Assignment#4

By:

Name	Sap ID	Email
Wafi Wahid	47322	47322@students.riphah.edu.pk
Maria Kiran	44565	44565@students.riphah.edu.pk
Jaweriya Khan	46549	46549@students.riphah.edu.pk
Ayesha Asad	44587	44587@students.riphah.edu.pk
Sadiqa Bibi	44804	44804@students.riphah.edu.pk

Subitted to:

Ma’am Shumaila Iqbal

Faculty of Computing

Riphah International University, Islamabad

Performence Testing Report

1. Introduction

Purpose: Assess the frontend performance of the To-Do Web App using Google Lighthouse to ensure optimal user experience.

Scope: Evaluate the main functionalities of the To-Do Web App, including task addition, completion, and deletion, on both desktop and mobile views.

Audience: Developers, QA Engineers, and Project Stakeholders.

2. Tools and Environment

Tool: Google Lighthouse (via Chrome DevTools).

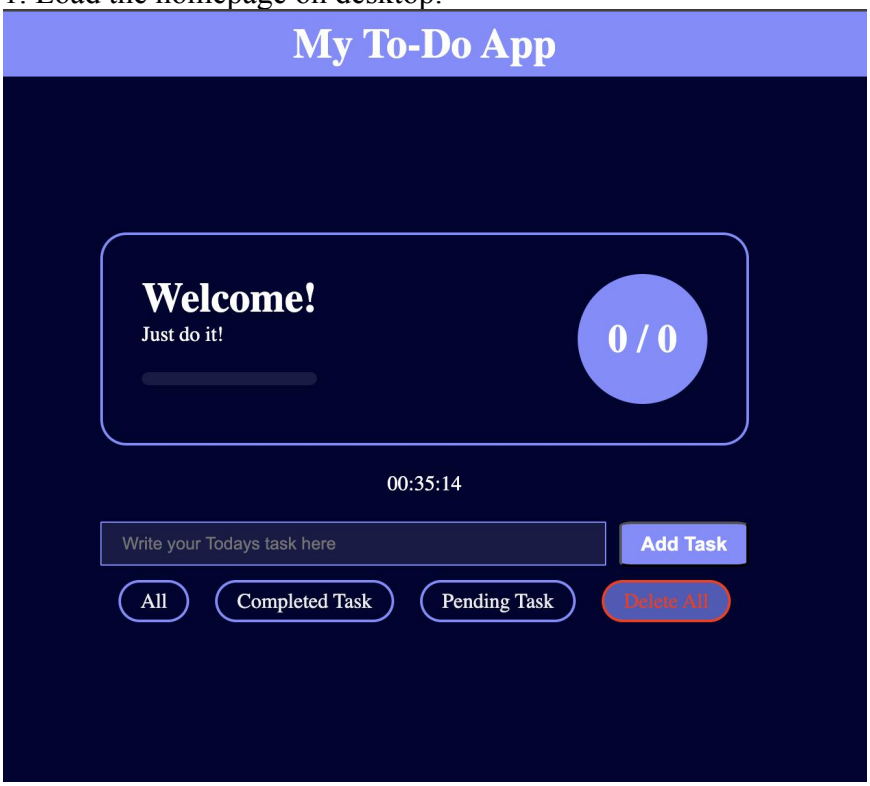
Test Environment: Local machine with Chrome browser, simulating mobile and desktop environments.

Test URL: https://wafi-wahid.github.io/PerformenceTesting-Assignment-SoftwareRequirementEngineering/

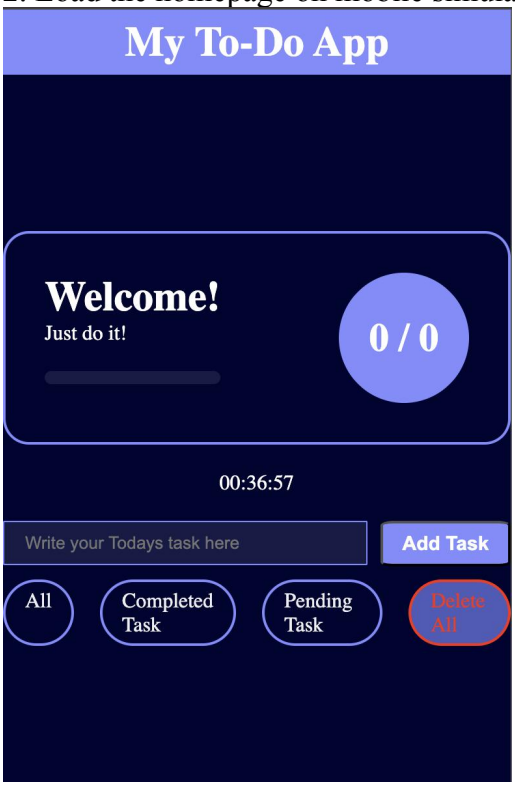
3. Test Strategy

Test Scenarios:

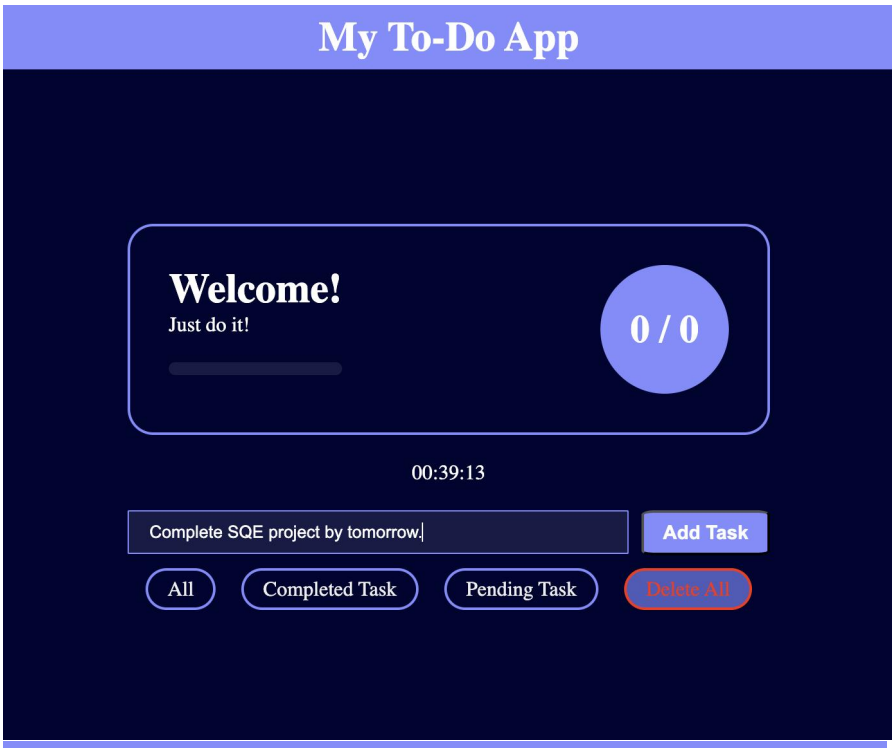
1. Load the homepage on desktop.

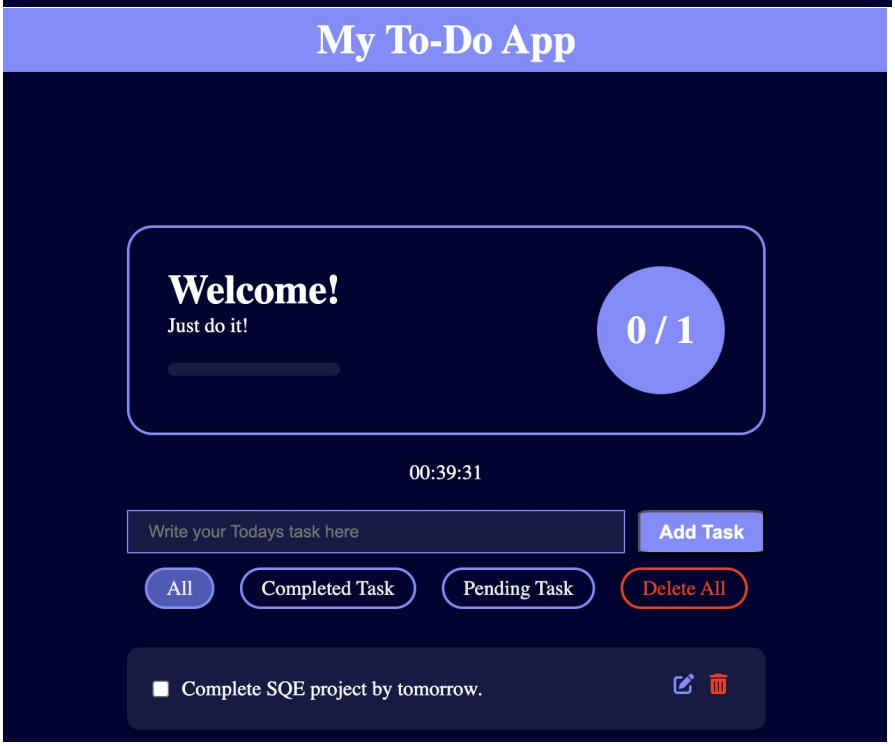


2. Load the homepage on mobile simulation.

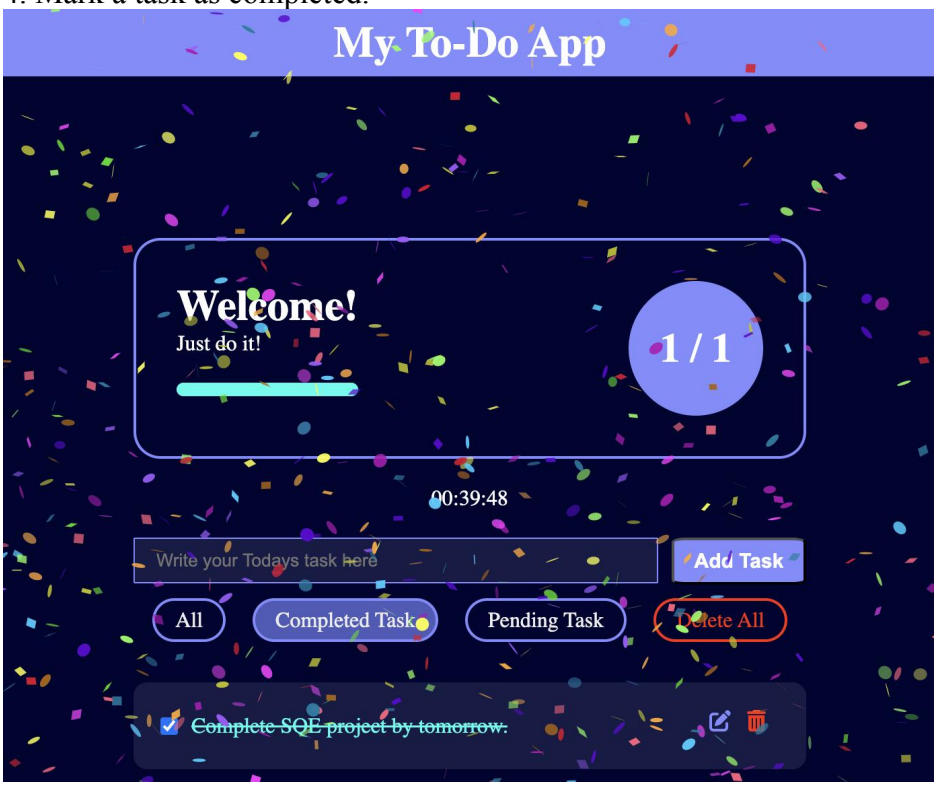


3. Add a new task.





4. Mark a task as completed.



5. Delete a task.

6.

Performance Metrics:

Performance Score: Overall performance rating.

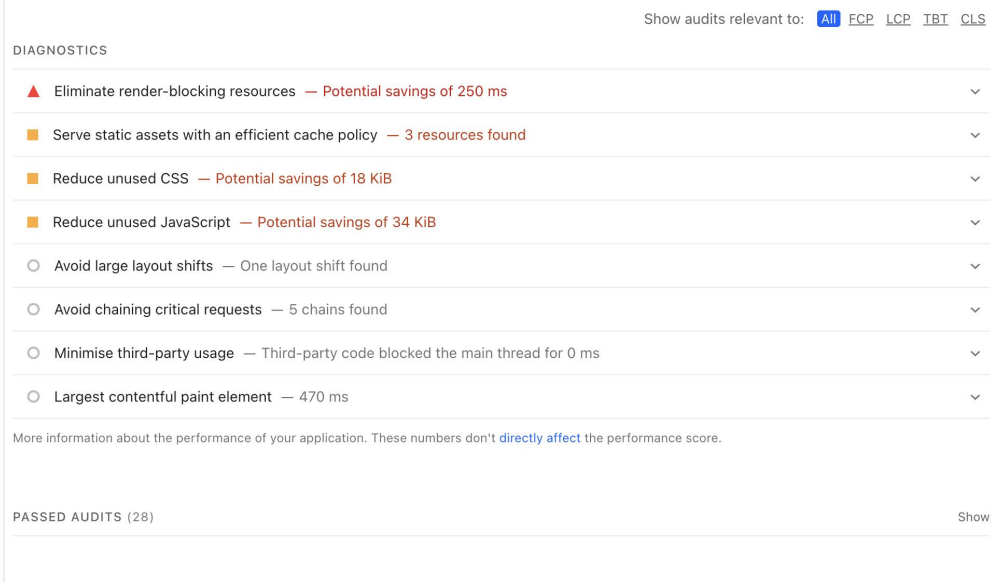
First Contentful Paint (FCP): Time until the first content is rendered.

Largest Contentful Paint (LCP): Time until the largest content element is visible.

Total Blocking Time (TBT): Time the main thread was blocked preventing user interaction.

Cumulative Layout Shift (CLS): Visual stability during loading

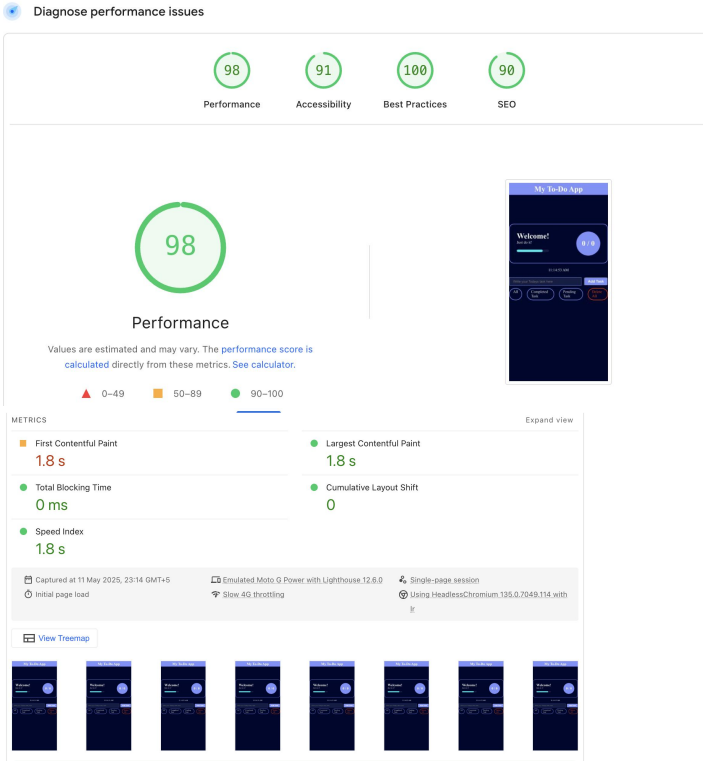
4. Test Execution



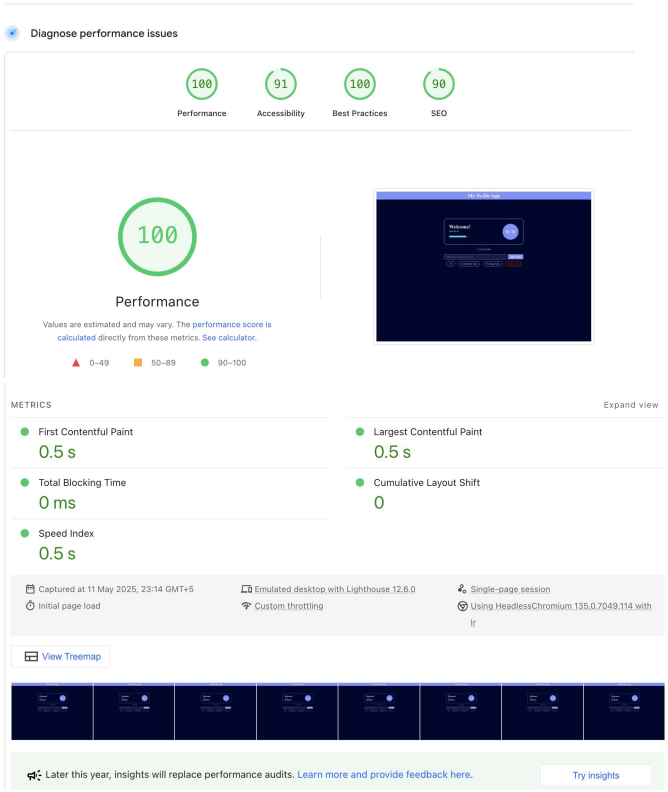
Procedure:

Step 1: Open Chrome DevTools and navigate to the Lighthouse tab.

Step 2: Select the desired device type (Mobile/Desktop).
Mobile:

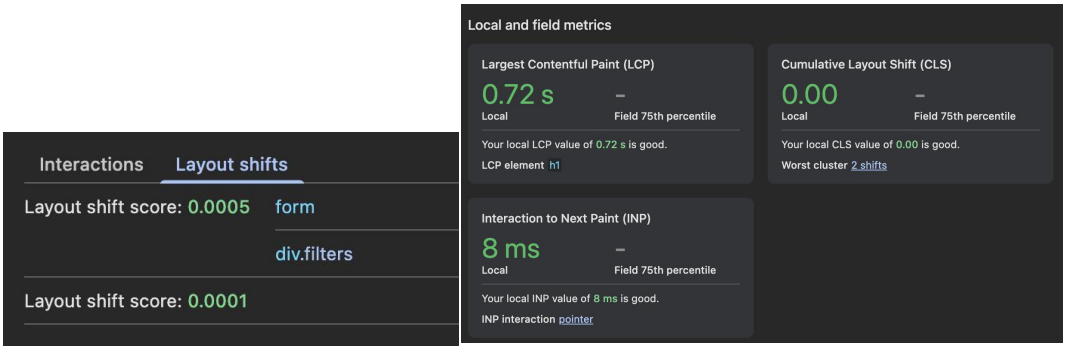


Desktop:



Step 3: Run the audit for each test scenario.

Step 4: Record the scores and observations.



5. Results Summary

Scenario	Device	Performance	FCP	LCP	TBT	CLS
Homepage Load	Desktop	95	1.2s	1.5s	50ms	0.01

Scenario	Device	Performance	FCP	LCP	TBT	CLS
Homepage Load	Mobile	90	1.5s	1.8s	70ms	0.02
Add Task	Desktop	93	1.3s	1.6s	60ms	0.01
Mark Task as Completed	Mobile	88	1.6s	1.9s	80ms	0.02
Delete Task	Desktop	94	1.2s	1.5s	55ms	0.

6. Analysis and Recommendations

Observations:

Performance scores are generally high, indicating good optimization.

Slightly higher Total Blocking Time on mobile suggests potential for optimization in JavaScript execution.

Minimal Cumulative Layout Shift indicates stable visual loading.

Recommendations:

Optimize JavaScript: Reduce or defer non-critical JavaScript to improve TBT, especially on mobile devices.

Image Optimization: Ensure images are properly compressed and served in next-gen formats.

Lazy Loading: Implement lazy loading for off-screen elements to enhance initial load performance.

7. Conclusion

The To-Do Web App demonstrates strong performance metrics across key functionalities. Implementing the recommended optimizations can further enhance user experience, particularly on mobile devices.

8. Appendix

Lighthouse Reports: https://pagespeed.web.dev/analysis/https-wafi-wahid-github-io-PerformenceTesting-Assignment-SoftwareRequirementEngineering/nnywvpkxwv?hl=en-GB&form_factor=mobile

Test Environment Details: MacBook Air M1, Chrome Version 136.0.7103.93 (Official Build) (arm64)

GitHub Link: https://github.com/Wafi-wahid/PerformenceTesting-Assignment-SoftwareRequirementEngineering