DAY 06: DEPLOYMENT PREPARATION AND STAGING ENVIRONMENT SETUP

Step 01: Hosting Platform Setup.

Choose Platform:

 Select Vercel for deployment as it provides a seamless, efficient, and scalable solution for hosting Next.js applications.

• Connect Repository:

 Link project repository, such as **GitHub**, and integrate it with **Vercel** for automated deployments and version control management.

Step 02: Configure Environment Variables.

- Create a ".env.local" file and upload variables to hosting platform:
 - NEXT_PUBLIC_SANITY_PROJECT_ID = ✓
 - NEXT_PUBLIC_SANITY_DATASET =
 - API_KEY = ✓
 - CLERK_SECRET_KEY = ✓

Step 03: Deploy to Staging.

- Conducted functional tests for booking, availability search, and pricing operations.
- **Performed basic performance checks** using browser developer tools for smooth navigation.

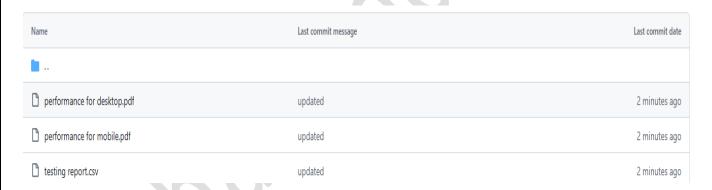
Conducted security testing for input validation, secure payment processing, and API communication.

Name: Muhammad Awais (Senior Student)

• Testing Report.

A	В	C	D	E	F	G	Н	I
Test Case ID	Test Case Description	Test Steps	Expected Result	Actual Result	Status	Severity Level	Assigned To	Remarks
TC001	Car Listing	Home Page > Verify Car Listing	Cars should be displayed correctly	All cars are displayed	Passed	Low		No issues found
TC002	Dynamic Page	Click on a car for details	Dynamic route should work properly	Dynamic route works properly	Passed	Low	-	No issues found
TC003	Add to Whishlist	Click on the heart icon of a car	Car should be successfully added to the wishlist	Car successfully added to the wishlist	Passed	Low	-	No issues found
TC004	Search Bar	Click on the search bar in the header	Search cars by their name or type	Cars are successfully displayed	Passed	Low	-	No issues found
TC005	Filter Bar	Select "SUV" type in the filter bar	"SUV" cars should be displayed successfully	"SUV" cars are displayed successfully	Passed	Low	-	No issues found
TC006	Responsive Design	Test on desktop, tablet, and mobile	Design should be responsive on all devices	Works on desktop and mobile, not on tablet	Failed	Medium	-	Fix tablet layout
TC007	Cross-browser compatibility test	Test the website on Chrome, Firefox, and Edge	Website should render correctly on all browsers	Website displayed correctly on all browsers	Passed	Medium	-	Verified on multiple browsers
TC008	Test data submission to Sanity	Enter car details in the form > Click submit	Data should be successfully entered into Sanity and shown in the browser console	Data successfully entered into Sanity and logged in the console	Passed	low	-	No issues found
TC009	Test "Rent Now" flow and login	Click "Rent Now" > Redirect to sign-in page > Login > Redirect to billing page	User should be redirected to the sign- in page, logged in, and then redirected to the billing page	User was redirected to sign-in, logged in, and redirected to billing page	Passed	Low		No issues found
TC010	Error handling	Disconnect API > Load cars page	Show fallback UI with error message	Error message shown successfully	Passed	low	-	No issues found

• Performance Report Uploaded on Github.



THANK YOU