EXPERIMENT NO: - 11

Manorath Ital
D15A/19

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

THEORY: -

Google Lighthouse is an open-source tool that audits web applications based on multiple key parameters, including performance, accessibility, Progressive Web App (PWA) implementation, and best practices. It provides a detailed, automated report that helps developers optimize their websites efficiently. Unlike traditional manual audits, which can take days or even weeks, Lighthouse generates insights within minutes.

One of the key advantages of Lighthouse is its ease of use—no complex setup is required. Simply run it on a webpage or provide a URL, and it will generate an extensive performance report.

Key Features and Audit Metrics

Lighthouse can audit both desktop and mobile versions of a webpage. The core evaluation criteria include:

1. Performance

This metric measures how efficiently a webpage loads and displays content. Key factors influencing the performance score include:

- Page load speed – How quickly the page becomes visible to the user.
- First Contentful Paint (FCP) – The time taken for the first piece of content to appear.
- Largest Contentful Paint (LCP) – The time taken for the main content to fully load.
- Cumulative Layout Shift (CLS) – Measures how visually stable a page is (i.e., avoiding unexpected shifts in content).

- Time to Interactive (TTI) – The time it takes for the page to become fully functional. Lighthouse assigns a score from 0 to 100 based on percentile rankings, where:
- 100 → Top 2% of websites (98th percentile)
- 50 → Around the 75th percentile
- Lower scores → Indicate areas that need optimization

2. Progressive Web App (PWA) Score (Mobile)

With the rise of PWAs, modern web applications aim to provide a native app-like experience. Lighthouse evaluates the PWA implementation based on Google's Baseline PWA Checklist, which includes:

- Service Worker implementation – Ensuring offline support and background synchronization.
- App Manifest compliance – Providing metadata for better mobile integration.
- Viewport configuration – Optimizing mobile responsiveness.
- Performance in script-disabled environments – Ensuring the page functions even when JavaScript is disabled.

A high PWA score indicates that the application meets essential PWA criteria and provides an app-like user experience.

3. Accessibility

Accessibility ensures that web applications are usable by individuals with disabilities. Lighthouse audits a webpage based on:

- ARIA attributes – Enhancing accessibility through attributes like aria-required.
- Text alternatives for media – Ensuring audio and visual content is accessible.
- Semantic HTML – Proper use of <section>, <article>, <button>, and other elements that improve screen-reader compatibility.

Unlike other metrics, accessibility checks follow a pass/fail approach—if a necessary feature is missing, it significantly impacts the score. A higher accessibility score ensures inclusivity for users with visual or cognitive impairments. 4. Best Practices

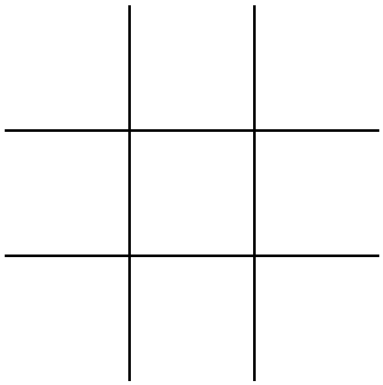
This metric evaluates whether the website follows modern web development best practices, including:

- Use of HTTPS – Ensuring secure data transmission.
- Avoiding deprecated code – Preventing the use of outdated elements, directives, and libraries.
- Secure password inputs – Disabling paste-into fields to mitigate credential theft risks.
- User security alerts – Prompting users about geo-location access and cookie usage on load. A high score indicates that the website follows industry standards, improving security, usability, and maintainability.

Manifest.json

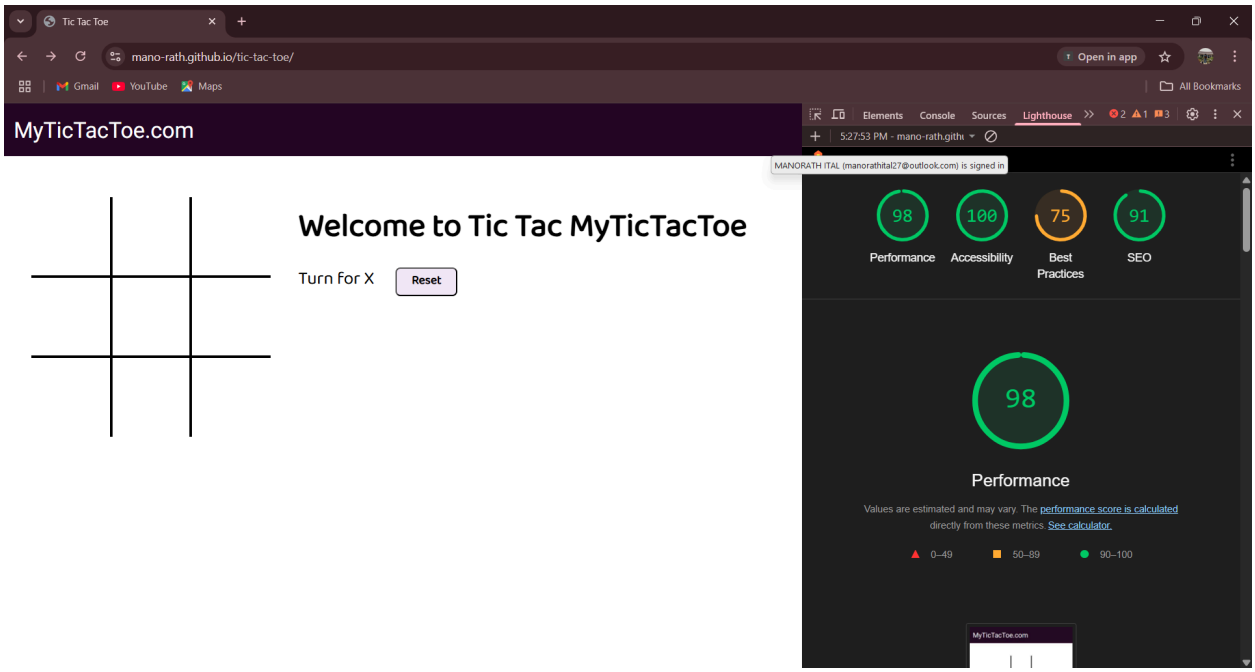
```
{
 "name": "Tic Tac Toe",
 "short_name": "TicTacToe",
 "description": "A simple Tic-Tac-Toe game.",
 "start_url": ".",
 "display": "standalone",
 "background_color": "#ffffff",
 "theme_color": "#4CAF50",
 "icons": [
   {
     "src": "icons/tic-tac-toe-icon.png",
     "sizes": "192x192",
     "type": "image/png"
   },
   {
     "src": "icons/tic-tac-toe-icon.png",
     "sizes": "512x512",
     "type": "image/png"
   }
 ]
}
```

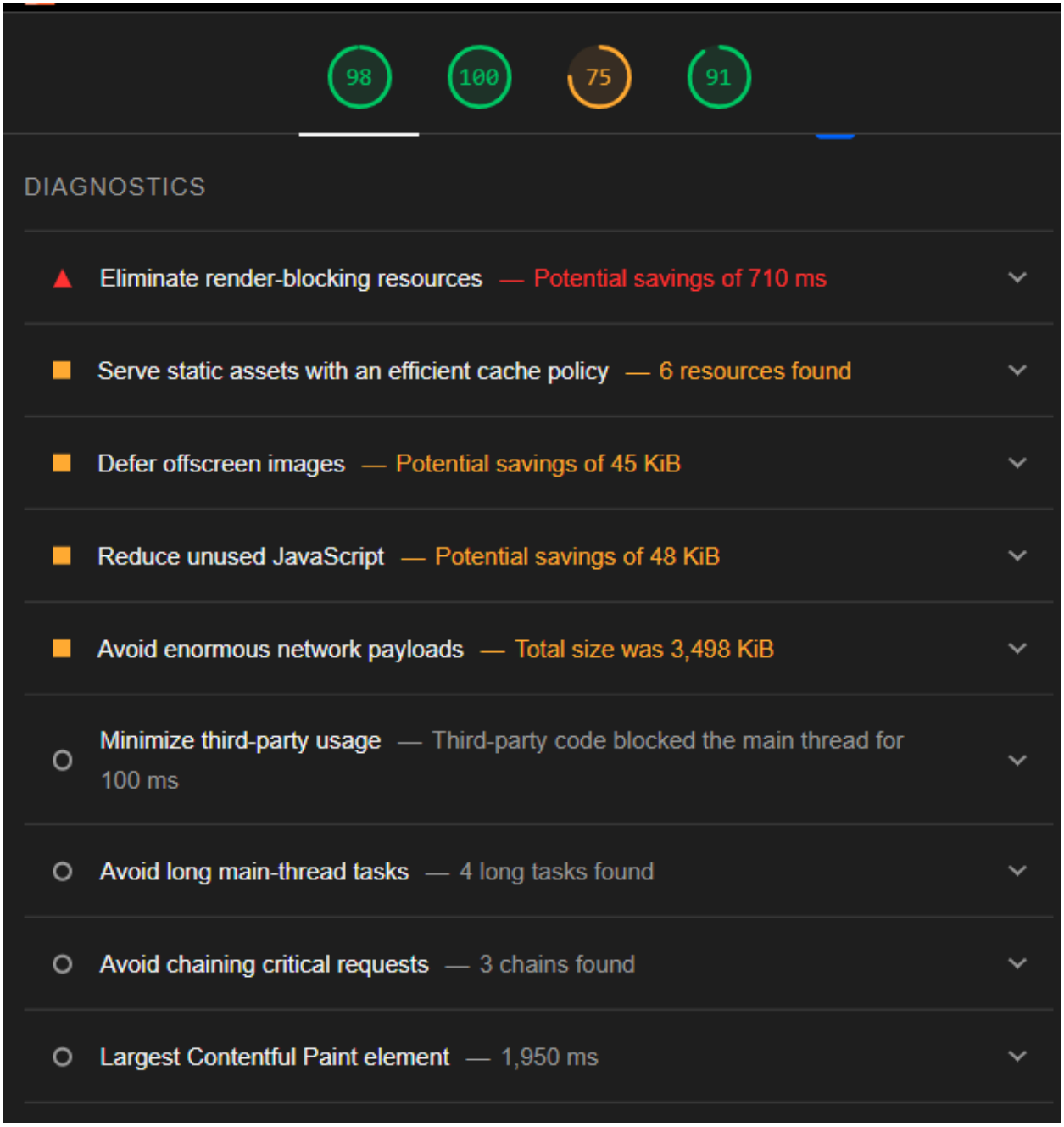
MyTicTacToe.com

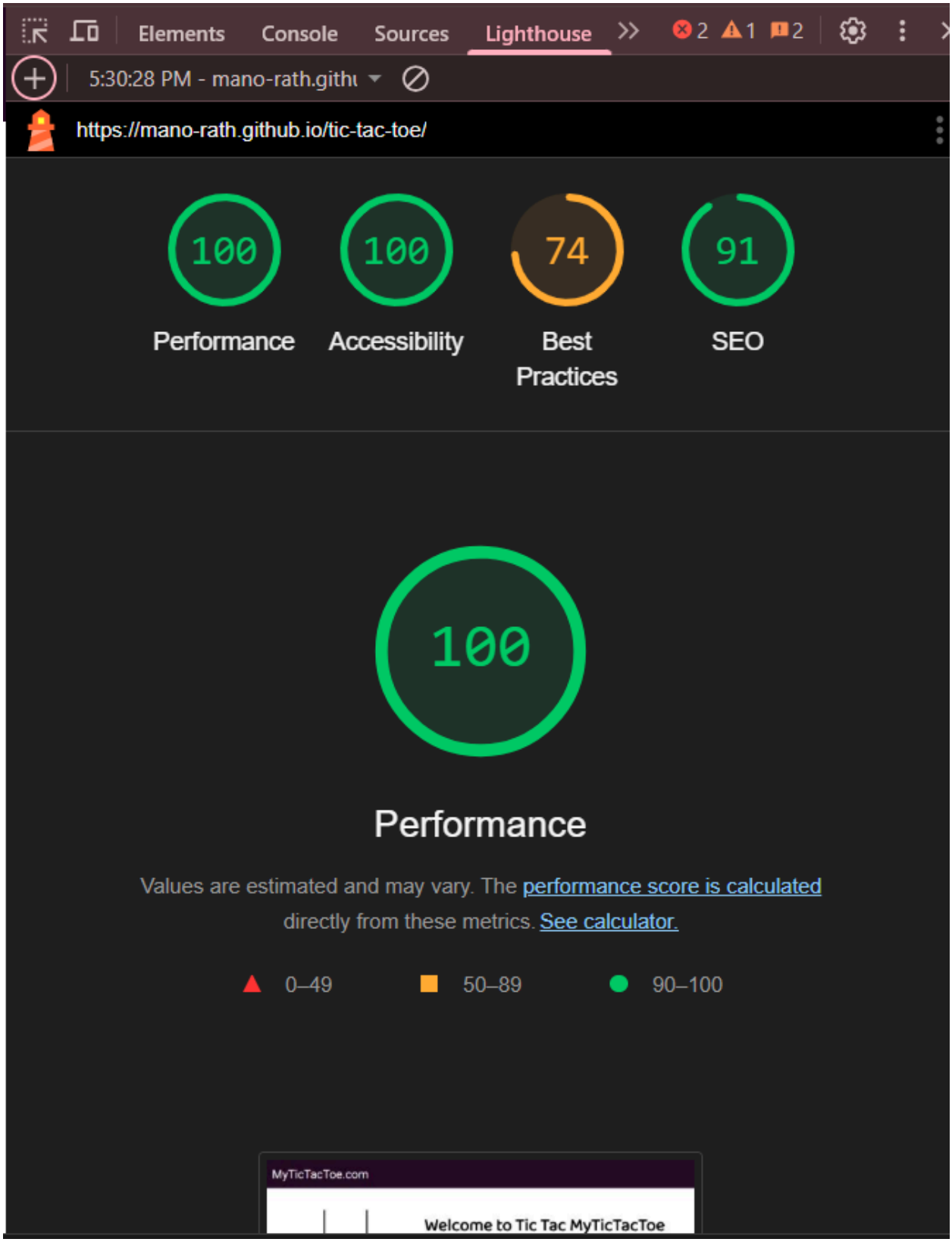


Welcome to Tic Tac MyTicTacToe

Turn for X Reset




98
100
75
91
DIAGNOSTICS
Eliminate render-blocking resources — Potential savings of 710 ms
Serve static assets with an efficient cache policy — 6 resources found
Defer offscreen images — Potential savings of 45 KiB
Reduce unused JavaScript — Potential savings of 48 KiB
Avoid enormous network payloads — Total size was 3,498 KiB
Minimize third-party usage — Third-party code blocked the main thread for 100 ms
Avoid long main-thread tasks — 4 long tasks found
Avoid chaining critical requests — 3 chains found
Largest Contentful Paint element — 1,950 ms


Elements
Console
Sources
Lighthouse
2
1
2
5:30:28 PM - mano-rath.githu
https://mano-rath.github.io/tic-tac-toe/
100
Performance
100
Accessibility
74
Best Practices
91
SEO
100
Performance
Values are estimated and may vary. The performance score is calculated directly from these metrics. See calculator.
0–49
50–89
90–100
MyTicTacToe.com
Welcome to Tic Tac MyTicTacToe