Name: Atif Ansari   Roll no: 04   Class: D15B

MPL Experiment - 11

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

Theory:

Google Lighthouse is an open-source, automated tool developed by Google to analyze and audit web applications. It provides detailed reports on various performance aspects of a website, including speed, accessibility, and adherence to Progressive Web App (PWA) standards.

Key Features and Audit Metrics:

Google Lighthouse evaluates web applications based on multiple factors:

1. **Performance:** Measures the loading speed, rendering times, and how quickly a website becomes interactive. A high performance score indicates an optimized and fast-loading website.

2. **PWA Score:** Assesses whether the web application follows Google's PWA guidelines, including Service Worker implementation, offline capabilities, and mobile responsiveness.

3. **Accessibility:** Ensures that the website is user-friendly for all users, including those with disabilities, by analyzing ARIA attributes, semantic HTML ~~pages~~ usage, and screen-reader compatibility.

4. **Best Practices:** Checks for security implementations like HTTPS usage, modern JavaScript practices, and user-friendly experiences such as clear

cookie alerts and password protection measures.

To enhance PWA functionality, modifications were made to the manifest.json file and index.html including:
- Adding a theme color using <meta name = "theme-color" content = "#4285f4">
- Implementing a maskable icon in the manifest file.
- Adding an Apple Touch icon for better compatibility.

Conclusion:

In this experiment, we successfully utilized Google Lighthouse to analyze the performance and PWA compatibility of our web application. The tool provided insights into various aspects such as loading speed, accessibility compliance, and adherence to best practices. By making necessary adjustments to the manifest file and implementing features like theme color and maskable icons, we improved the applications overall PWA score.

The experiment demonstrated the importance of auditing a web app's performance and user experience using Google Lighthouse. By addressing identified issues, developers can create web applications that are faster, more accessible, and better optimized for mobile and offline use. This ensures a higher level of user satisfaction and compliance with modern web standards.

Implementation of Experiment 11:

Service-worker.js

```javascript
const CACHE_NAME = "sneakcart-v2";
const ASSETS_TO_CACHE = [
 "/Sneakcart/",
 "/Sneakcart/index.html",
 "/Sneakcart/style.css",
 "/Sneakcart/script.js",
 "/Sneakcart/manifest.json",
 "/Sneakcart/icons/icon-192.png",
 "/Sneakcart/icons/icon-512.png",
 "/Sneakcart/products/shoe1.png",
 "/Sneakcart/products/shoe2.jpg",
 "/Sneakcart/products/shoe3.jpg", // ➕ Add any additional assets here
 "https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.0/css/all.min.css",
 "https://fonts.googleapis.com/css2?family=Poppins:wght@300;400;500;600;700&display=swap"
];


// ✅ Install
self.addEventListener("install", (event) => {
 console.log("📄 Service Worker installing...");
 event.waitUntil(
  caches.open(CACHE_NAME).then((cache) => {
   console.log("✅ Caching app shell and assets");
   return cache.addAll(ASSETS_TO_CACHE);
  })
 );
 self.skipWaiting(); // Activate worker immediately
});

// ✅ Activate
self.addEventListener("activate", (event) => {
 console.log("🔃 Service Worker activating...");
 event.waitUntil(
  caches.keys().then((keyList) => {
   return Promise.all(
    keyList.map((key) => {
     if (key !== CACHE_NAME) {
```

```javascript
        console.log("🗑 Removing old cache:", key);
        return caches.delete(key);
      }
    })
  );
})
);
self.clients.claim(); // Take control immediately
});

// ✅ Fetch (Offline-first)
self.addEventListener("fetch", (event) => {
  event.respondWith(
    caches.match(event.request).then((cachedResponse) => {
      return (
        cachedResponse ||
        fetch(event.request).catch(() => {
          // Offline fallback for navigation requests
          if (event.request.mode === "navigate") {
            return caches.match("/Sneakcart/index.html");
          }
        })
      );
    })
  );
});

// ✅ Background Sync
self.addEventListener("sync", (event) => {
  if (event.tag === "sync-dummy-form") {
    event.waitUntil(
      // Simulate form re-submission or API retry
      new Promise((resolve) => {
        console.log("🔁 Background Sync triggered!");
        setTimeout(resolve, 2000);
      })
    );
  }
});
```

```javascript
// ✅ Push Notifications (demo setup)
self.addEventListener("push", (event) => {
  const data = event.data ? event.data.text() : "🔥 Hot new drops just landed!";
  event.waitUntil(
    self.registration.showNotification("👟 SneakCart", {
      body: data,
      icon: "/Sneakcart/icons/icon-192.png",
      badge: "/Sneakcart/icons/icon-192.png"
    })
  );
});

self.addEventListener('install', (event) => {
  console.log("🗂 Service Worker installing...");
  event.waitUntil(
    caches.open(CACHE_NAME)
    .then(cache => {
      return cache.addAll(assets)
        .catch(err => {
          console.error(" ✖ Caching failed:", err);
        });
    })
  );
});
```

Manifest.json:

```json
{
  "name": "SneakCart – Sneaker Store",
  "short_name": "SneakCart",
  "start_url": "/Sneakcart/index.html",
  "scope": "/Sneakcart/",
  "display": "standalone",
  "background_color": "#ffffff",
  "theme_color": "#000000",
  "orientation": "portrait-primary",
  "description": "Discover and shop limited edition premium sneakers. Built as a Progressive Web App (PWA) for the ultimate experience.",
  "icons": [
    {
```

```json
      "src": "/Sneakcart/icons/icon-192.png",
      "sizes": "192x192",
      "type": "image/png",
      "purpose": "any maskable"
    },
    {
      "src": "/Sneakcart/icons/icon-512.png",
      "sizes": "512x512",
      "type": "image/png",
      "purpose": "any maskable"
    }
  ],
  "id": "/Sneakcart/",
  "categories": ["shopping", "fashion", "sneakers", "lifestyle"],
  "screenshots": [
    {
      "src": "/Sneakcart/screenshots/home.png",
      "sizes": "1080x1920",
      "type": "image/png",
      "label": "Homepage"
    },
    {
      "src": "/Sneakcart/screenshots/product.png",
      "sizes": "1080x1920",
      "type": "image/png",
      "label": "Product Page"
    }
  ],
  "lang": "en"
}
```

Output: