Name: **Abhinav Swaminathan**	Div-Roll no: **D15C-01**
DOP:	DOS:
Sign:	Grade:

Experiment 11

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

Theory:

Google Lighthouse is an open-source, automated tool developed by Google to improve the quality of web pages. It provides audits for performance, accessibility, SEO, best practices, and Progressive Web App (PWA) standards.

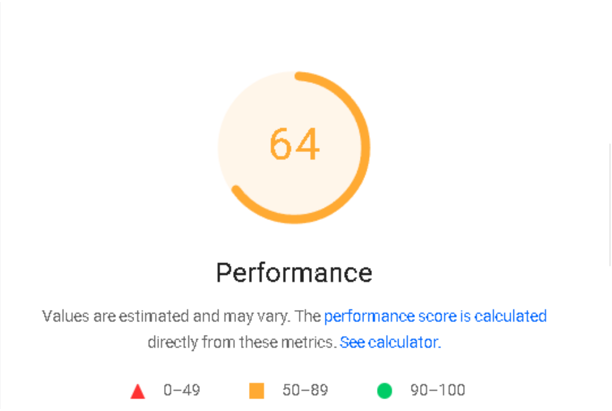
When testing a PWA, Lighthouse checks for critical requirements like:

- Valid web app manifest
- Presence and correct behavior of service workers
- HTTPS usage
- Responsive design
- Offline functionality
- “Add to Home Screen” capability

Lighthouse gives a score out of 100 based on how well the app performs as a PWA and offers suggestions for improvement. It is accessible directly in Chrome DevTools or as a browser extension.

This tool is crucial for ensuring the app meets modern web standards and provides a high-quality user experience on mobile and desktop.

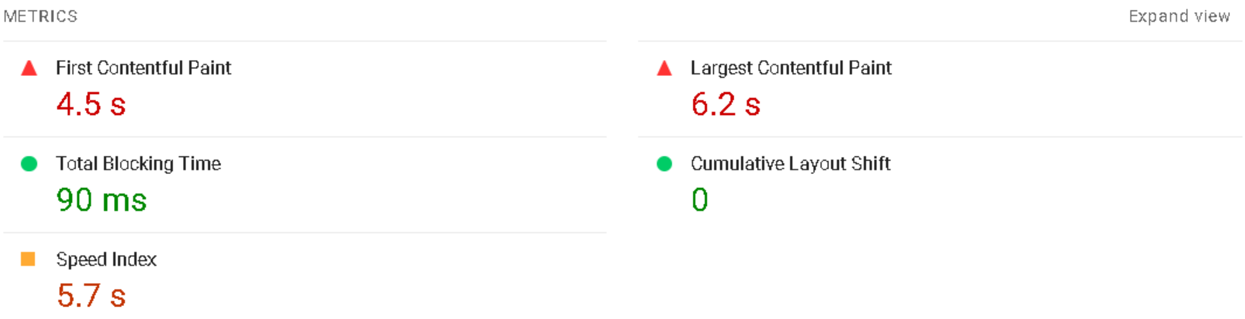
Output:



Performance

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

▲ 0–49 ■ 50–89 ● 90–100



METRICS Expand view

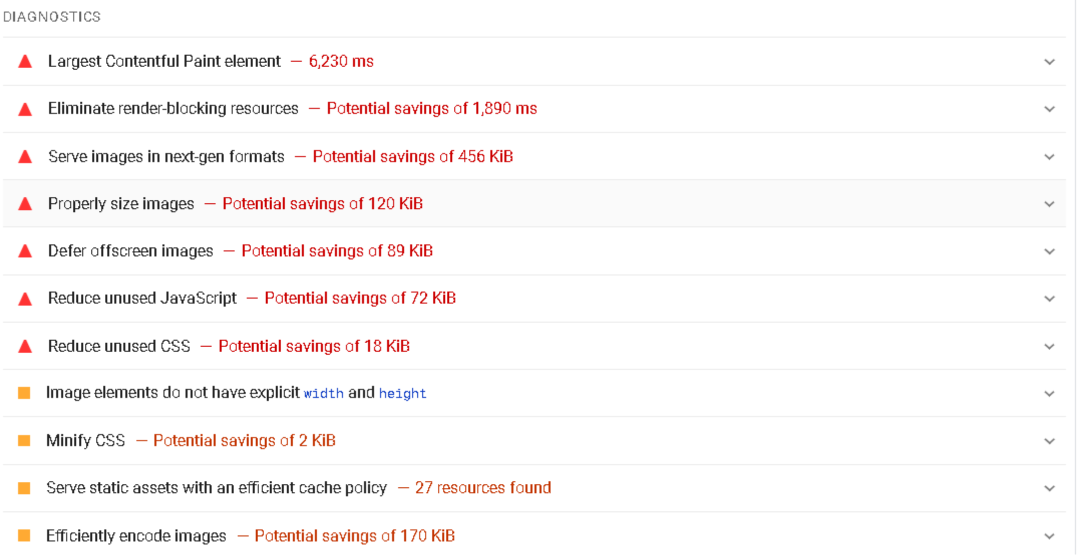
▲ First Contentful Paint
4.5 s

▲ Largest Contentful Paint
6.2 s

● Total Blocking Time
90 ms

● Cumulative Layout Shift
0

■ Speed Index
5.7 s



DIAGNOSTICS

▲ Largest Contentful Paint element — 6,230 ms

▲ Eliminate render-blocking resources — Potential savings of 1,890 ms

▲ Serve images in next-gen formats — Potential savings of 456 KiB

▲ Properly size images — Potential savings of 120 KiB

▲ Defer offscreen images — Potential savings of 89 KiB

▲ Reduce unused JavaScript — Potential savings of 72 KiB

▲ Reduce unused CSS — Potential savings of 18 KiB

■ Image elements do not have explicit `width` and `height`

■ Minify CSS — Potential savings of 2 KiB

■ Serve static assets with an efficient cache policy — 27 resources found

■ Efficiently encode images — Potential savings of 170 KiB

Conclusion:

Using Google Lighthouse, we successfully analyzed the PWA capabilities of railway app. The tool helped us verify important aspects like offline access, manifest configuration, and service worker functionality. It also provided valuable suggestions to optimize user experience and app performance, making it a vital step in PWA development and deployment.