Experiment 11

Name: Shashwat Tripathi
Div: D15A
Roll no: 64

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

Theory :
Reference : https://www.semrush.com/blog/google-lighthouse/

Google Lighthouse :

Google Lighthouse is a tool that lets you audit your web application based on a number of parameters including (but not limited to) performance, based on a number of metrics, mobile compatibility, Progressive Web App (PWA) implementations, etc. All you have to do is run it on a page or pass it a URL, sit back for a couple of minutes and get a very elaborate report, not much short of one that a professional auditor would have compiled in about a week.

The best part is that you have to set up almost nothing to get started. Let's begin by looking at some of the top features and audit criteria used by Lighthouse.

Key Features and Audit Metrics

Google Lighthouse has the option of running the Audit for Desktop as well as mobile version of your page(s). The top metrics that will be measured in the Audit are:
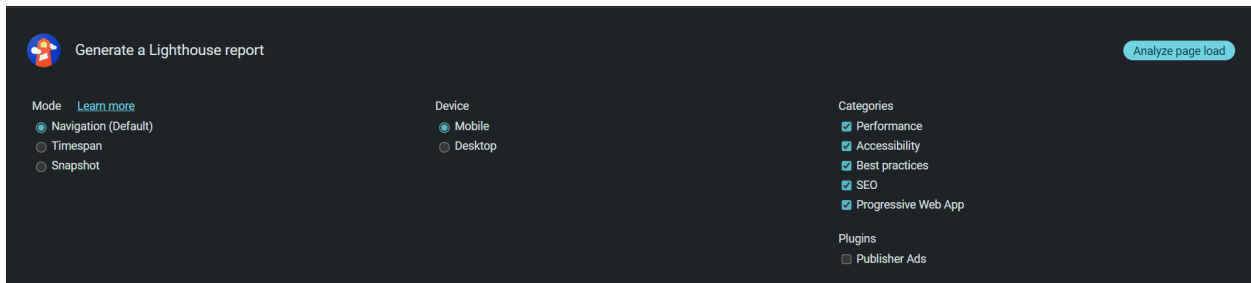
1.   **Performance:** This score is an aggregation of how the page fared in aspects such as (but not limited to) loading speed, time taken for loading for basic frame(s), displaying meaningful content to the user, etc. To a layman, this score is indicative of how decently the site performs, with a score of 100 meaning that you figure in the 98th percentile, 50 meaning that you figure in the 75th percentile and so on.

2.   **PWA Score (Mobile):** Thanks to the rise of Service Workers, app manifests, etc., a lot of modern web applications are moving towards the PWA paradigm, where the objective is to make the application behave as close as possible to native mobile applications. Scoring points are based on the Baseline PWA checklist laid down by Google which includes Service Worker implementation(s), viewport handling, offline functionality, performance in script-disabled environments, etc.
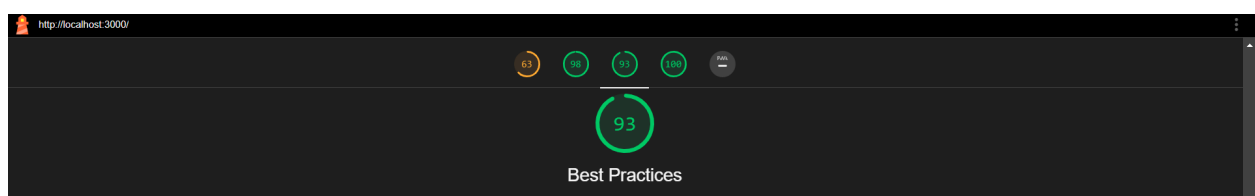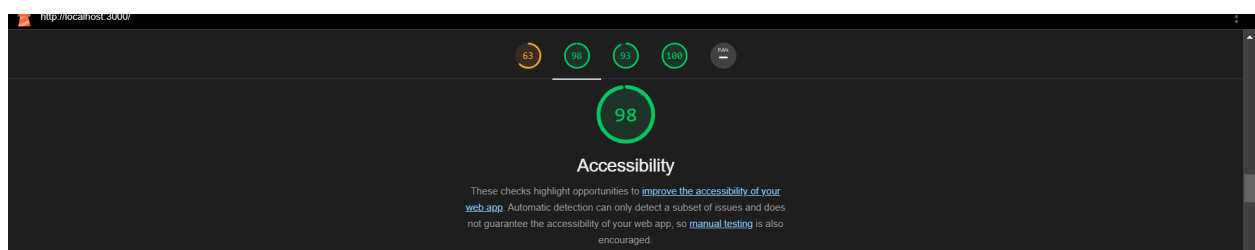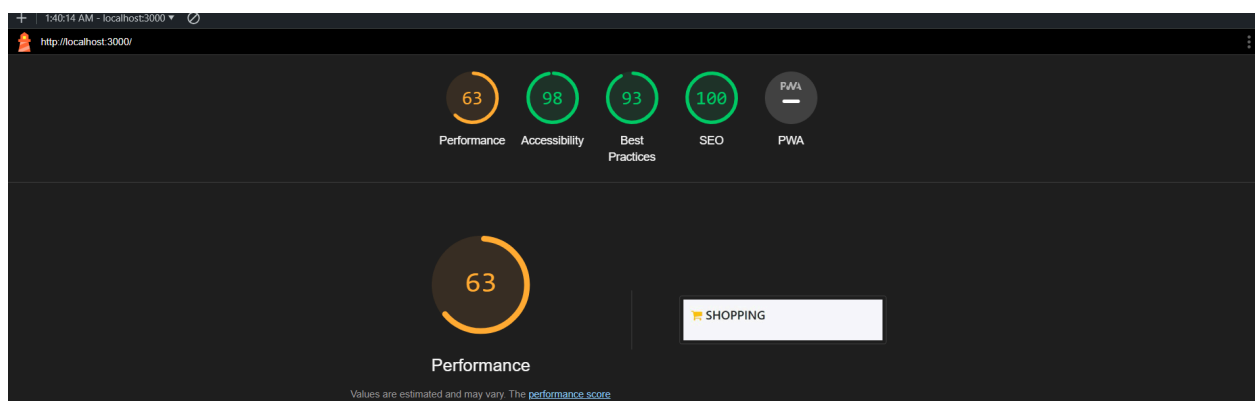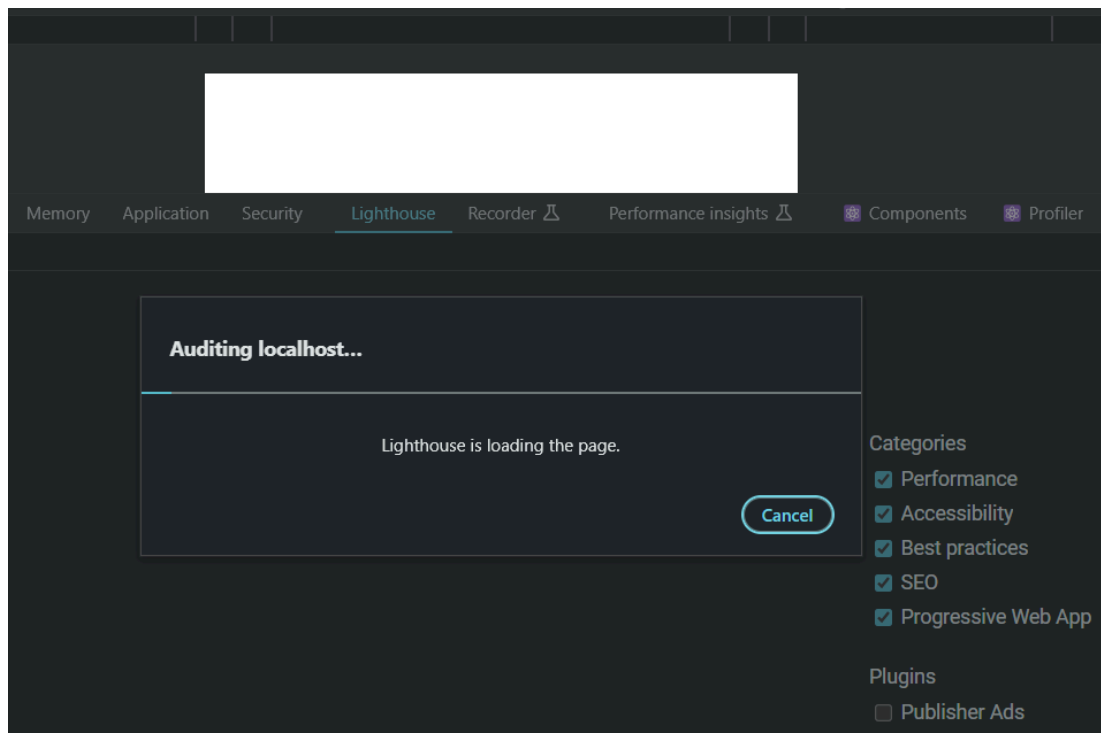
3.   **Accessibility:** As you might have guessed, this metric is a measure of how accessible your website is, across a plethora of accessibility features that can be implemented in your page (such as the 'aria-' attributes like aria-required, audio captions, button names, etc.). Unlike the other metrics though, Accessibility metrics score on a pass/fail basis i.e. if all possible elements of the page are not screen-reader friendly (HTML5 introduced features that would make pages easy to interpret for screen readers used by visually challenged people like tag names, tags such as <section>, <article>, etc.), you get a 0 on that score. The aggregate of these scores is your Accessibility metric score.
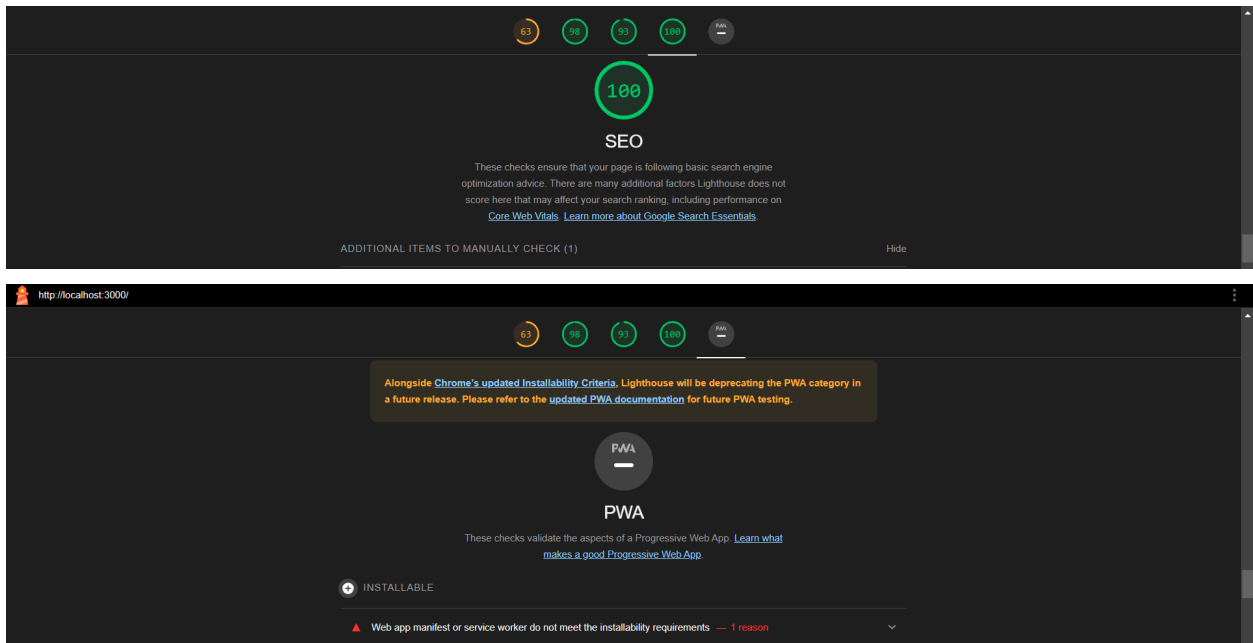
4.   **Best Practices:** As any developer would know, there are a number of practices that have been deemed 'best' based on empirical data. This metric is an aggregation of many such points, including but not limited to:Use of HTTPS

Avoiding the use of deprecated code elements like tags, directives, libraries, etc. Password input with paste-into disabled

Geo-Location and cookie usage alerts on load, etc.

Memory   Application   Security   Lighthouse   Recorder 🜇   Performance insights 🜇   ⚙ Components   ⚙ Profiler

Auditing localhost...

Lighthouse is loading the page.

Cancel

Categories
☑ Performance
☑ Accessibility
☑ Best practices
☑ SEO
☑ Progressive Web App

Plugins
☐ Publisher Ads

1:40:14 AM - localhost:3000 ▾ ⊘

http://localhost:3000/

63   98   93   100   PWA
Performance Accessibility Best   SEO   PWA
                            Practices

63

Performance

Values are estimated and may vary. The performance score

http://localhost:3000/

63   98   93   100   PWA

98

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.

http://localhost:3000/

63   98   93   100   PWA

93

Best Practices

🛒 SHOPPING

Changes made to manifest.json

```json
{
        "name":"Shopify e-commerce",
        "short_name":"Shopify e-commerce",
        "start_url":"index.html",
        "display":"standalone",
        "background_color":"#5900b3",
        "theme_color":"black",
        "scope": "./",
        "description":"Shopify e-commerce.",
        "icons":[
        {
        "src":"images/image.jpg",
        "sizes":"192x192",
        "type":"image/png",
        "purpose": "maskable"
        },
        {
        "src":"images/image.jpg",
        "sizes":"512x512",
        "type":"image/png"
    ,"purposes": "any maskable"
        },
    {
        "src": "images/image.jpg",
        "type": "image/png",
        "sizes":"144x144",
        "purpose": "any maskable"
```

```
    }
]
}
```
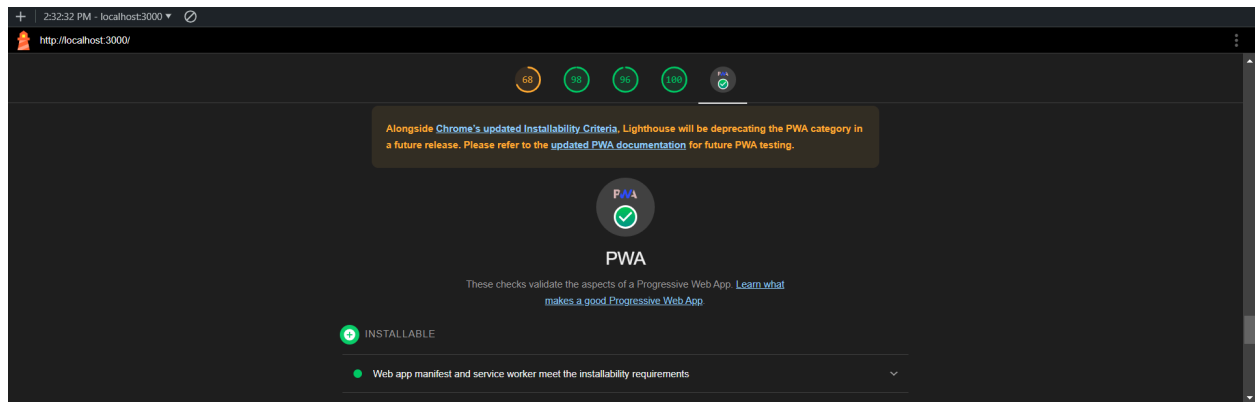

Conclusion: Therefore, created my first Progressive Web App by adding metadata to the index.html of my e-commerce web app and installed the Application on my local machine.