

Day 6 - Deployment Preparation and Staging Environment Setup (Morent Car Rental Service)

Day 6 focuses on preparing my marketplace for deployment by setting up a staging environment, and ensuring readiness for a customer facing application. Building on the testing and optimization work from Day 5, this stage emphasizes ensuring the marketplace operates seamlessly in a production-like environment.

I. Hosting Platform Setup

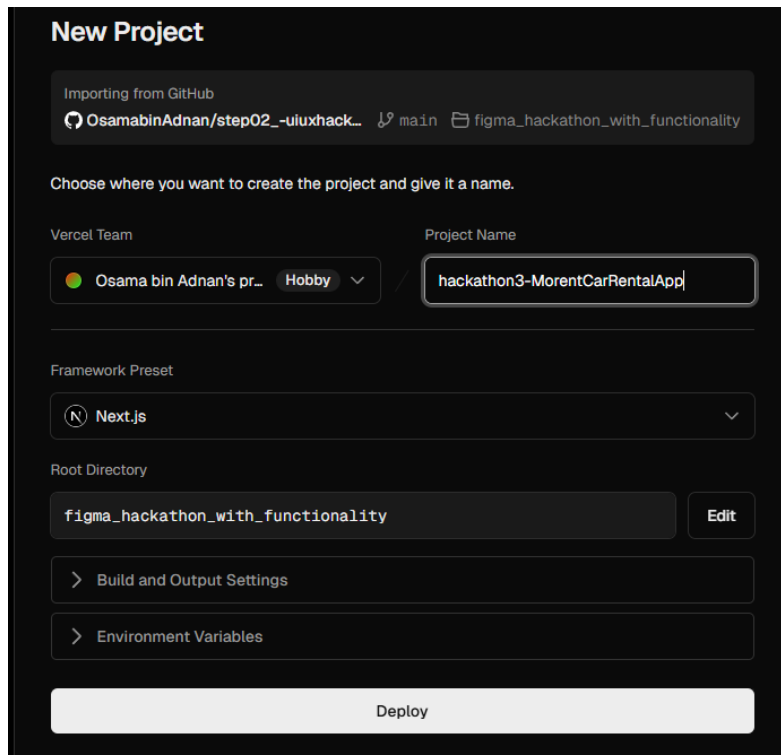
A. Choose a Platform:

Vercel is a user-friendly hosting platform designed for modern web applications, particularly those built with frameworks like Next.js, React, or Vue.js. It's perfect for quick deployments with minimal setup.

a) Why choose Vercel:

- Easy setup with GitHub/GitLab/Bitbucket repositories.
- Automatic build and deployment on every commit.
- Free tier with good limits for small projects.
- Optimized for front-end applications.

B. Connect repository



The screenshot shows the 'New Project' setup page on Vercel. At the top, it says 'Importing from GitHub' and shows a repository 'OsamabinAdnan/step02_-uluxhack...' with branches 'main' and 'figma_hackathon_with_functionality'. Below this, it prompts to 'Choose where you want to create the project and give it a name.' The 'Vercel Team' dropdown is set to 'Osama bin Adnan's pr...' with a 'Hobby' plan. The 'Project Name' field contains 'hackathon3-MorentCarRentalApp'. The 'Framework Preset' is set to 'Next.js'. The 'Root Directory' is 'figma_hackathon_with_functionality'. There are expandable sections for 'Build and Output Settings' and 'Environment Variables'. A 'Deploy' button is at the bottom.

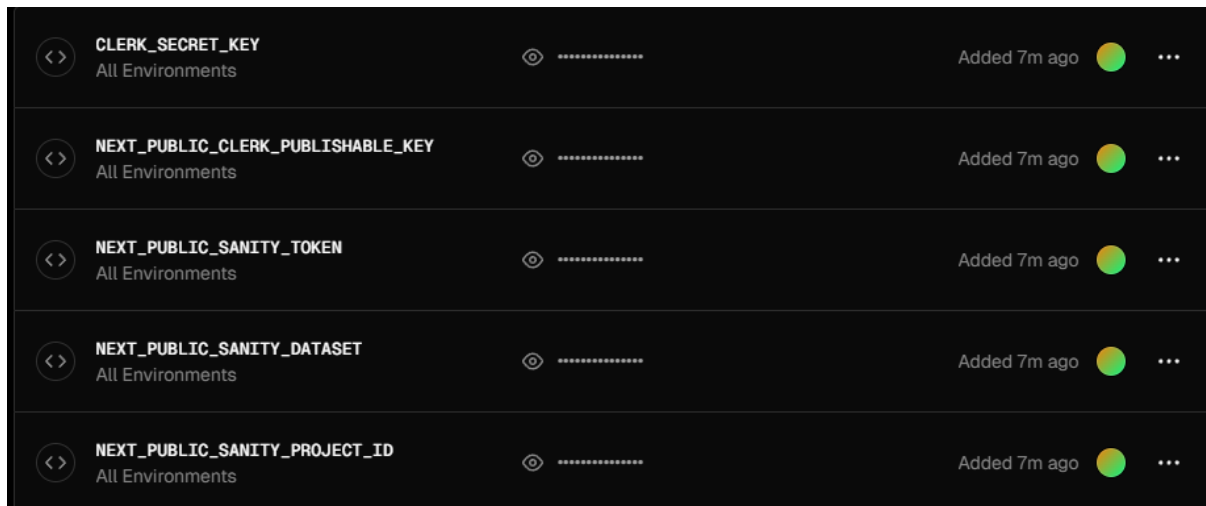
II. Configure Environment Variables

Environment Variables Used: I created a .env file to securely store the API keys, database credentials, and other sensitive information.

Steps Taken: I created the .env file in the root of the project folder. I added variables such as:

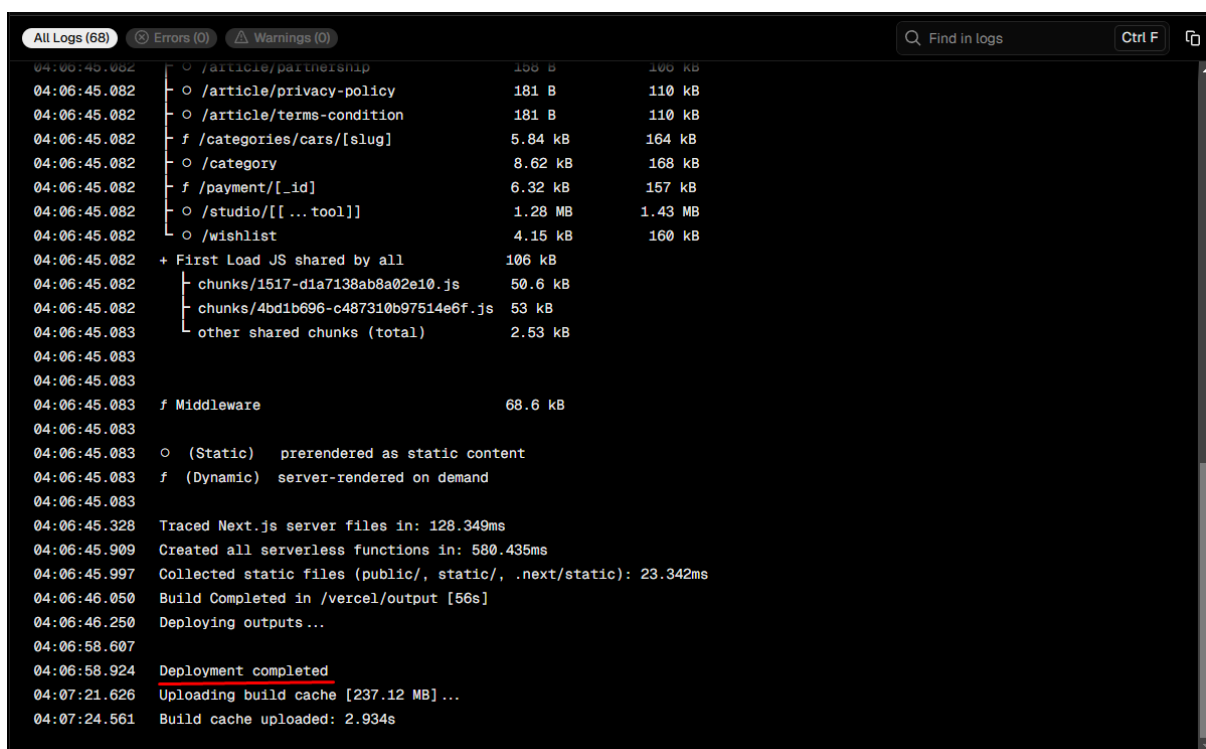
```
NEXT_PUBLIC_SANITY_PROJECT_ID=my_project_id
NEXT_PUBLIC_SANITY_DATASET=production
SANITY_API_TOKEN=my_api_key
```

I uploaded the environment variables through **Vercels dashboard** to ensure secure deployment.



III. Deploy to Staging

First, I run the command in terminal **npm run build** to see if there is any error or not, once I am satisfied, I triggered the deployment by clicking the Deploy button on **Vercel**.



IV. Staging Environment Testing

A. Testing Types

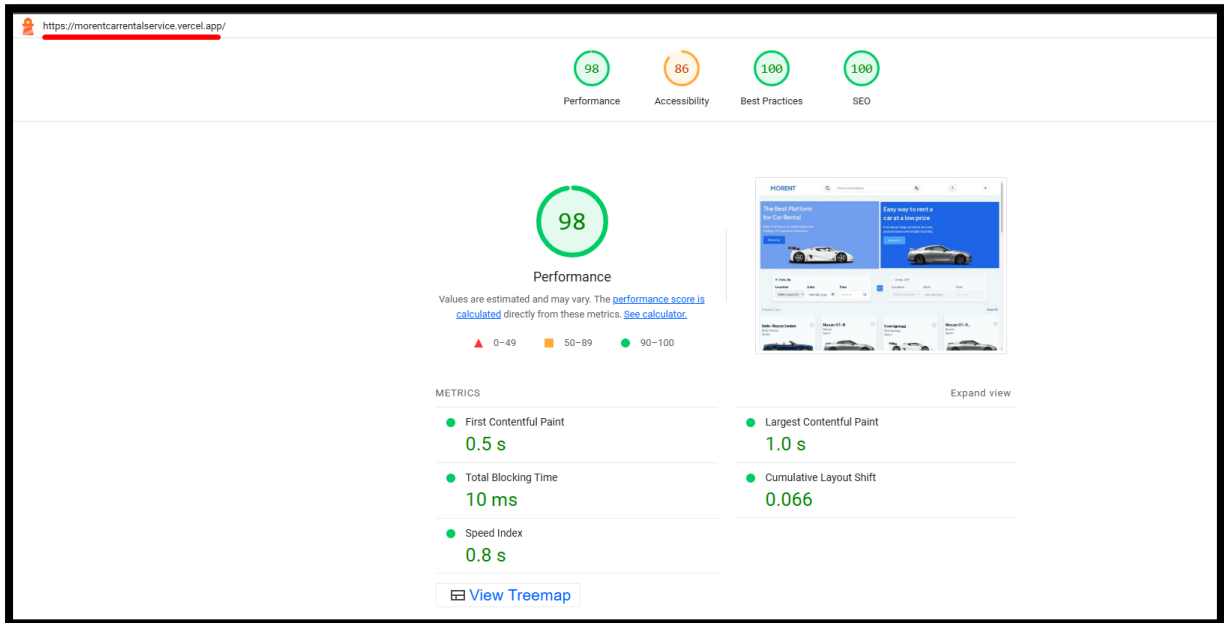
a) Functional Testing:

I used manual testing to verify features like the cars listing page, car detail page etc.

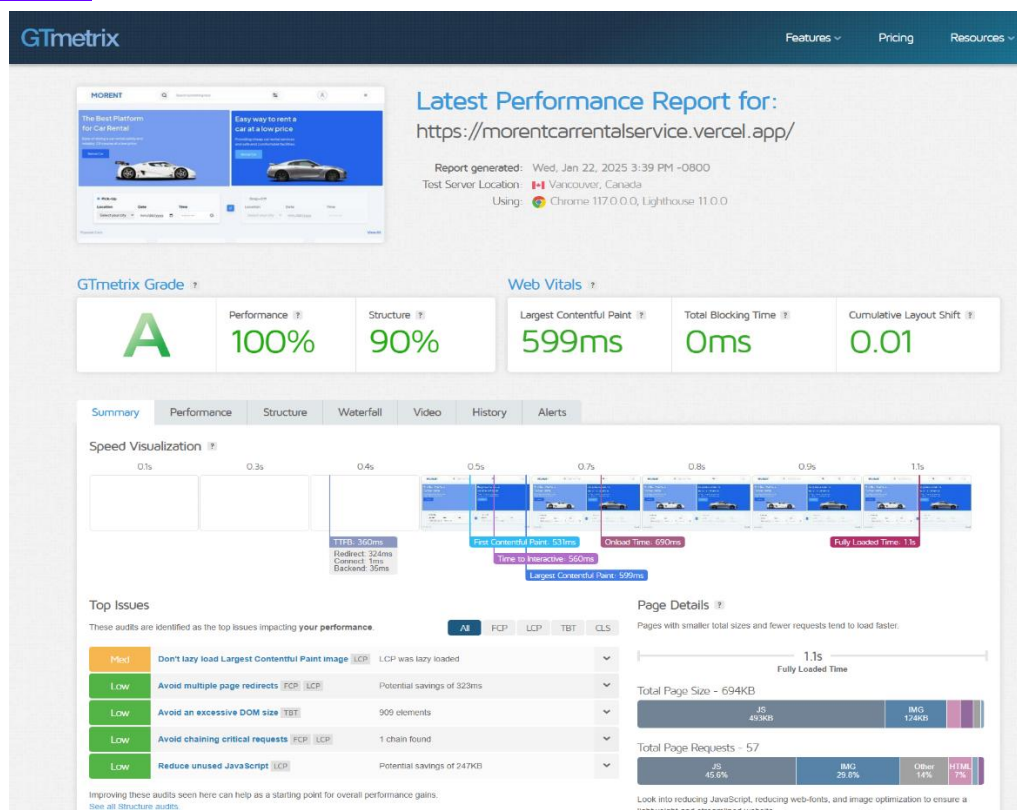
b) Performance Testing:

I used **Lighthouse** to analyse the apps speed and responsiveness and **GTMetrix** for additional performance insights.

Lighthouse

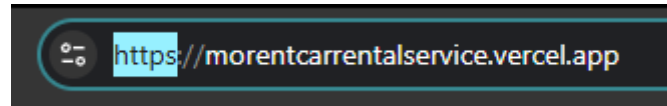


GTmetrix



c) Security Testing

As rental app deployed on vercel, now http status is now secured 'https'.



B. Test Case Reporting

- Documented all test cases in a CSV file with fields like Test Case ID, Description, Steps, Expected Result, Actual Result, Status, and Remarks

Test Case ID	Test Case Description	Test Steps	Expected Results	Actual Result	Status	Severity Level	Assigned to:	Remarks
TC001	Test navigation links	Open app => check all links	All links navigate correctly	All links navigate as expected	Passed	Low	-	No issue Found
TC002	Verify product listing display	Open car page => Verify cars	Cars display correctly	Cars display as expected	Passed	Low	-	No issue Found
TC003	Check filter on category page	Open category page ==> apply filter	Cars display correctly	Cars display as expected	Passed	Medium	-	No issue Found
TC004	Test API	Check API via PostAPI	Data fetch from API	Car data fetched from API	Passed	Medium	-	No issue Found
TC005	Payment Page Car fetch using car id	Click on car which user wants to get on rent	User's car data fetch on payment page	User's car data fetch on payment page	Passed	High	-	No issue Found
TC006	Responsiveness on Mobile	Resize browser layout => check layout	layout adjust according to screen size	layout works as intended	Passed	Medium	-	Work as expected
TC007	Validate payment gateway	initiate payment ==> verify transaction	Still working on it	Still working on it	Failed	High	-	-
TC008	Login page validation	login/signup using credentials	Still working on it	Still working on it	Failed	High	-	-

V. Documentation Updates

Day 6 focused on preparing for deployment and ensuring the application performs as expected in a staging environment. This phase is crucial to delivering a reliable and efficient final product. The screenshots attached further illustrate the successful execution of each step in the deployment process.

Create README.md: Summarized all project activities, including deployment steps and test case results.

Ensured all files from Days 1 to 6 are in a structured folder hierarchy.

A. Folder structure

You can find all documents on below link, see folder structure below

https://github.com/OsamabinAdnan/step02_-uiuxhackathon_with-functionality

Click on above link, you will observe below structure of folder

documentation (main folder)

- day01 laying the foundation for marketplace (folder)
- day02 planning the technical foundation (folder)
- day03 api integration and data migration (folder)
- day04_dynamic_frontend_component (folder)
- day05_testing_error_handling (folder)
- day06_deployment_preparation_&_staging_environment_setup (folder)

figma_hackathon_with_functionality (main folder)

- project code