Experiment 11

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

Theory:

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:

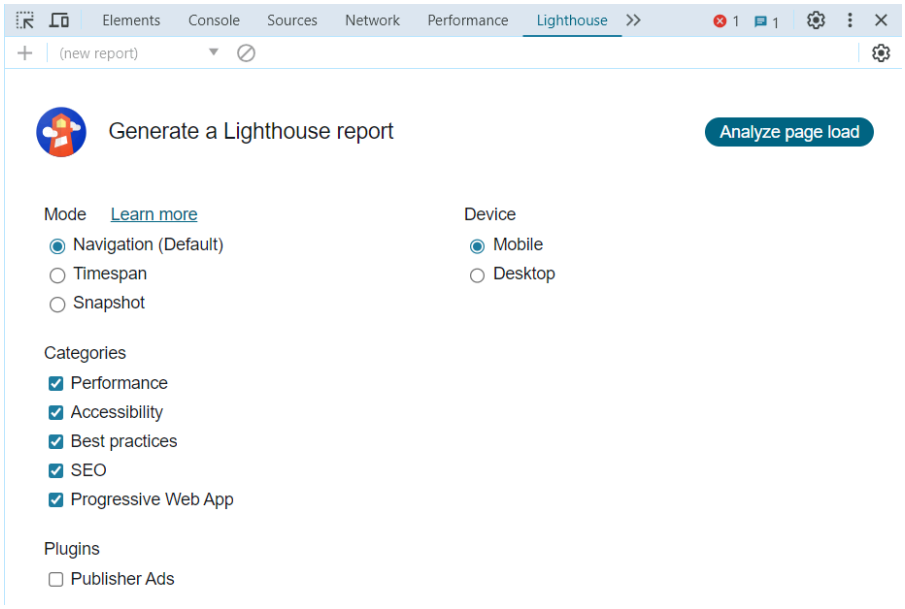
Performance: This score is an aggregation of how the page fared in aspects such as (but not limited to) loading speed, time taken for loading for basic frame(s), displaying meaningful content to the user, etc. To a layman, this score is indicative of how decently the site performs, with a score of 100 meaning that you figure in the 98th percentile, 50 meaning that you figure in the 75th percentile and so on.

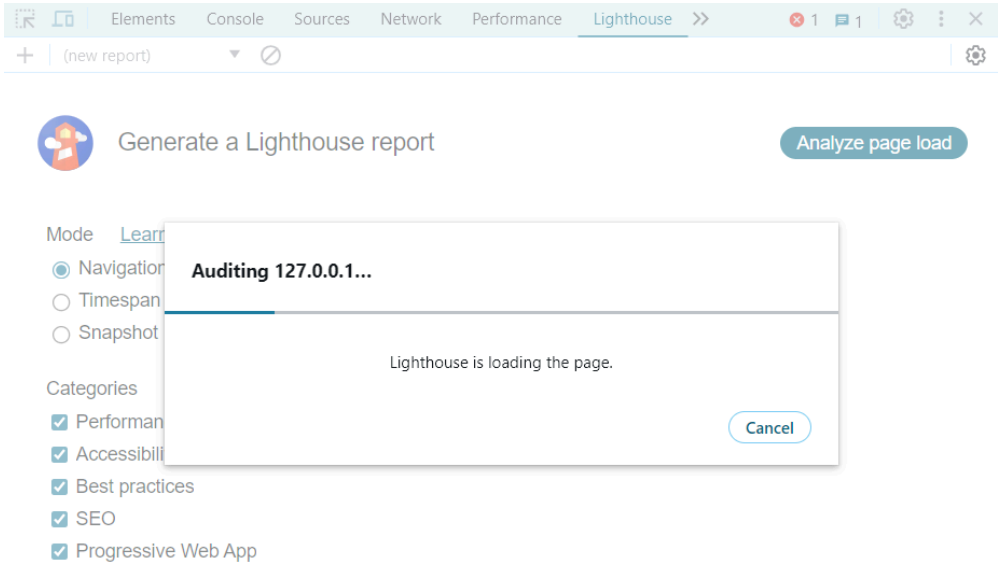
PWA Score (Mobile): Thanks to the rise of Service Workers, app manifests, etc., a lot of modern web applications are moving towards the PWA paradigm, where the objective is to make the application behave as close as possible to native mobile applications. Scoring points are based on the Baseline PWA checklist laid down by Google which includes Service Worker implementation(s), viewport handling, offline functionality, performance in script-disabled environments, etc.

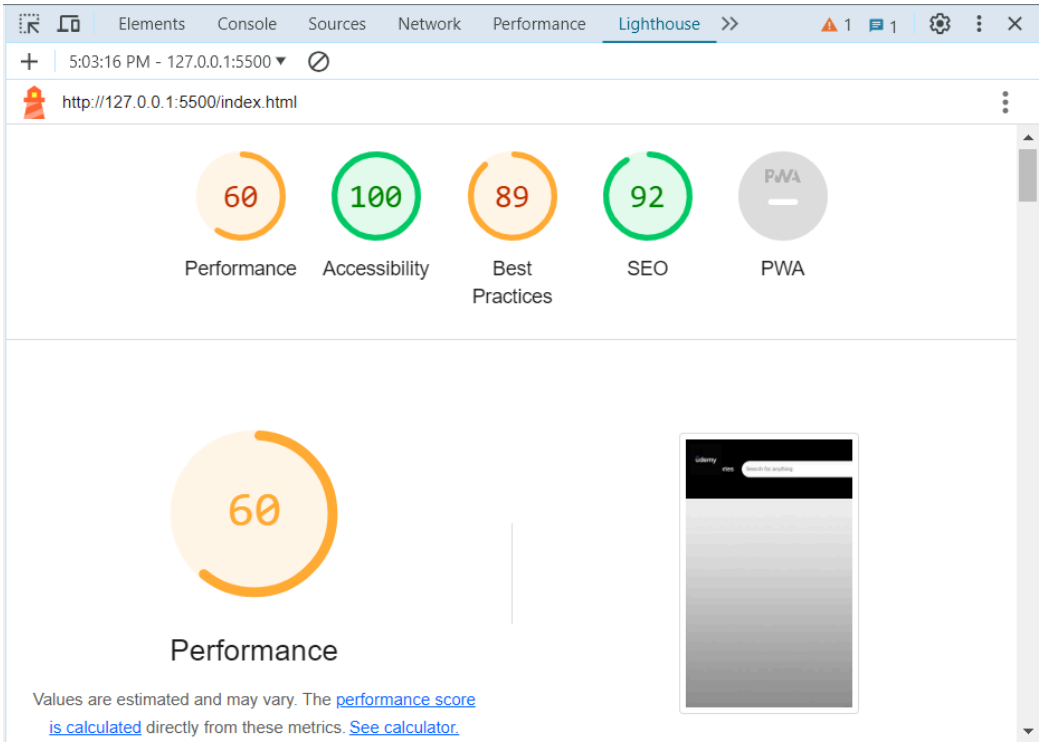
Accessibility: As you might have guessed, this metric is a measure of how accessible your website is, across a plethora of accessibility features that can be implemented in your page (such as the 'aria-' attributes like aria-required, audio captions, button names, etc.). Unlike the other metrics though, 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.

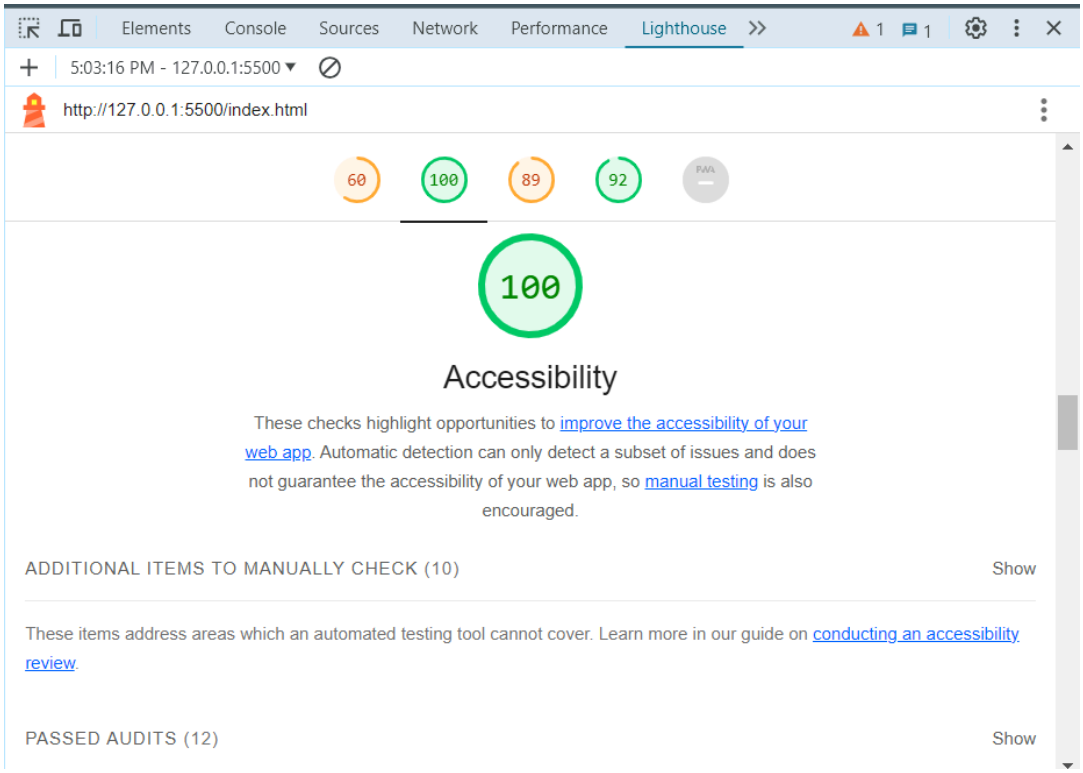
Best Practices: As any developer would know, there are a number of practices that have been deemed ‘best’ based on empirical data. This metric is an aggregation of many such points, including but not limited to:Use of HTTPS Avoiding the use of deprecated code elements like tags, directives, libraries, etc. Password input with paste-into disabled Geo-Location and cookie usage alerts on load, etc.

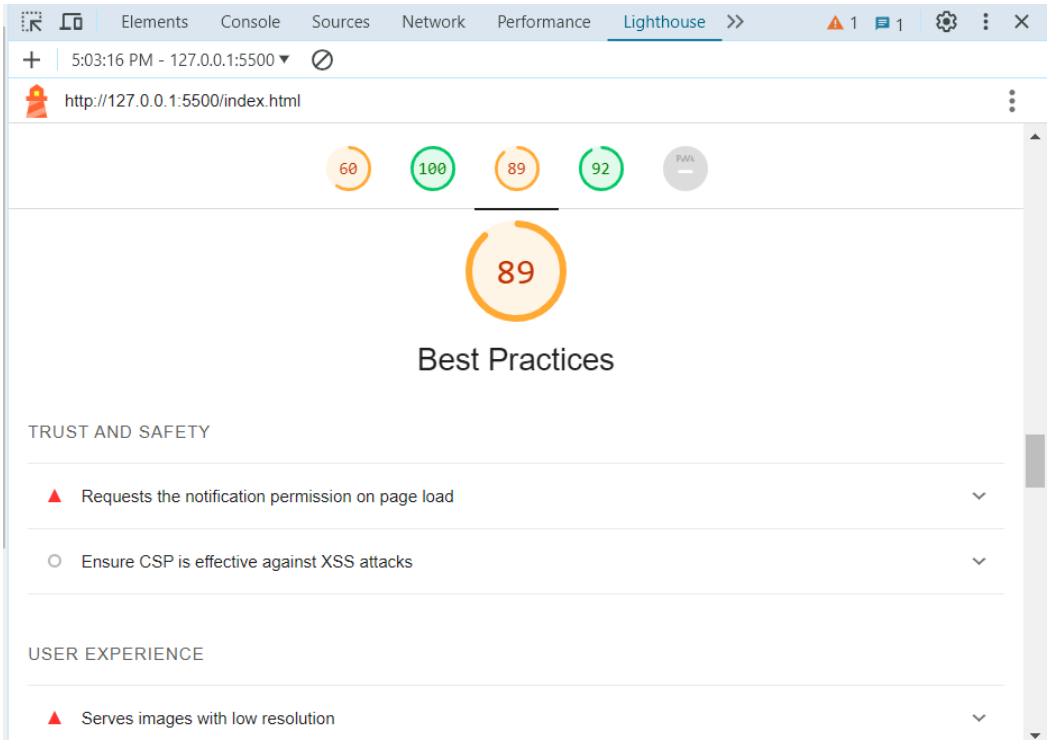
Code and Output:

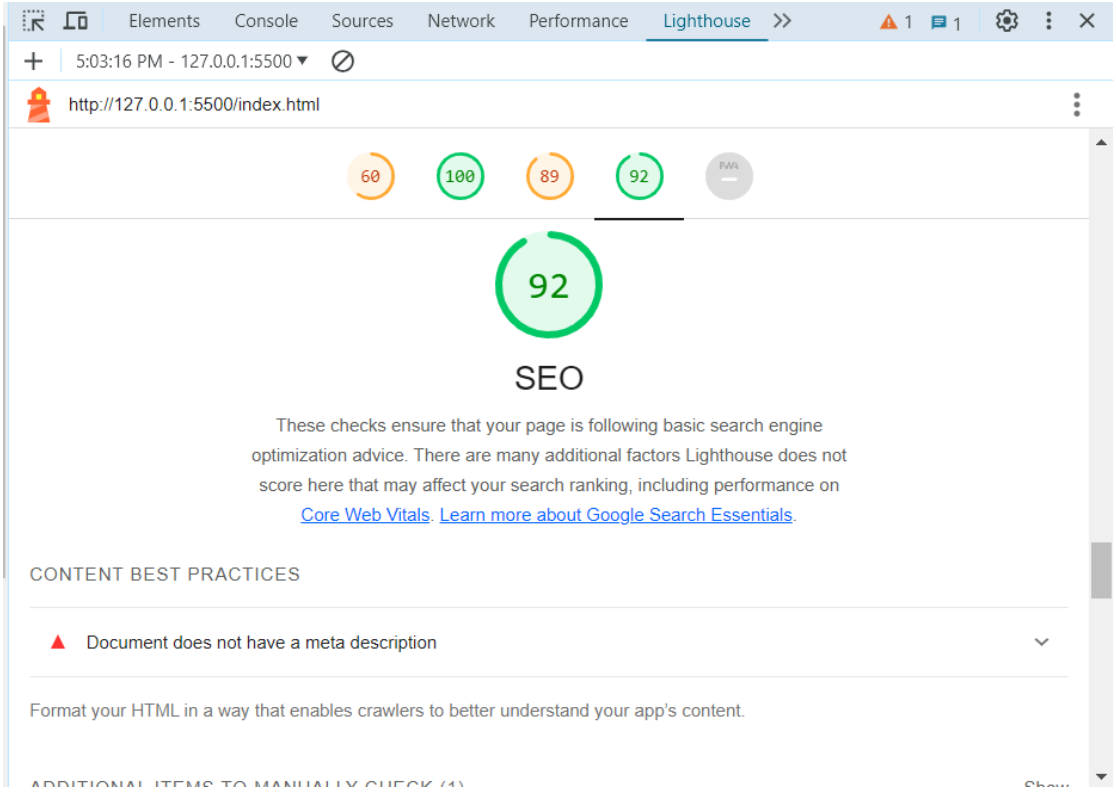





Elements
Console
Sources
Network
Performance
Lighthouse
5:03:16 PM - 127.0.0.1:5500
http://127.0.0.1:5500/index.html
60
Performance
100
Accessibility
89
Best Practices
92
SEO
PWA
60
Performance
Values are estimated and may vary. The performance score is calculated directly from these metrics. See calculator.

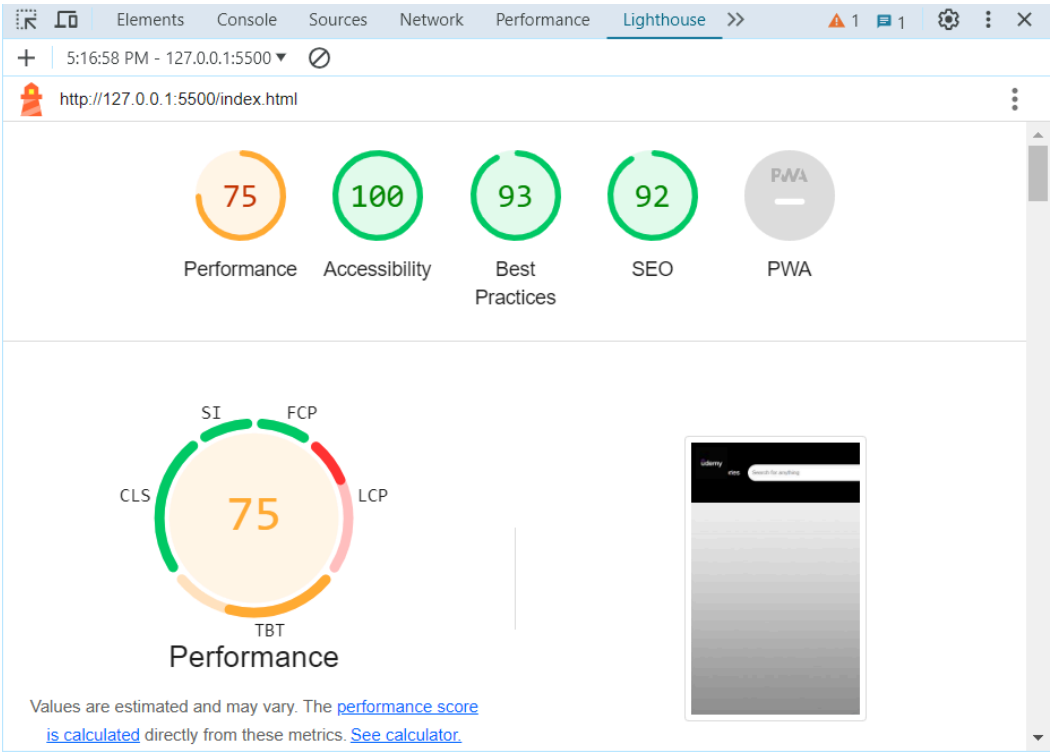

Elements
Console
Sources
Network
Performance
Lighthouse
5:03:16 PM - 127.0.0.1:5500
http://127.0.0.1:5500/index.html
60
100
89
92
100
Accessibility
These checks highlight opportunities to improve the accessibility of your web app. Automatic detection can only detect a subset of issues and does not guarantee the accessibility of your web app, so manual testing is also encouraged.
ADDITIONAL ITEMS TO MANUALLY CHECK (10)
Show
These items address areas which an automated testing tool cannot cover. Learn more in our guide on conducting an accessibility review.
PASSED AUDITS (12)
Show


Elements
Console
Sources
Network
Performance
Lighthouse
5:03:16 PM - 127.0.0.1:5500
http://127.0.0.1:5500/index.html
60
100
89
92
89
Best Practices
TRUST AND SAFETY
Requests the notification permission on page load
Ensure CSP is effective against XSS attacks
USER EXPERIENCE
Serves images with low resolution


Elements
Console
Sources
Network
Performance
Lighthouse
5:03:16 PM - 127.0.0.1:5500
http://127.0.0.1:5500/index.html
60
100
89
92
92
SEO
These checks ensure that your page is following basic search engine optimization advice. There are many additional factors Lighthouse does not score here that may affect your search ranking, including performance on Core Web Vitals. Learn more about Google Search Essentials.
CONTENT BEST PRACTICES
Document does not have a meta description
Format your HTML in a way that enables crawlers to better understand your app's content.

After the changes in manifest.json:


```
{
 "name": "Udemy App",
 "short_name": "Udemy",
 "start_url": "index.html",
 "scope": ".",
 "description": "This is a Udemy App.",
 "icons": [
   {
    "src": "images/icon.png",
    "sizes": "3219x3219",
    "type": "image/png",
    "purpose": "maskable"
   }
 ],
 "screenshots": [
   {
    "src": "images/Screenshot.png",
    "sizes": "3642x1751",
    "type": "image/png",
    "purpose": "any",
    "platform": "web",
    "form_factor": "wide",
    "orientation": "landscape"
   }
 ],
 "theme_color": "white",
 "background_color": "black",
 "display": "standalone"
}
```



Conclusion:

We analyzed the website with the help of a lighthouse tool and found some issues with the website. Hence we made changes in the manifest.json file we added "purpose":"any maskable" for a maskable error face.