

HACKATHON 03

DAY 06

Deployment Preparation
and Staging Environment
Setup

Hosting Platform Setup

➤ Chosen Platform:

Vercel was selected for quick and easy deployment.

➤ Connect Repository:

<https://github.com/Anoosharashid/HACKATHON-E-COMMERCE-WEBSITES-UI-UX-FIGMA-TEMPLATE.git>

Configure Environment Variables

➤ Create .env.local File:

- Created the .env.local file to store sensitive data like API keys and tokens.

Example:

```
1 NEXT_PUBLIC_SANITY_PROJECT_ID="vsttmfkr"
2 NEXT_PUBLIC_SANITY_DATASET="production"
3 SANITY_API_TOKEN=skMjIeZueVDZUfGeGVsXrbjMaUUnT5YbbkTT4hSy2SIQK7yQPL5u0byoNTZBcZ5bk1ZdDGG9cdpIJFRy
```

➤ Upload Variables to Vercel:

- Uploaded the environment variables to Vercel using the platform dashboard for secure handling.

Deploy to Staging:

➤ Deploy Application:

- Deployed the application to Vercel's staging environment for testing.

➤ Validate Deployment:

- Ensured the deployment build completed without errors.
- Verified that the application was loading correctly, and all content was fetched properly from Sanity CMS.

Staging Environment Testing

1. Testing Types

➤ Functional Testing:

Verified the following features:

- Product Listing: Ensured all products were listed correctly.
- Product Details: Verified product details page displayed the correct

information.

- User Profile: Checked user login, profile update, and profile display.
- Cart Operations: Ensured products could be added, removed, and quantities updated in the cart.
- Wishlist: Validated the ability to add and remove products from the wishlist.
- Category: Ensured categories displayed correct product listings and filtered accordingly.
- Dynamic Routing: Verified that dynamic routing worked properly for product and Hapages.

Performance Testing:

- Used Lighthouse and GTmetrix to analyze the performance, speed, and responsiveness of the application.
- Ensured the application was optimized for various devices, screen sizes, and network conditions to deliver a smooth user experience.

Security Testing:

- Validated input fields to ensure they were protected from vulnerabilities such as SQL injection and other malicious attacks.
- Ensured HTTPS was enabled for secure communication between the client and server.
- Verified that sensitive data, including API keys and user credentials, was transmitted securely and stored safely to avoid any data breaches.

2. Test Case Reporting:

STEPS:

- | | |
|---|--|
| 1. Open the website | 6. Click the "Update Cart" button on the PDP |
| 2. Open the "Cart" page | 7. Click the "QTY" field |
| 3. Click the "Delete" icon on the product in the product list | 8. Enter any invalid data (-13, qwerq) |
| 4. Click the "Update" icon | 9. Click the "Update Shopping Cart" button |
| 5. Change any data on the Product Details Page (PDP) | 10. Repeat steps 9, 10 with valid data |
| | 11. Click the "Move to wishlist" button |
-

EXPECTED RESULT:

- | | |
|--|---|
| 1. The website is opened | 8. The new invalid data is set |
| 2. The "Cart" page is opened | 9. The error message has appeared. There is no possibility to proceed to checkout |
| 3. The product is deleted. The "Success" message is displayed | 10. The "Success" message is displayed. There is a possibility to proceed to checkout |
| 4. The PDP is opened | 11. The product has disappeared from the "Cart" page. The "Success" message has appeared. The product has been moved to the wishlist. |
| 5. New data is set | |
| 6. The "Success" message is displayed. Product with new set data is displayed on the Cart page | |
| 7. The field is marked | |
-

Performance Testing

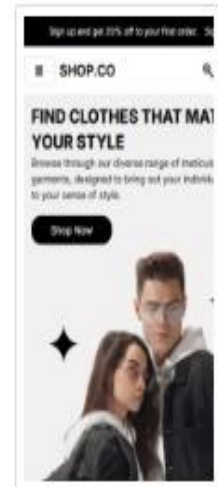
Here is performance report generate by lighthouse tools;



Performance

Values are estimated and may vary. The [performance score is calculated](#) directly from these metrics. [See calculator.](#)

▲ 0–49 ■ 50–89 ● 90–100



Mobile Performance

Desktop performance



Performance



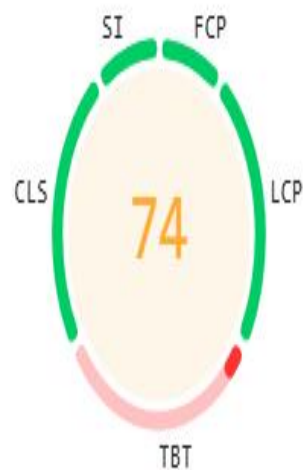
Accessibility



Best
Practices



SEO



Performance

Values are estimated and may vary. The [performance score](#) is calculated directly from these metrics. [See calculator.](#)


▲ 0-49





■ 50-89




● 90-100





Github


 **HACKATHON-E-COMMERCE-WEBSITES-UI-UX-FIGMA-TEMPLATE** Public



 Pin  Unwatch 1  Fork 0  Star 0

















 main  1 Branch  0 Tags

 Add file


 Code


About 


 **Anoosharashid** Add files via upload c4b4943 · 13 hours ago  1 Commit


 (1).gitignore	Add files via upload	13 hours ago
 .gitignore	Add files via upload	13 hours ago
 README(1).md	Add files via upload	13 hours ago
 README.md	Add files via upload	13 hours ago
 components(1).json	Add files via upload	13 hours ago
 components.json	Add files via upload	13 hours ago
 importData(1).js	Add files via upload	13 hours ago
 importData(1).ts	Add files via upload	13 hours ago
 importData.js	Add files via upload	13 hours ago
 importData.ts	Add files via upload	13 hours ago
 next.config(1).mjs	Add files via upload	13 hours ago
 next.config.mjs	Add files via upload	13 hours ago
 package(1).json	Add files via upload	13 hours ago
 package-lock(1).json	Add files via upload	13 hours ago
 package-lock.json	Add files via upload	13 hours ago
 package.json	Add files via upload	13 hours ago


No description, website, or topics provided.

 Readme

 Activity

 0 stars

 1 watching

 0 forks

Releases

No releases published

[Create a new release](#)

Packages

No packages published

[Publish your first package](#)


Languages

TypeScript 61.7%

JavaScript 38.3%

Suggested workflows

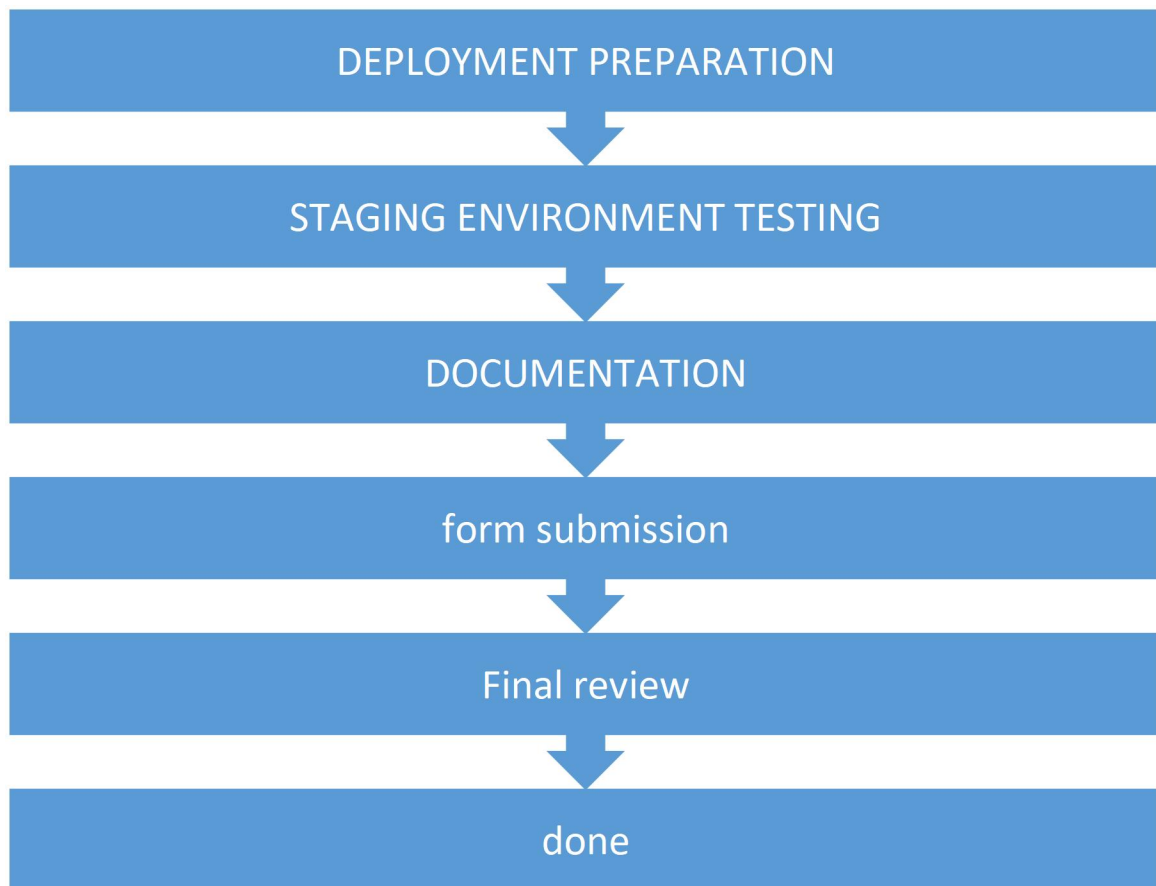
Based on your tech stack

 **SLSA Generic generator** Configure

Generate SLSA3 provenance for your existing release workflows

Conclusion for Deployment Preparation and Staging Step

Day 6 focused on setting up a staging environment for deployment, including configuring environment variables, testing functionality, and updating documentation. This ensures a smooth and secure transition to the live platform, minimizing risks and enhancing readiness for production.



Prepared by: anoosha rashid

Class teacher: sir Muhammad bilal and sir aneeq

