

Experiment 08

Aim: To code and register a service worker, and complete the install and activation process for a new service worker for the E-commerce PWA

Theory:

A service worker is a background script that intercepts network requests. Installing and activating it is the first step toward enabling offline features and caching in a PWA.

Service Worker in Progressive Web Apps (PWAs)

A Service Worker is a script that the browser runs in the background, separate from a web page, enabling features such as offline access, background synchronization, and push notifications. It is a core technology in building Progressive Web Apps (PWAs) that function reliably regardless of network conditions.

1. What is a Service Worker?

A Service Worker acts as a network proxy between the web application and the internet. It intercepts network requests, giving developers the ability to:

- Control caching strategies,
- Pre-fetch and cache assets,
- Serve content from the cache when offline.

2. Service Worker Lifecycle

The service worker lifecycle includes three main phases:

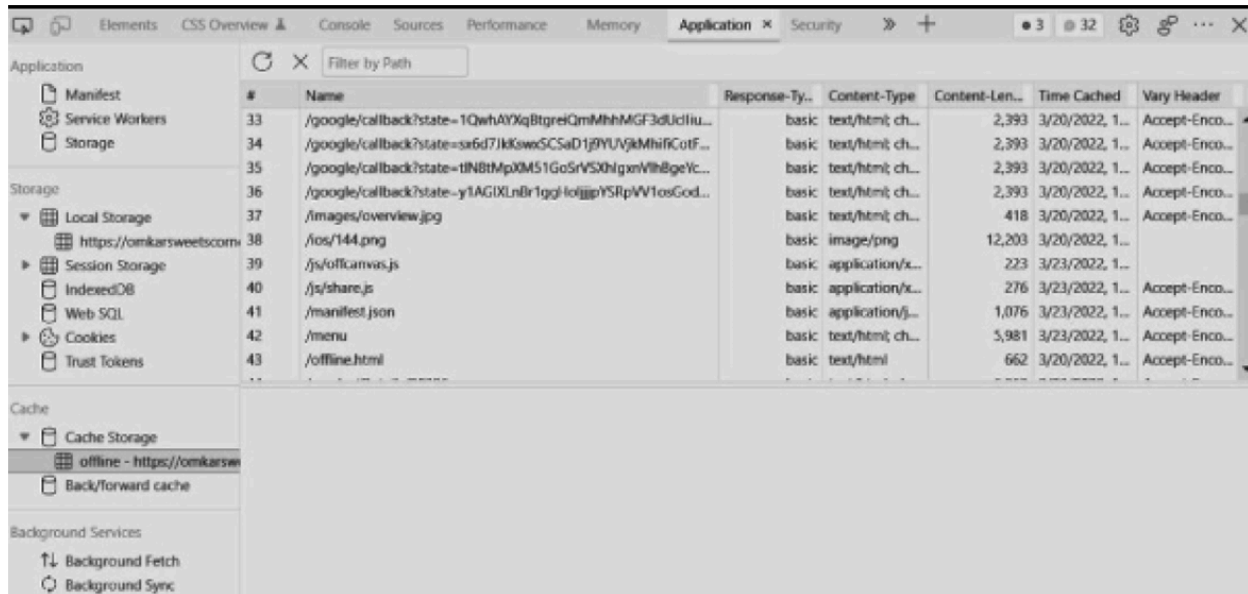
- Registration
The service worker is linked to the browser via the main JavaScript file.
- Installation
During this phase, static assets such as HTML, CSS, JavaScript, and images are cached.
- Activation
The service worker takes control of all pages under its scope and cleans up any old caches if necessary.

3. Registration Process

To register a service worker, include the following code in your main JavaScript file:

```
if ('serviceWorker' in navigator) {  
  window.addEventListener('load', () => {  
    navigator.serviceWorker.register('/service-worker.js')  
      .then(registration => {  
        console.log('Service Worker registered with scope:', registration.scope);  
      })  
      .catch(error => {  
        console.log('Service Worker registration failed:', error);  
      });  
  });  
}
```

Once registered, the browser manages the lifecycle automatically, and the service worker is activated if all steps succeed.



	#	Name	Response-Ty..	Content-Type	Content-Len..	Time Cached	Vary Header
Application	33	/google/callback?state=1QwhAYXqBtgreiQmMhMGF3dUdliu...	basic	text/html; ch...	2,393	3/20/2022, 1...	Accept-Enco...
Service Workers	34	/google/callback?state=saf6d7IdKswSCSaD1jBYUvjdMhifCotf...	basic	text/html; ch...	2,393	3/20/2022, 1...	Accept-Enco...
Storage	35	/google/callback?state=tIN8tMp0M51GoSrV50higxVih8geYc...	basic	text/html; ch...	2,393	3/20/2022, 1...	Accept-Enco...
Local Storage	36	/google/callback?state=y1AGIXLn8r1ggjlojggpYSRpVV1osGod...	basic	text/html; ch...	2,393	3/20/2022, 1...	Accept-Enco...
Session Storage	37	/images/overview.jpg	basic	text/html; ch...	418	3/20/2022, 1...	Accept-Enco...
IndexedDB	38	/ios/144.png	basic	image/png	12,203	3/20/2022, 1...	Accept-Enco...
Web SQL	39	/js/offcanvas.js	basic	application/x...	223	3/23/2022, 1...	Accept-Enco...
Cookies	40	/js/share.js	basic	application/x...	276	3/23/2022, 1...	Accept-Enco...
Trust Tokens	41	/manifest.json	basic	application/js...	1,076	3/23/2022, 1...	Accept-Enco...
	42	/menu	basic	text/html; ch...	5,981	3/23/2022, 1...	Accept-Enco...
	43	/offline.html	basic	text/html	662	3/20/2022, 1...	Accept-Enco...
Cache							
Cache Storage							
Offline - https://omkarswe...							
Back/forward cache							
Background Services							
Background Fetch							
Background Sync							

Conclusion:

The install and activate phases successfully set up our Society App's offline-ready PWA architecture.