HACKATHON-3 DAY 6

Preparing for Deployment and Setting Up the Staging Environment

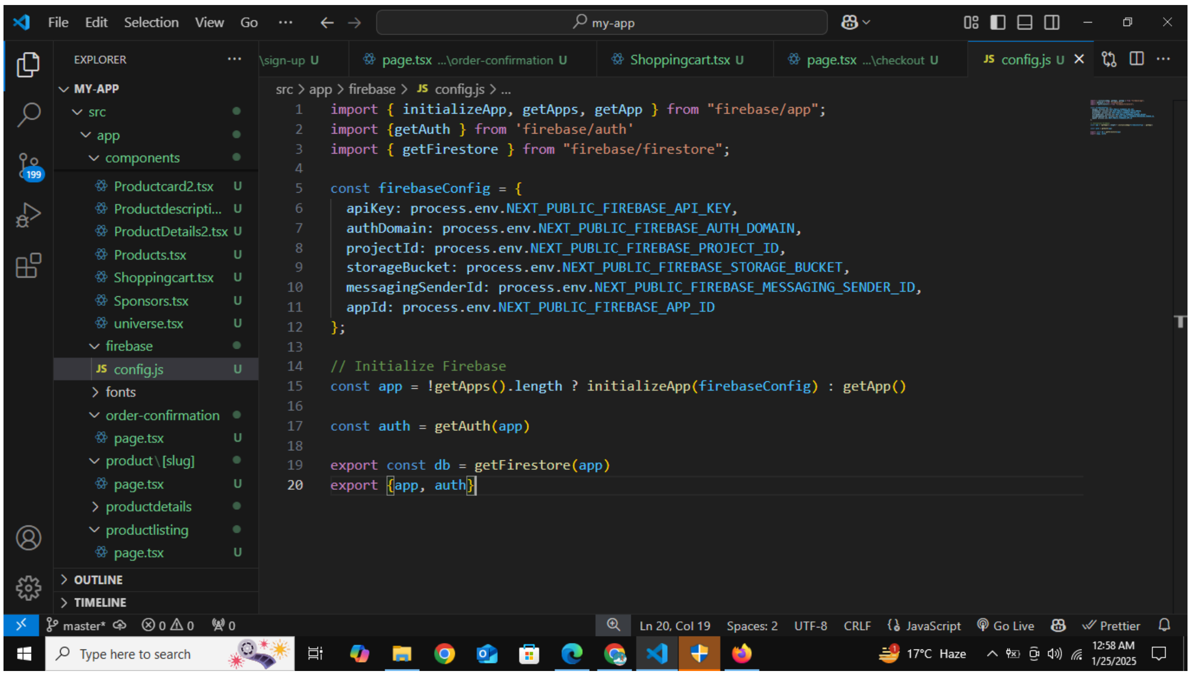
Setting Up the Hosting Platform

I decided to use **Vercel** as the hosting platform for my marketplace application. Vercel stood out because of its user-friendly interface, smooth GitHub integration, and fast deployment features. Here's how I set it up:

1. **Connecting the GitHub Repository:**
   a. Linked my GitHub repository to Vercel.
   b. Adjusted the build settings and added the required deployment scripts.
2. **Configuring Environment Variables:**
   a. Created a `.env` file to securely store sensitive information like API keys and tokens.
   b. Uploaded these environment variables to Vercel's dashboard for secure access during deployment.
3. **Deploying to the Staging Environment:**
   a. Deployed the application to a staging environment using Vercel.
   b. Verified that the build process was successful, and the site loaded without issues.

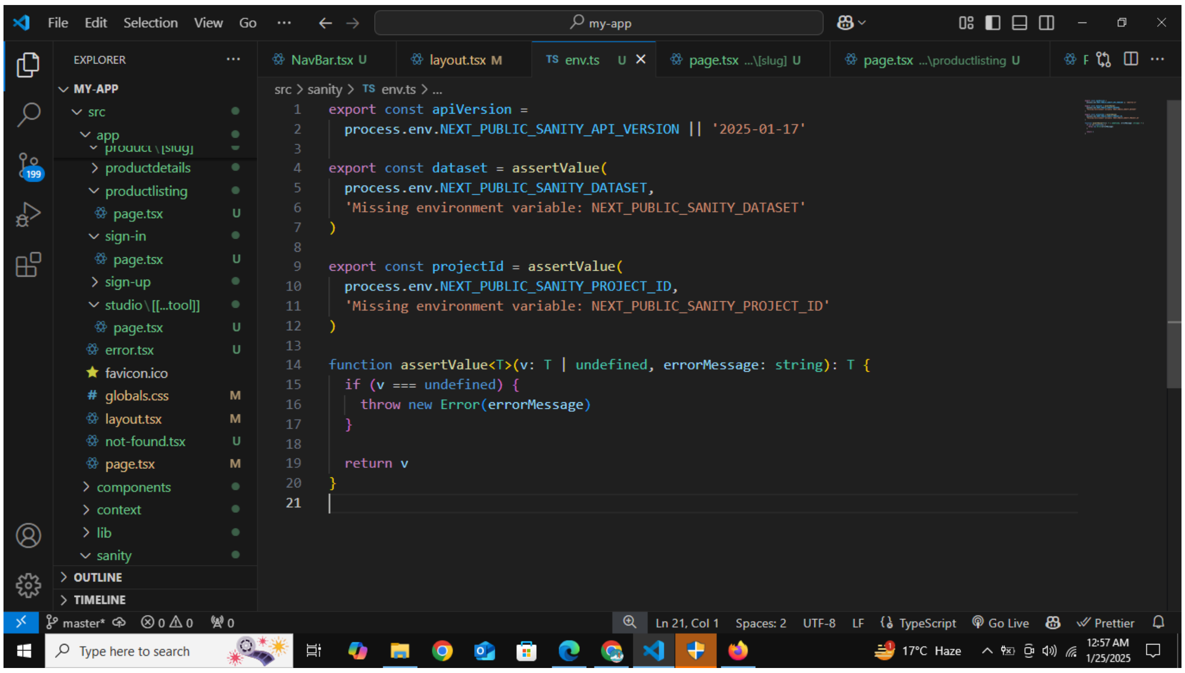
Firebase and Sanity Configuration

Firebase Configuration (JavaScript)



```javascript
import { initializeApp, getApps, getApp } from "firebase/app";
import {getAuth } from 'firebase/auth'
import { getFirestore } from "firebase/firestore";

const firebaseConfig = {
  apiKey: process.env.NEXT_PUBLIC_FIREBASE_API_KEY,
  authDomain: process.env.NEXT_PUBLIC_FIREBASE_AUTH_DOMAIN,
  projectId: process.env.NEXT_PUBLIC_FIREBASE_PROJECT_ID,
  storageBucket: process.env.NEXT_PUBLIC_FIREBASE_STORAGE_BUCKET,
  messagingSenderId: process.env.NEXT_PUBLIC_FIREBASE_MESSAGING_SENDER_ID,
  appId: process.env.NEXT_PUBLIC_FIREBASE_APP_ID
};

// Initialize Firebase
const app = !getApps().length ? initializeApp(firebaseConfig) : getApp()

const auth = getAuth(app)

export const db = getFirestore(app)
export {app, auth}
```

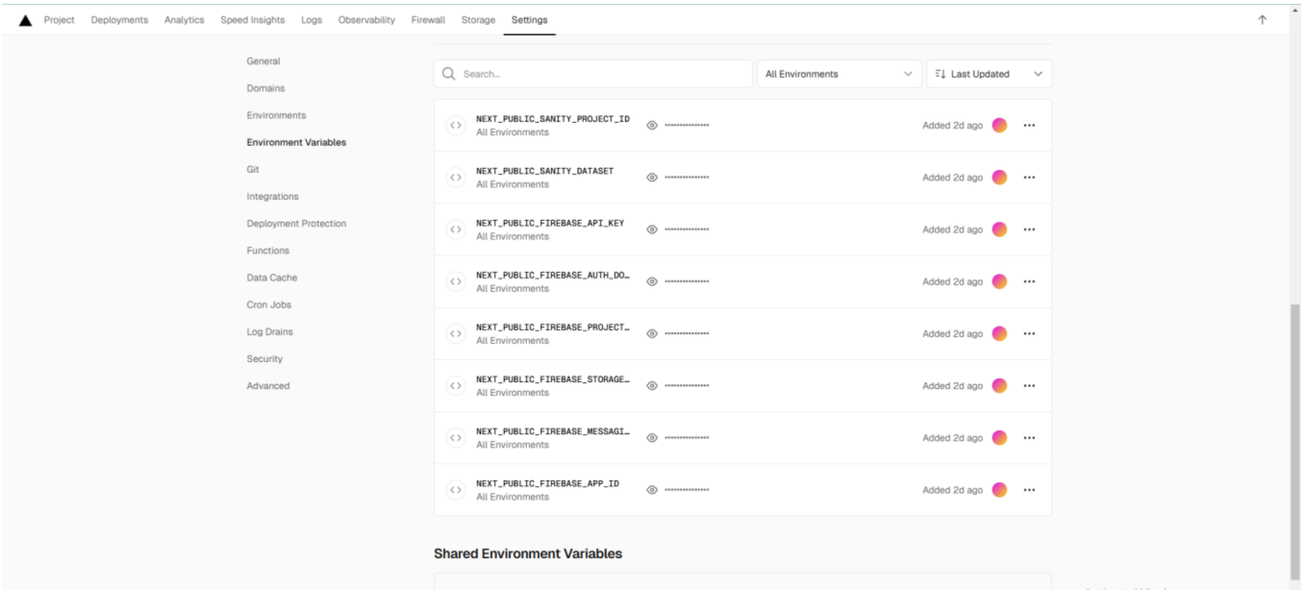
Sanity Configuration (TypeScript)



```typescript
export const apiVersion =
  process.env.NEXT_PUBLIC_SANITY_API_VERSION || '2025-01-17'

export const dataset = assertValue(
  process.env.NEXT_PUBLIC_SANITY_DATASET,
  'Missing environment variable: NEXT_PUBLIC_SANITY_DATASET'
)

export const projectId = assertValue(
  process.env.NEXT_PUBLIC_SANITY_PROJECT_ID,
  'Missing environment variable: NEXT_PUBLIC_SANITY_PROJECT_ID'
)

function assertValue<T>(v: T | undefined, errorMessage: string): T {
  if (v === undefined) {
    throw new Error(errorMessage)
  }

  return v
}
```

Environment Variables in Vercel

The following environment variables were added to Vercel for all environments:



NEXT_PUBLIC_SANITY_PROJECT_ID All Environments	•••••••••••••••	Added 2d ago	•••
NEXT_PUBLIC_SANITY_DATASET All Environments	•••••••••••••••	Added 2d ago	•••
NEXT_PUBLIC_FIREBASE_API_KEY All Environments	•••••••••••••••	Added 2d ago	•••
NEXT_PUBLIC_FIREBASE_AUTH_DO... All Environments	•••••••••••••••	Added 2d ago	•••
NEXT_PUBLIC_FIREBASE_PROJECT... All Environments	•••••••••••••••	Added 2d ago	•••
NEXT_PUBLIC_FIREBASE_STORAGE... All Environments	•••••••••••••••	Added 2d ago	•••
NEXT_PUBLIC_FIREBASE_MESSAGI... All Environments	•••••••••••••••	Added 2d ago	•••
NEXT_PUBLIC_FIREBASE_APP_ID All Environments	•••••••••••••••	Added 2d ago	•••

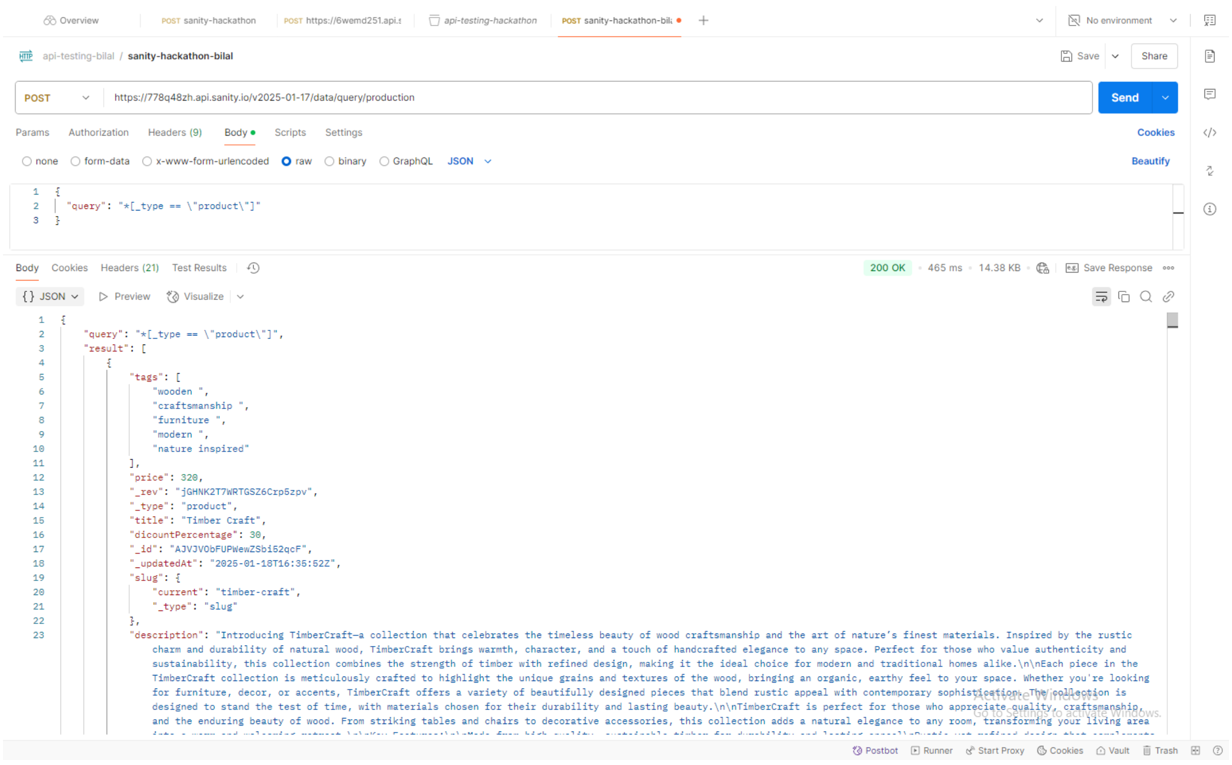
Shared Environment Variables

Testing the Staging Environment

After deploying the application to the staging environment, I conducted extensive testing to ensure it functioned as expected in a production-like setting. The tests included:

1. **Functional Testing:**
    a. **Tools:** Cypress for workflow testing and Postman for API validation.
    b. **Test Cases:** Verified key features like product listings, cart operations, and API error handling.
2. **Performance Testing:**
    a. **Tools:** Lighthouse for analyzing speed, responsiveness, and load times.
    b. **Results:** The application performed well, with acceptable load times and responsiveness.
3. **Security Testing:**
    a. **Validations:** Ensured HTTPS was enabled, input fields were secure, and sensitive data (e.g., API keys) were handled properly.

Postman Api Test



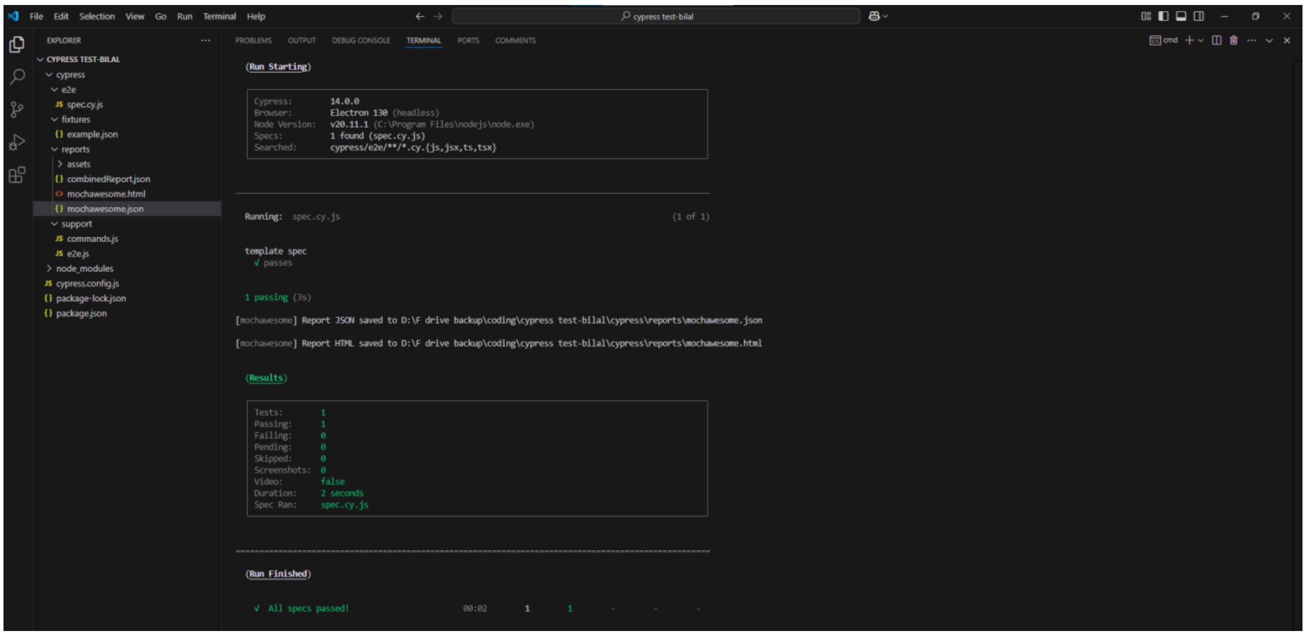
POST https://778q48zh.api.sanity.io/v2025-01-17/data/query/production

none | form-data | x-www-form-urlencoded | [✓ raw] | binary | GraphQL | JSON

```
{
  "query": "*[_type == \"product\"]"
}
```

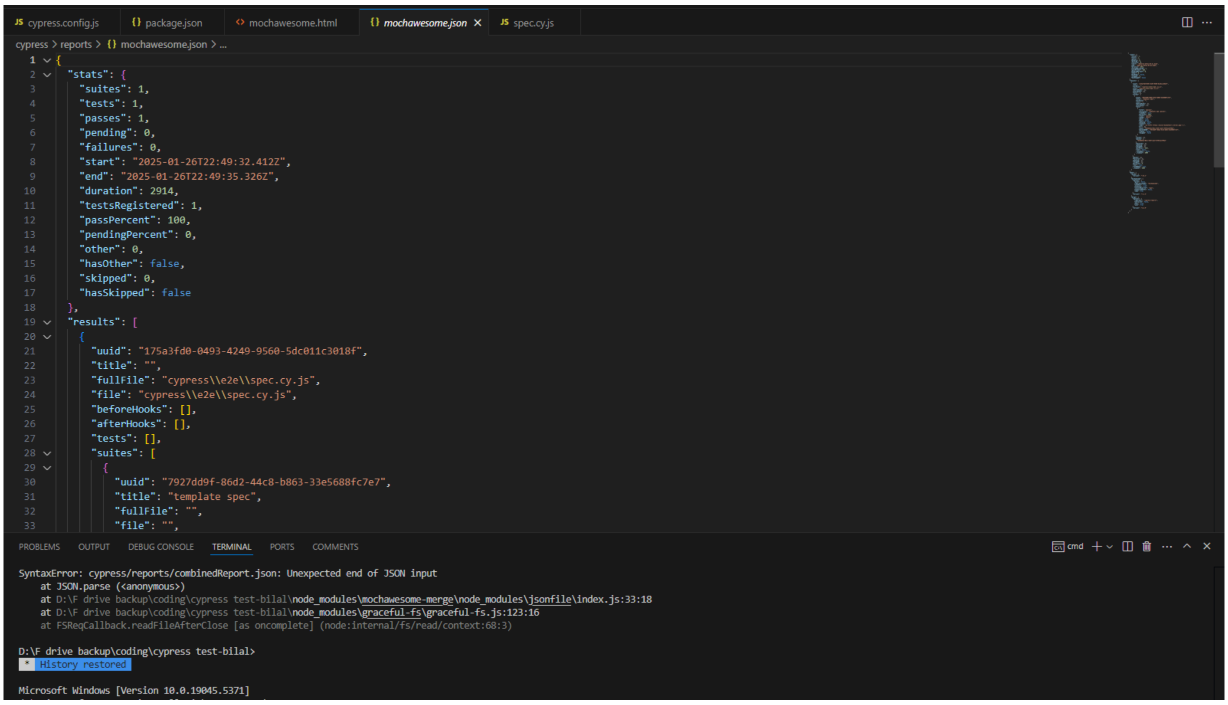
```
{
    "query": "*[_type == \"product\"]",
    "result": [
        {
            "tags": [
                "wooden ",
                "craftsmanship ",
                "furniture ",
                "modern ",
                "nature inspired"
            ],
            "price": 320,
            "_rev": "jGHNK2T7WRTGSZ6Crp5zpv",
            "_type": "product",
            "title": "Timber Craft",
            "dicountPercentage": 30,
            "_id": "AJVJVObFUPWewZSbi52qcF",
            "_updatedAt": "2025-01-18T16:35:52Z",
            "slug": {
                "current": "timber-craft",
                "_type": "slug"
            },
            "description": "Introducing TimberCraft—a collection that celebrates the timeless beauty of wood craftsmanship and the art of nature's finest materials. Inspired by the rustic
                charm and durability of natural wood, TimberCraft brings warmth, character, and a touch of handcrafted elegance to any space. Perfect for those who value authenticity and
                sustainability, this collection combines the strength of timber with refined design, making it the ideal choice for modern and traditional homes alike.\n\nEach piece in the
                TimberCraft collection is meticulously crafted to highlight the unique grains and textures of the wood, bringing an organic, earthy feel to your space. Whether you're looking
                for furniture, decor, or accents, TimberCraft offers a variety of beautifully designed pieces that blend rustic appeal with contemporary sophistication. The collection is
                designed to stand the test of time, with materials chosen for their durability and lasting beauty.\n\nTimberCraft is perfect for those who appreciate quality, craftsmanship,
                and the enduring beauty of wood. From striking tables and chairs to decorative accessories, this collection adds a natural elegance to any room, transforming your living area
```

Cypress Test



```
  (Run Starting)

  ┌────────────────────────────────────────────────────────────────────────┐
  │ Cypress:        14.0.0                                                 │
  │ Browser:        Electron 130 (headless)                                │
  │ Node Version:   v20.11.1 (C:\Program Files\nodejs\node.exe)            │
  │ Specs:          1 found (spec.cy.js)                                   │
  │ Searched:       cypress/e2e/**/*.cy.{js,jsx,ts,tsx}                    │
  └────────────────────────────────────────────────────────────────────────┘

────────────────────────────────────────────────────────────────────────────

  Running:  spec.cy.js                                              (1 of 1)


  template spec
    √ passes


  1 passing (3s)

[mochawesome] Report JSON saved to D:\F drive backup\coding\cypress test-bilal\cypress\reports\mochawesome.json

[mochawesome] Report HTML saved to D:\F drive backup\coding\cypress test-bilal\cypress\reports\mochawesome.html


  (Results)

  ┌────────────────────────────────────────────────────────────────────────┐
  │ Tests:        1                                                        │
  │ Passing:      1                                                        │
  │ Failing:      0                                                        │
  │ Pending:      0                                                        │
  │ Skipped:      0                                                        │
  │ Screenshots:  0                                                        │
  │ Video:        false                                                    │
  │ Duration:     2 seconds                                                │
  │ Spec Ran:     spec.cy.js                                               │
  └────────────────────────────────────────────────────────────────────────┘


============================================================================

  (Run Finished)


    √  All specs passed!                   00:02     1      1      -      -      -
```

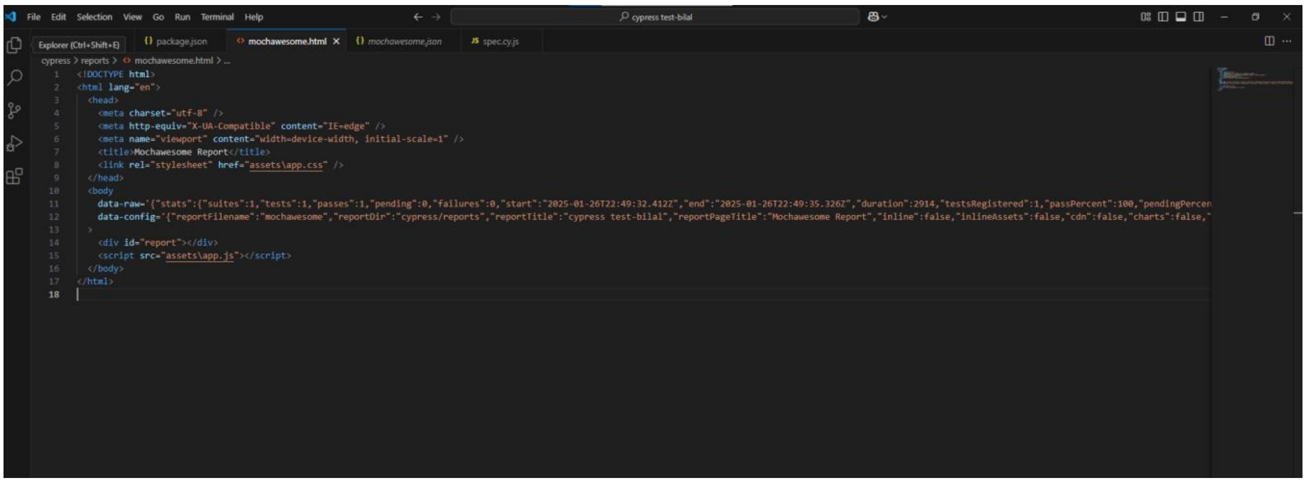
Cypress Test Report Json



```
{
  "stats": {
    "suites": 1,
    "tests": 1,
    "passes": 1,
    "pending": 0,
    "failures": 0,
    "start": "2025-01-26T22:49:32.412Z",
    "end": "2025-01-26T22:49:35.326Z",
    "duration": 2914,
    "testsRegistered": 1,
    "passPercent": 100,
    "pendingPercent": 0,
    "other": 0,
    "hasOther": false,
    "skipped": 0,
    "hasSkipped": false
  },
  "results": [
    {
      "uuid": "175a3fd0-0493-4249-9560-5dc011c3018f",
      "title": "",
      "fullFile": "cypress\\e2e\\spec.cy.js",
      "file": "cypress\\e2e\\spec.cy.js",
      "beforeHooks": [],
      "afterHooks": [],
      "tests": [],
      "suites": [
        {
          "uuid": "7927dd9f-86d2-44c8-b863-33e5688fc7e7",
          "title": "template spec",
          "fullFile": "",
          "file": "",
```

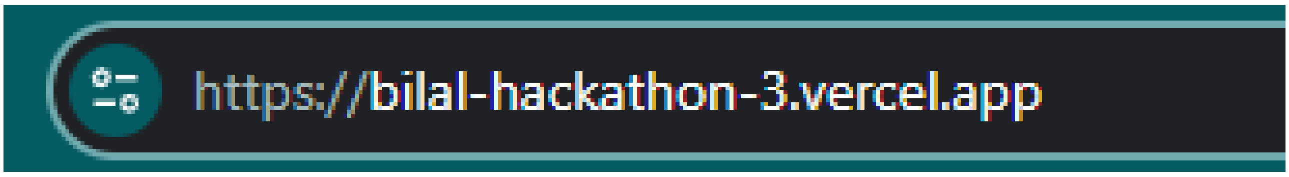
```
SyntaxError: cypress/reports/combinedReport.json: Unexpected end of JSON input
    at JSON.parse (<anonymous>)
    at D:\F drive backup\coding\cypress test-bilal\node_modules\mochawesome-merge\node_modules\jsonfile\index.js:33:18
    at D:\F drive backup\coding\cypress test-bilal\node_modules\graceful-fs\graceful-fs.js:123:16
    at FSReqCallback.readFileAfterClose [as oncomplete] (node:internal/fs/read/context:68:3)

D:\F drive backup\coding\cypress test-bilal>
 *  History restored

Microsoft Windows [Version 10.0.19045.5371]
```

Cypress Test Report Html Format



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <title>Mochawesome Report</title>
    <link rel="stylesheet" href="assets\app.css" />
  </head>
  <body
    data-raw='{"stats":{"suites":1,"tests":1,"passes":1,"pending":0,"failures":0,"start":"2025-01-26T22:49:32.412Z","end":"2025-01-26T22:49:35.326Z","duration":2914,"testsRegistered":1,"passPercent":100,"pendingPercen
    data-config='{"reportFilename":"mochawesome","reportDir":"cypress/reports","reportTitle":"cypress test-bilal","reportPageTitle":"Mochawesome Report","inline":false,"inlineAssets":false,"cdn":false,"charts":false,"
  >
    <div id="report"></div>
    <script src="assets\app.js"></script>
  </body>
</html>
```

Https Validation

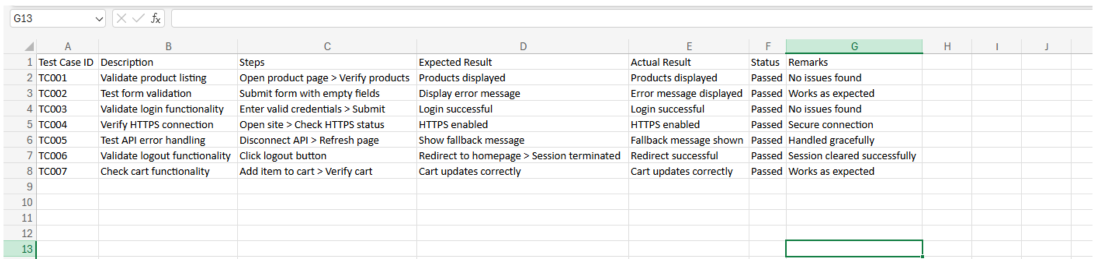


https://bilal-hackathon-3.vercel.app

Documentation Updates

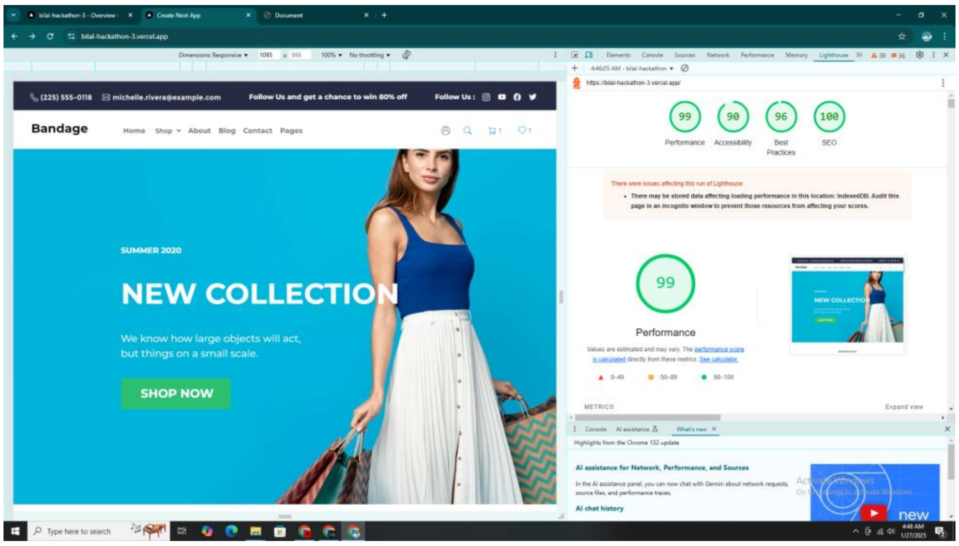
1. **README.md File:**
   a. Created a comprehensive README.md file summarizing the six days of activities, including deployment steps, test results, and the project structure.
   b. Provided a clear folder hierarchy in the GitHub repository (e.g., `documents/`, `src/`, `public/`).
2. **GitHub Repository:**
   a. Organized all project files and documents in a structured manner.
   b. Included:
      i. Test case reports in CSV format.
      ii. Performance testing results generated by Lighthouse.

CSV Report



Test Case ID	Description	Steps	Expected Result	Actual Result	Status	Remarks
TC001	Validate product listing	Open product page > Verify products	Products displayed	Products displayed	Passed	No issues found
TC002	Test form validation	Submit form with empty fields	Display error message	Error message displayed	Passed	Works as expected
TC003	Validate login functionality	Enter valid credentials > Submit	Login successful	Login successful	Passed	No issues found
TC004	Verify HTTPS connection	Open site > Check HTTPS status	HTTPS enabled	HTTPS enabled	Passed	Secure connection
TC005	Test API error handling	Disconnect API > Refresh page	Show fallback message	Fallback message shown	Passed	Handled gracefully
TC006	Validate logout functionality	Click logout button	Redirect to homepage > Session terminated	Redirect successful	Passed	Session cleared successfully
TC007	Check cart functionality	Add item to cart > Verify cart	Cart updates correctly	Cart updates correctly	Passed	Works as expected

Light House Report



There were issues affecting this run of Lighthouse

- There may be stored data affecting loading performance in this location: IndexedDB. Audit this page in an incognito window to prevent those resources from affecting your scores.

99
Performance

Values are estimated and may vary. The performance score is calculated directly from these metrics. See calculator.

▲ 0–49 ■ 50–89 ● 90–100

Expected Outcomes

1. **Staging Environment:**
    a. The application is fully deployed to the staging environment on Vercel.
    b. Environment variables are securely configured.
2. **Test Case and Performance Reports:**
    a. All test cases (passed or failed) are documented in a CSV file.
    b. Performance testing results are included in the GitHub repository.
3. **GitHub Repository:**
    a. All project files and documentation are well-organized and accessible.
    b. A professional README.md file summarizes the project activities and results.

(CSV in text format in the next page)

CSV report in text:

Test Case ID,Description,Steps,Expected Result,Actual Result,Status,Remarks

TC001,Validate product listing,Open product page > Verify products,Products displayed,Products displayed,Passed,No issues found

TC002,Test form validation,Submit form with empty fields,Display error message,Error message displayed,Passed,Works as expected

TC003,Validate login functionality,Enter valid credentials > Submit,Login successful,Login successful,Passed,No issues found

TC004,Verify HTTPS connection,Open site > Check HTTPS status,HTTPS enabled,HTTPS enabled,Passed,Secure connection

TC005,Test API error handling,Disconnect API > Refresh page,Show fallback message,Fallback message shown,Passed,Handled gracefully

TC006,Validate logout functionality,Click logout button,Redirect to homepage > Session terminated,Redirect successful,Passed,Session cleared successfully

TC007,Check cart functionality,Add item to cart > Verify cart,Cart updates correctly,Cart updates correctly,Passed,Works as expected