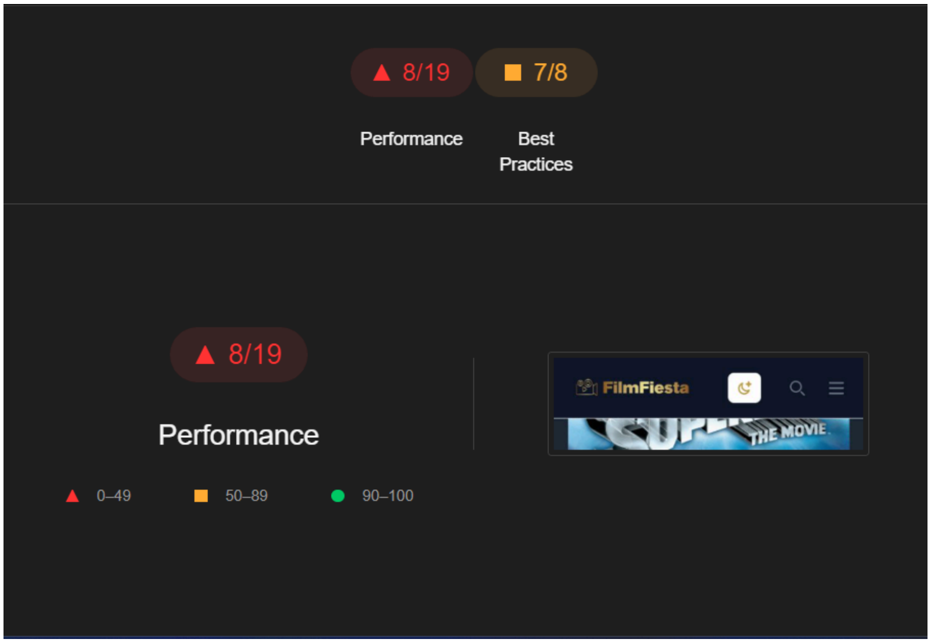
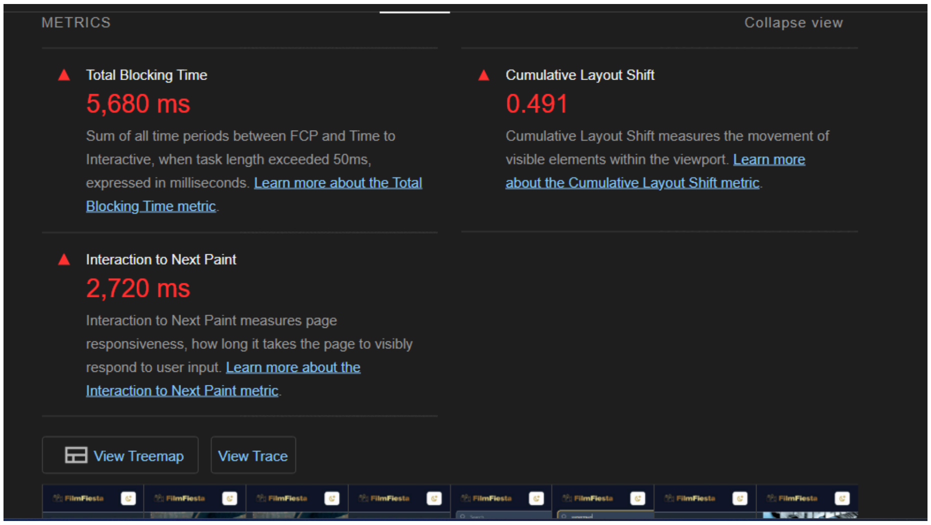
- Aim:To use the Google Lighthouse PWA Analysis Tool to test the Progressive Web App (PWA) functionality.

- Theory:Progressive Web Apps (PWAs) are web applications designed to provide a seamless, app-like experience on the web. Key characteristics of PWAs include:
    - Responsive Design: PWAs automatically adjust to different screen sizes and devices, providing a consistent user experience across mobile, tablet, and desktop.
    - Offline Capabilities: PWAs can function without an internet connection, thanks to service workers that cache assets and enable offline functionality.
    - App-like Experience: PWAs provide an app-like feel, including features such as home screen installation, push notifications, and improved loading performance.
    - Web App Manifest: A PWA includes a manifest file that defines how the app should appear when installed on the device, such as app name, icons, and theme color.
- Google Lighthouse is an automated tool by Google that audits web pages for performance, accessibility, SEO, and best practices. Specifically, for PWAs, Lighthouse checks:
    - Service Worker: Ensures that your app has a properly configured service worker for offline functionality.
    - Web App Manifest: Validates that your app includes a manifest file with the necessary metadata for installation.
    - Offline Functionality: Verifies that the app works offline by caching assets and supporting service worker operations.
    - Performance Metrics: Assesses loading time, interactivity, and other critical aspects for a smooth user experience.
- Objective
    - To evaluate the performance, PWA compliance, and best practices of the web app.\
    - To identify areas for improvement in the offline functionality, manifest file configuration, and service worker setup.
- Steps to Use Google Lighthouse for PWA Analysis:
    - Open Google Chrome:
        - Make sure you're using Google Chrome, as Lighthouse is integrated into Chrome's DevTools.
    - Open Your PWA in Google Chrome:
    - Navigate to your PWA URL in Google Chrome.
    - Open Chrome DevTools:
        - Right-click anywhere on the page and select Inspect (or use the shortcut Ctrl+Shift+I on Windows/Linux or Cmd+Opt+I on macOS).
        - Click on the Lighthouse tab in the DevTools panel. If it’s not visible, click the » icon to find it.

- Configure Lighthouse Settings: ○ Categories: Choose the categories you want to audit, including Performance, PWA, Accessibility, Best Practices, and SEO.
- Device Mode: Select between Mobile or Desktop to simulate the audit on different devices.
- Make sure to select the PWA category for a specific audit on Progressive Web App functionality.
- Run the Lighthouse Audit: Click the "Generate report" button to begin the audit. Lighthouse will analyze your PWA and generate a report.

- Review the Results:
    - Once the audit is completed, you'll see a detailed report with scores in different categories. In the PWA section, Lighthouse will evaluate:
        - Once the audit is completed, you'll see a detailed report with scores in different categories. In the PWA section, Lighthouse will evaluate:
            - Service Worker: Checks if a service worker is installed and functioning correctly.
            - Web App Manifest: Verifies the presence and correctness of your manifest file.
            - Offline Capability: Assesses if your PWA can function offline.
            - App Installability: Tests whether the app can be installed on the home screen.
- Output :





METRICS

▲ Total Blocking Time

5,680 ms

Sum of all time periods between FCP and Time to Interactive, when task length exceeded 50ms, expressed in milliseconds. Learn more about the Total Blocking Time metric.

▲ Cumulative Layout Shift

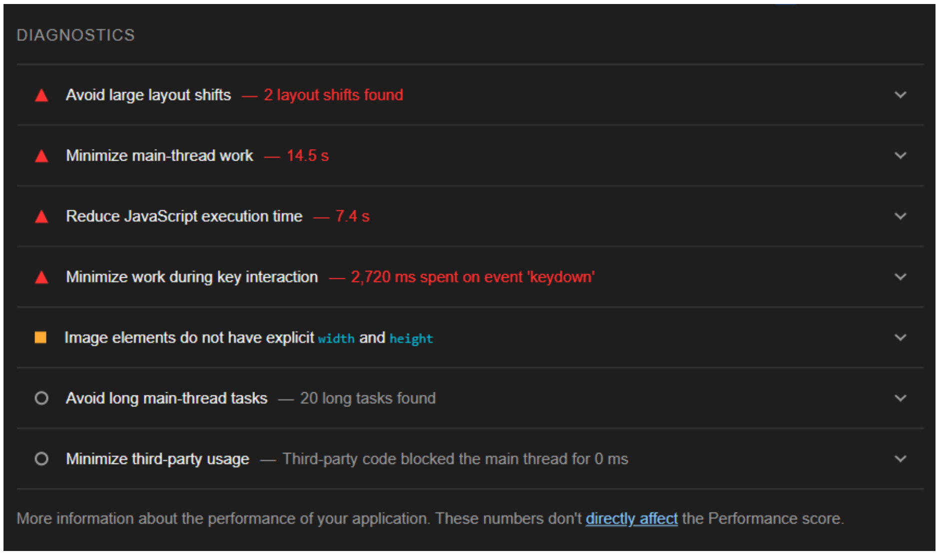
0.491

Cumulative Layout Shift measures the movement of visible elements within the viewport. Learn more about the Cumulative Layout Shift metric.

▲ Interaction to Next Paint

2,720 ms

Interaction to Next Paint measures page responsiveness, how long it takes the page to visibly respond to user input. Learn more about the Interaction to Next Paint metric.



DIAGNOSTICS

▲ Avoid large layout shifts — 2 layout shifts found

▲ Minimize main-thread work — 14.5 s

▲ Reduce JavaScript execution time — 7.4 s

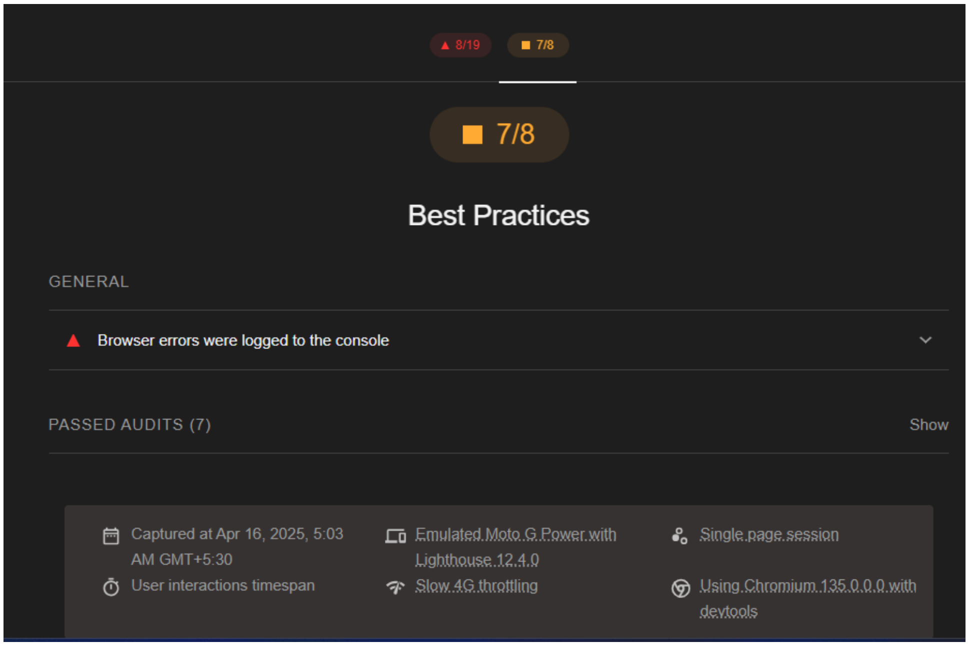
▲ Minimize work during key interaction — 2,720 ms spent on event 'keydown'

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

○ Avoid long main-thread tasks — 20 long tasks found

○ Minimize third-party usage — Third-party code blocked the main thread for 0 ms

More information about the performance of your application. These numbers don't directly affect the Performance score.



▲ 8/19 ■ 7/8

■ 7/8

Best Practices

GENERAL

▲ Browser errors were logged to the console

PASSED AUDITS (7) Show

Captured at Apr 16, 2025, 5:03 AM GMT+5:30

User interactions timespan

Emulated Moto G Power with Lighthouse 12.4.0

Slow 4G throttling

Single page session

Using Chromium 135.0.0.0 with devtools

- Conclusion : Using the Google Lighthouse PWA Analysis Tool is a crucial step in ensuring that your Progressive Web App meets key functionality standards. By following the steps above, you can identify and resolve any issues related to service workers, manifest files, and offline capabilities. This process helps optimize performance, enhances the user experience, and ensures compliance with PWA best practices.