

## Experiment No. 8

**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 :

#### Service Worker

Service Worker is a script that works on browser background without user interaction independently. Also, It resembles a proxy that works on the user side. With this script, you can track network traffic of the page, manage push notifications and develop “offline first” web applications with Cache API.

#### 1. Network Proxy:

- Service workers act as an intermediary between your web page and the network.
- They intercept all outgoing HTTP requests made by your application. They can choose how to handle these requests:
- Serve content from a local cache if available.

#### 2. Offline Capabilities:

- Service workers enable offline functionality by allowing caching of essential application resources (HTML, CSS, JavaScript, images).
- When a user is offline, the service worker can retrieve the requested content from the cache, providing a seamless experience even without an internet connection.

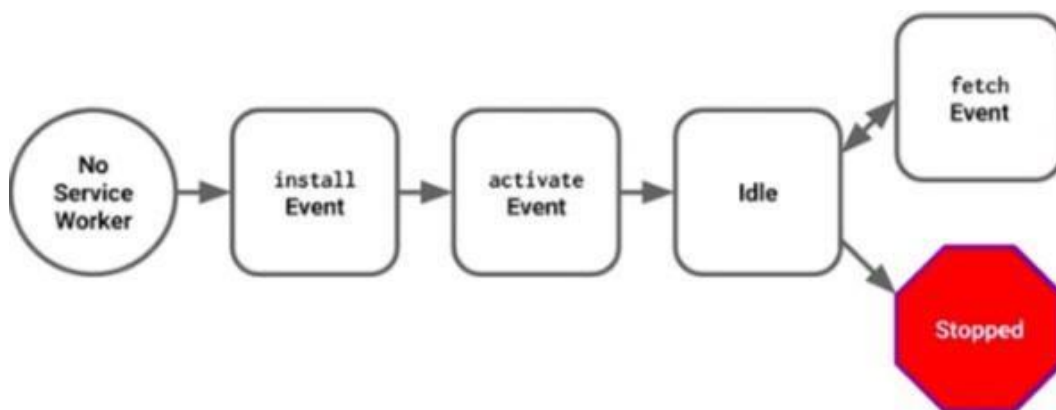
#### 3. HTTPS Requirement:

- Due to security concerns, service workers can only function on HTTPS connections.
- This ensures secure communication between the service worker, your application, and the server.

## What can we do with Service Workers?

- You can dominate Network Traffic  
You can manage all network traffic of the page and do any manipulations. For example, when the page requests a CSS file, you can send plain text as a response or when the page requests an HTML file, you can send a png file as a response. You can also send a true response too.
- You can Cache  
You can cache any request/response pair with Service Worker and Cache API and you can access these offline content anytime.
- You can manage Push Notifications  
You can manage push notifications with Service Worker and show any information message to the user.
- You can Continue  
Although Internet connection is broken, you can start any process with Background Sync of Service Worker.

## Service Worker Cycle



**Steps for coding and registering a service worker for your E-commerce PWA completing the install and activation process:**

## 1. Create the Service Worker File (sw.js):

```
JS sw.js > ...
1  self.addEventListener('install', function(event) {
2      event.waitUntil(
3          caches.open('offline')
4              .then(function(cache) {
5                  return cache.addAll([
6                      '/',
7                      '/index.html',
8                      // Add other essential static assets (CSS, JavaScript, images)
9                  ]);
10             });
11  });
12  });
13
14  self.addEventListener('fetch', function(event) {
15      event.respondWith(
16          fetch(event.request)
17              .catch(function() {
18                  return caches.match(event.request)
19                      .then(function(matching) {
20                          return matching || caches.match('offline.html');
21                      });
22              })
23      );
24  });
25  });
```

## 2. Register the Service Worker:

In your main JavaScript file (e.g., `main.js` or `app.js`), add the following code:

```
24  },
25
26  if ('serviceWorker' in navigator) {
27      navigator.serviceWorker.register('/sw.js')
28          .then(function(registration) {
29              console.log('Service worker registration successful:', registration.scope);
30          })
31          .catch(function(error) {
32              console.log('Service worker registration failed:', error);
33          });
34  }
35
```

## Output

Welcome to Our Store

Home Shop About Contact

Product 1  
Rs30000  
Buy

Product 2  
Rs.350000  
Buy

Product 3  
\$20.00  
Buy

Service workers

☐ Offline ☐ Update on reload ☐ Bypass for network

**http://127.0.0.1:5500/** [Network requests](#) [Update](#) [Unregister](#)

Source [serviceworker.js](#)  
Received 3/30/2024, 9:48:58 AM

Status ● #3762 activated and is stopped [start](#)

Push  [Push](#)

Sync  [Sync](#)

ServiceWorker registration successful with scope: <http://127.0.0.1:5500/> (index):57

Application

- Manifest
- Service workers
- Storage

Storage

- Local storage
- Session storage
- IndexedDB
- Web SQL
- Cookies
- Private state tokens
- Interest groups
- Shared storage
- Cache storage
- offline - http://127.0.0.1:5500/

Background services

- Back/forward
- Background fetch
- Background sync

Filter by Path

http://127.0.0.1:5500

Origin http://127.0.0.1:5500

Bucket name default

Is persistent No

Durability strict

Quota 0 B

Expiration None

#	Name	Resp...	Conte...	Cont...	Time ...	Vary ...
0	/	basic	text/h...	2,984	17/3/...	Origin
1	/index.html	basic	text/h...	2,984	17/3/...	Origin

**Conclusion :** I have understood and successfully registered a service worker, and completed the install and activation process for a new service worker for the E-commerce PWA.

