

6. Post-Deployment Testing

- Smoke Testing: Ensure app functionality in production.
- User Feedback: Collect feedback to ensure good user experience.
- Continuous Monitoring: Monitor errors, performance, and user activity.

Congratulations!

You just deployed a new project to 🟣 sakeenamajeed's projects.

Vercel link

https://glow-up-beauty.vercel.app/

Github Link

https://github.com/SakeenaMajeed/GlowUpBeauty

Thank you

By Sakeena Majeed