Experiment No.11

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

- **Theory:**

Google Lighthouse:

Google Lighthouse is an automated tool developed by Google that assesses the performance and quality of web pages. It is integrated into the Google Chrome browser and provides insights into various aspects of web development, including performance, accessibility, best practices, and search engine optimization (SEO). Lighthouse performs audits on web pages by simulating a user's experience and analyzing the page's behavior and structure.

Progressive Web App (PWA) Analysis Tool:

The Progressive Web App (PWA) Analysis Tool within Google Lighthouse specifically evaluates the characteristics of a web page that make it a Progressive Web App. PWAs are web applications that provide a native app-like experience to users, including features like offline functionality, push notifications, and responsive design.

Lighthouse evaluates several key factors:

1. Service Worker: Lighthouse checks if the web page has a service worker registered and properly configured. Service workers are essential for enabling features like offline caching and background sync in PWAs.

2. Offline Capabilities: Lighthouse assesses whether the PWA can function offline by caching essential resources and providing a meaningful offline experience to users.

3. App-like Behavior: Lighthouse examines if the PWA behaves like a native app, including features such as a splash screen, full-screen mode, and home screen installation prompts.

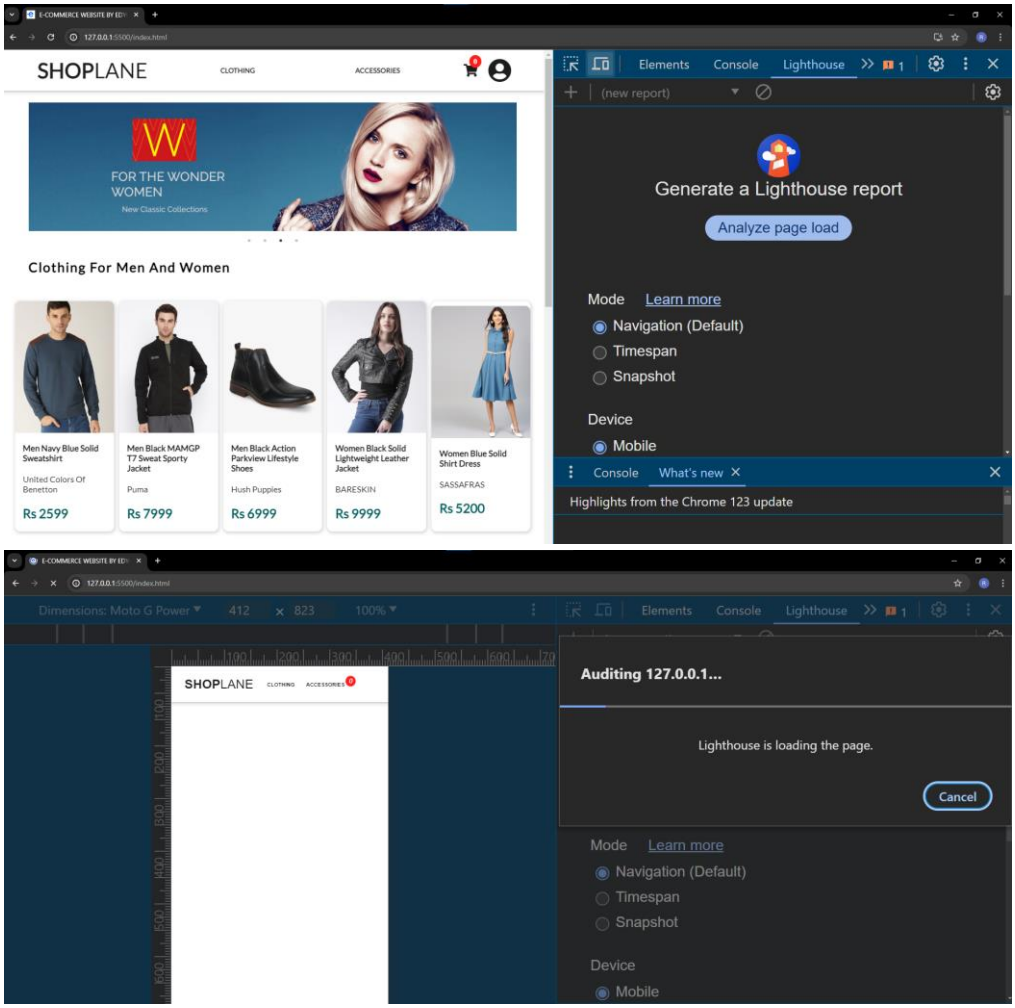
4. Responsive Design: Lighthouse verifies that the PWA is responsive and works well across various devices and screen sizes, ensuring a consistent user experience.

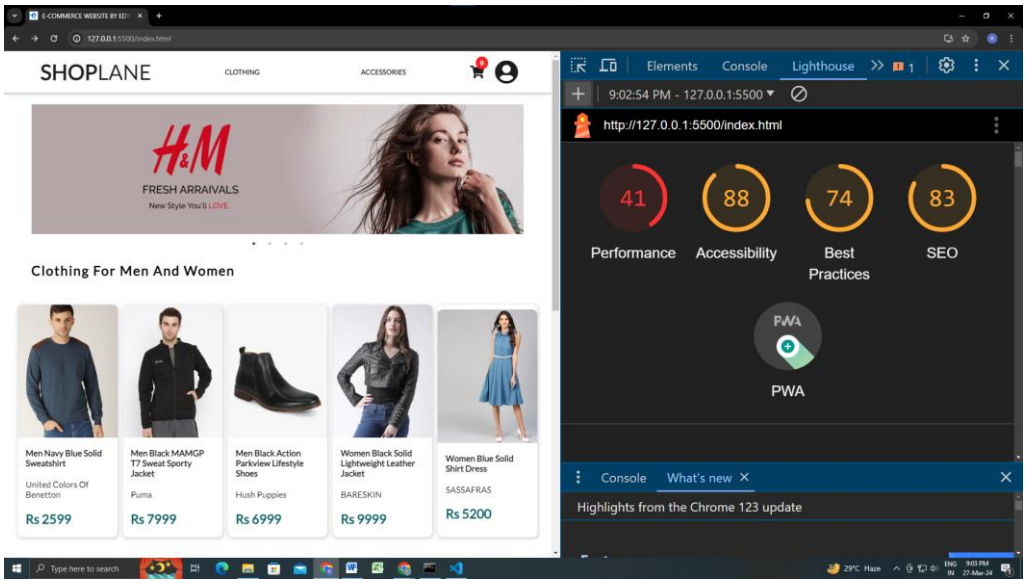
5. Performance: While not specific to PWAs, Lighthouse evaluates the performance of the web page, including metrics such as load time, speed index, and time to interactive. A fast and optimized PWA enhances user engagement and satisfaction.

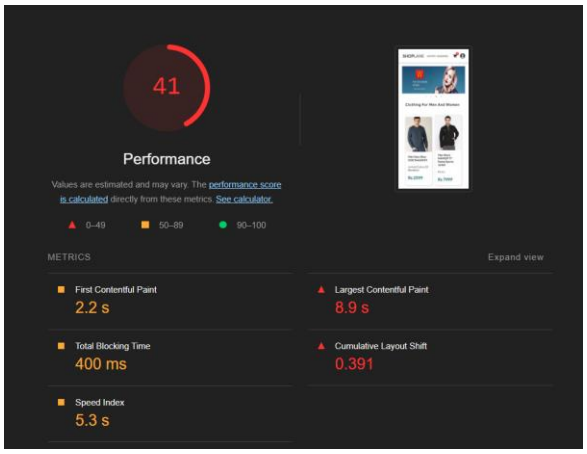
Process to use Google Lighthouse PWA Analysis Tool to test the PWA functioning:

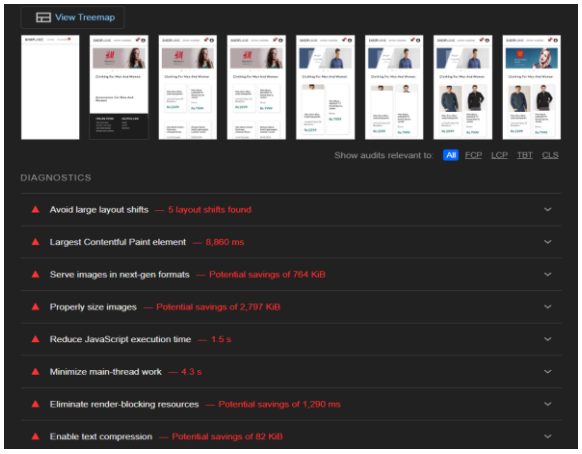
1. Open Google Chrome browser and navigate to the webpage of your PWA.
2. Right-click on the page and select "Inspect" to open the Chrome DevTools.
3. Go to the "Audits" tab.
4. Check the "Progressive Web App" option.
5. Click on the "Run audits" button.
6. Wait for the analysis to complete.
7. Review the results, which will include scores for various aspects of your PWA such as performance, accessibility, best practices, and SEO.
8. Use the feedback provided to optimize and enhance the functionality of your PWA.

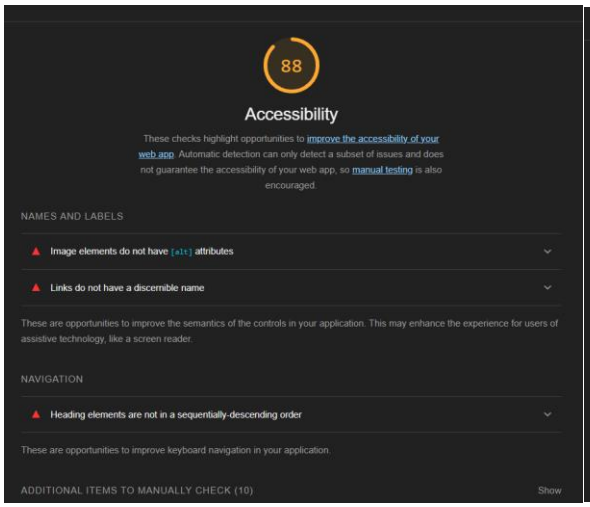
- **Output:-**

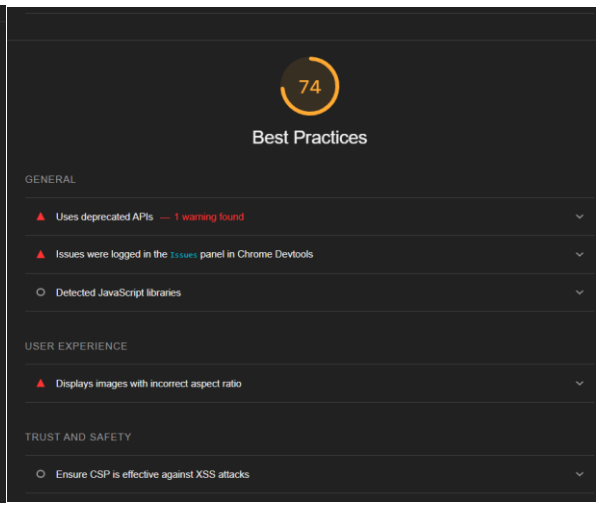


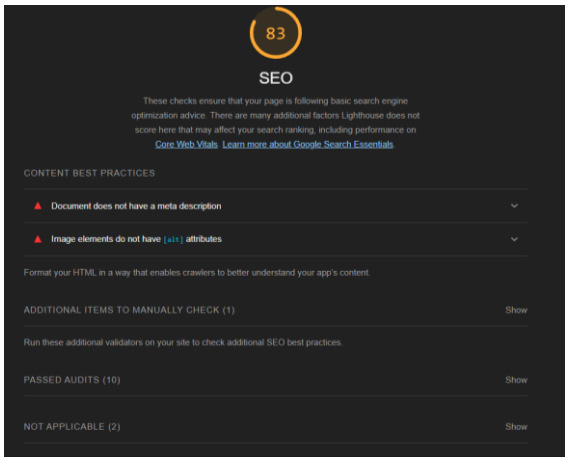

SHOPLANE
CLOTHING
ACCESSORIES
H&M
FRESH ARRAIVALS
New Style You'll LOVE
Clothing For Men And Women
Men Navy Blue Solid Sweatshirt
United Colors Of Benetton
Rs 2599
Men Black MAMGP T7 Sweat Sporty Jacket
Puma
Rs 7999
Men Black Action Parkview Lifestyle Shoes
Hush Puppies
Rs 6999
Women Black Solid Lightweight Leather Jacket
BARESKIN
Rs 9999
Women Blue Solid Shirt Dress
SASSAFRAS
Rs 5200
Elements
Console
Lighthouse
http://127.0.0.1:5500/index.html
41
Performance
88
Accessibility
74
Best Practices
83
SEO
PWA
Console
What's new
Highlights from the Chrome 123 update

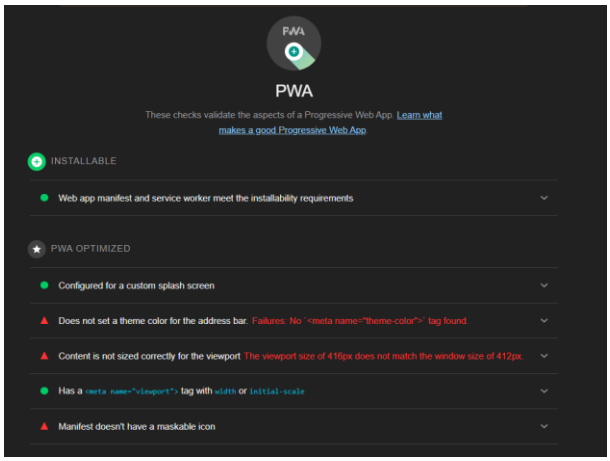

41
Performance
Values are estimated and may vary. The performance score is calculated directly from these metrics. See calculator.
0–49
50–89
90–100
METRICS
Expand view
First Contentful Paint
2.2 s
Largest Contentful Paint
8.9 s
Total Blocking Time
400 ms
Cumulative Layout Shift
0.391
Speed Index
5.3 s


View Treemap
Show audits relevant to: All FCP LCP TBT CLS
DIAGNOSTICS
Avoid large layout shifts — 5 layout shifts found
Largest Contentful Paint element — 8,860 ms
Serve images in next-gen formats — Potential savings of 764 KiB
Properly size images — Potential savings of 2,797 KiB
Reduce JavaScript execution time — 1.5 s
Minimize main-thread work — 4.3 s
Eliminate render-blocking resources — Potential savings of 1,290 ms
Enable text compression — Potential savings of 82 KiB


88
Accessibility
These checks highlight opportunities to improve the accessibility of your web app. Automatic detection can only detect a subset of issues and does not guarantee the accessibility of your web app, so manual testing is also encouraged.
NAMES AND LABELS
Image elements do not have [alt] attributes
Links do not have a discernible name
These are opportunities to improve the semantics of the controls in your application. This may enhance the experience for users of assistive technology, like a screen reader.
NAVIGATION
Heading elements are not in a sequentially-descending order
These are opportunities to improve keyboard navigation in your application.
ADDITIONAL ITEMS TO MANUALLY CHECK (10)
Show


74
Best Practices
GENERAL
Uses deprecated APIs — 1 warning found
Issues were logged in the Issues panel in Chrome Devtools
Detected JavaScript libraries
USER EXPERIENCE
Displays images with incorrect aspect ratio
TRUST AND SAFETY
Ensure CSP is effective against XSS attacks





- **Conclusion:**

In conclusion, utilizing Google Lighthouse PWA Analysis Tool to test the PWA functionality has provided valuable insights into its performance, accessibility, and best practices. This comprehensive analysis helps identify areas for improvement, ensuring that the PWA delivers an optimal user experience across various devices and network conditions.