

EXPERIMENT NO : 09**MAD and PWA Lab****1. Aim:**

To implement Service worker events like fetch, sync, and push for E-commerce PWA.

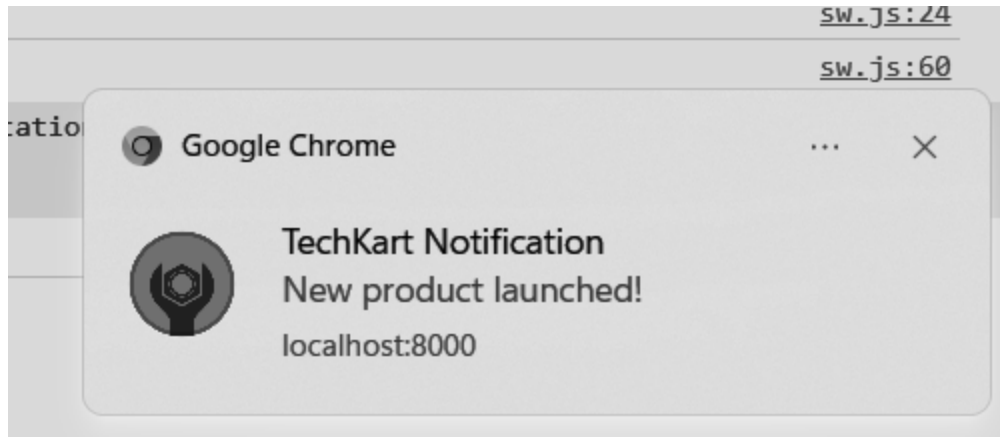
2. Description:

This experiment focuses on implementing key Service Worker events that enhance PWA functionality: fetch, sync, and push. These events enable PWAs to manage network requests, handle background data synchronization, and deliver push notifications.

- Key Service Worker Events:
 - Fetch Event: Allows the Service Worker to intercept and handle network requests. This is crucial for implementing caching strategies (e.g., "cache first," "network first") and providing offline functionality.
 - Sync Event: Enables background data synchronization when the network connection is available. This allows PWAs to defer actions until a stable connection is present.
 - Push Event: Handles push notifications received from a server, allowing PWAs to deliver timely updates and engage users.

The screenshot displays a web application interface for managing service worker events. At the top, the status is indicated as '#385 activated and is running' with a 'Stop' button. Below this, the 'Clients' section shows 'http://localhost:8000/' with a refresh icon. The 'Push' section contains a text input with the JSON object '{ "message": "New product launched!" }' and a 'Push' button. The 'Sync' section has a text input with 'test-tag-from-devtools' and a 'Sync' button. The 'Periodic sync' section also has a text input with 'test-tag-from-devtools' and a 'Periodic sync' button.

Status	● #385 activated and is running	Stop
Clients	http://localhost:8000/	
Push	<input type="text" value='{ "message": "New product launched!" }'/>	Push
Sync	<input type="text" value="test-tag-from-devtools"/>	Sync
Periodic sync	<input type="text" value="test-tag-from-devtools"/>	Periodic sync



```
> navigator.serviceWorker.ready.then(sw => {  
  sw.sync.register('sync-message');  
});  
◀ ▼ Promise {<fulfilled>: undefined} i  
  ▶ [[Prototype]]: Promise  
    [[PromiseState]]: "fulfilled"  
    [[PromiseResult]]: undefined  
[SW] Background sync triggered!
```

3. Conclusion:

This experiment demonstrates the implementation of fetch, sync, and push events in a Service Worker. These events are essential for creating robust, offline-capable, and engaging PWAs.

4. GitHub Link:

<https://github.com/atharvnikam38/techkart-pwa>