St. Francis Institute of Technology, Mumbai-400 103
Department of Information Technology

A.Y. 2024-2025
Class: TE-ITA/B, Semester: VI

Subject: **MAD & PWA LAB**

Experiment – 11: Testing the PWA functioning using Google Lighthouse PWA Analysis Tool

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

2. **Objectives:** After study of this experiment, the student will be able to
   - Test and Deploy PWA.

3. **Outcomes:** After study of this experiment, the student will be able to
   - Analyse PWA Features.

4. **Prerequisite:** HTML/ CSS/ JavaScript.

5. **Requirements:** Visual Studio Code, Google Chrome, Internet Connection.

6. **Pre-Experiment Exercise:**
   Brief Theory:

   Lighthouse:
   Google Lighthouse is an open-source, automated tool originally developed by Google, for measuring the quality of web pages. Lighthouse can be used to automatically audit your app against a set of PWA best practices. It has audits for performance, accessibility, progressive web apps, SEO and more.
   Lighthouse is available both as a browser extension (shown in Figure 1), and as a command-line tool that you can integrate into your continuous integration pipeline. You give Lighthouse a URL to audit, it runs a series of audits against the page, and then it generates a report on how well the page did. From there, use the failing audits as indicators on how to improve the page. Each audit has a reference doc explaining why the audit is important, as well as how to fix it.

   Testing PWA using Google Lighthouse:
   Various lighthouse workflows possible to test your PWA.
   1. **In Chrome DevTools**. Easily audit pages that require authentication, and read your reports in a user-friendly format.
   2. **From the command line**. Automate your Lighthouse runs via shell scripts.
   3. **As a Node module.** Integrate Lighthouse into your continuous integration systems.
   4. **From a web UI.** Run Lighthouse and link to reports without installing a thing.
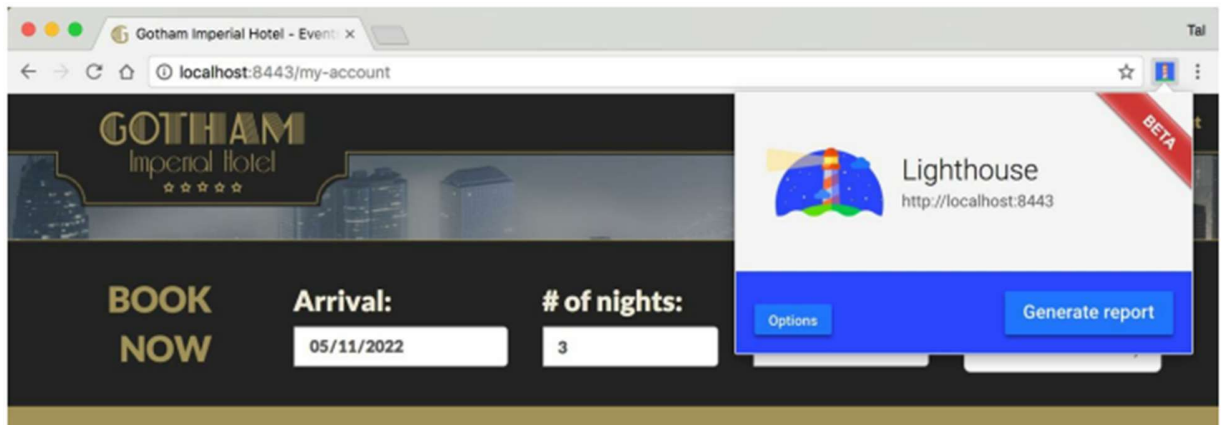
Figure 1: Lighthouse Chrome extension testing the PWA.

7. Laboratory Exercise
A. Program
1. Test and analyze your PWA using Google Lighthouse PWA analysis tool.

B. Result/Observation
1. Print out of program code and output.

8. Post-Experimental Exercise
A. Questions:
1. How to test PWAs?

B. Conclusion:
1. Write what you have learnt in the experiment.

C. References:
1. https://developers.google.com/web/tools/lighthouse
2. Building Progressive Web Apps, O'Reilly 2017.

Laboratory Exercise: