Name: Aditya Adinath Kirtane Class: D15C/ Batch B Roll No:26

Experiment 11

Aim: To use google Lighthouse PWA Analysis Tool to test the PWA functioning.

Theory:

Google Lighthouse : Google Lighthouse is a powerful tool for auditing Progressive Web Apps (PWAs) by evaluating various aspects of a web application, including performance, accessibility, best practices, and PWA-specific features. To implement Lighthouse for testing a PWA, you first need to ensure that your web app is PWA-compliant, which involves setting up a manifest file, service worker, and offline functionality. Once the PWA is ready, the Lighthouse audit can be run using Google Chrome's built-in Developer Tools or via the Lighthouse CLI. To do this, open your web app in Chrome, access the Developer Tools, and navigate to the Lighthouse tab, where you can select various audit categories such as Performance, PWA, Accessibility, and Best Practices.

The audit generates a detailed report highlighting scores for each metric. Performance is assessed based on loading speed, time to interactive, and content display. The PWA score evaluates the app's alignment with Google's PWA checklist, focusing on service worker implementation and offline capabilities. Accessibility is scored based on the app's compatibility with screen readers and other accessibility features, while the Best Practices score checks for secure and modern coding practices, such as HTTPS and avoiding deprecated elements.

Once the Lighthouse report is generated, review the findings to identify areas that need improvement, such as optimizing load times, enhancing offline functionality, or addressing accessibility issues. The report provides actionable insights, allowing developers to make informed changes to improve the app’s overall performance and user experience. By regularly auditing your PWA using Lighthouse, you can ensure it meets high standards in terms of speed, usability, and accessibility.

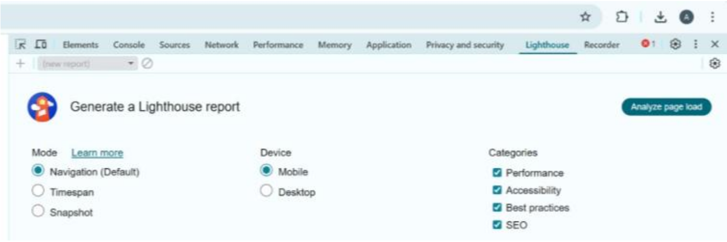
Implementation:

To implement Google Lighthouse for testing a Progressive Web App (PWA), first ensure your web app has a manifest.json file and a service worker for offline functionality. Once the PWA is set up, run the Lighthouse audit using Chrome

Developer Tools or the Lighthouse CLI. In Chrome, open the app, press F12, go to the Lighthouse tab, and select audit categories (Performance, PWA, Accessibility, Best Practices). Alternatively, use the CLI command lighthouse <URL> --pwa --output html --output-path ./lighthouse-report.html. The audit generates a report with scores for performance, PWA features, accessibility, and best practices. Review the results, make improvements, and re-audit to maintain a high-quality, compliant web app.

Output:

The Lighthouse audit report for your PWA indicates a performance score of 86%, which is a solid result, though there is room for improvement. A score of 86% suggests that your web app performs well in terms of load speed and interactivity, but there might be areas that could further enhance the user experience. To improve the performance score, consider optimizing images, reducing JavaScript execution time, improving server response times, and leveraging caching strategies. Regularly auditing the app and making performance improvements can help push the score closer to 100%, ensuring a faster and more responsive experience for users.



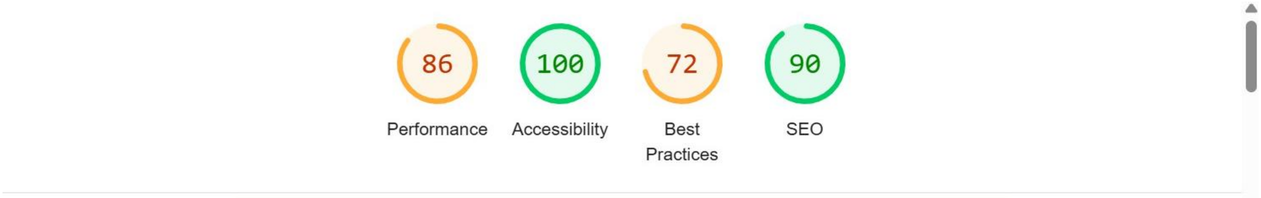
Generate a Lighthouse report

Mode
● Navigation (Default)
○ Timespan
○ Snapshot

Device
● Mobile
○ Desktop

Categories
☑ Performance
☑ Accessibility
☑ Best practices
☑ SEO

Conclusion:



The Lighthouse audit provides valuable insights into the performance and overall quality of your Progressive Web App (PWA). With a performance score of 86%, your app is performing well, but there are still opportunities to optimize load times and responsiveness. The audit's detailed recommendations, such as optimizing assets, improving caching strategies, and enhancing offline functionality, will guide you in making improvements. Regular use of Lighthouse audits ensures your app remains efficient, user-friendly, and aligned with best practices, ultimately providing a better experience for users and increasing the chances of your app being adopted as a reliable PWA.