MPL Lab 9
Mayur Jaiswal
D15B
24

Aim: To implement advanced service worker events—fetch, sync, and push—for improved performance, background syncing, and notification capabilities in an E-commerce PWA.

Theory:

- fetch event: Lets you intercept network requests and serve cached responses or perform custom network strategies (cache-first, network-first).
- sync event: Ensures data is sent to the server when the network is available, useful for retrying failed operations.
- push event: Allows background notifications using the Push API, improving engagement even when the app is not open.

Steps to Perform:

Intercept Requests with fetch:

javascript

```
self.addEventListener('fetch', event => {
  event.respondWith(
    caches.match(event.request)
      .then(response => response || fetch(event.request))
  );
});
```

1.
2. **Add Background Sync:**

In service-worker.js:

javascript

```
self.addEventListener('sync', event => {
  if (event.tag === 'sync-cart') {
    event.waitUntil(sendCartToServer());
  }
});

function sendCartToServer() {
  // retry pending cart orders
}
```

○

In app JS:

javascript

```
navigator.serviceWorker.ready.then(swReg => {
  return swReg.sync.register('sync-cart');
});
```

○

3. **Push Notifications:**

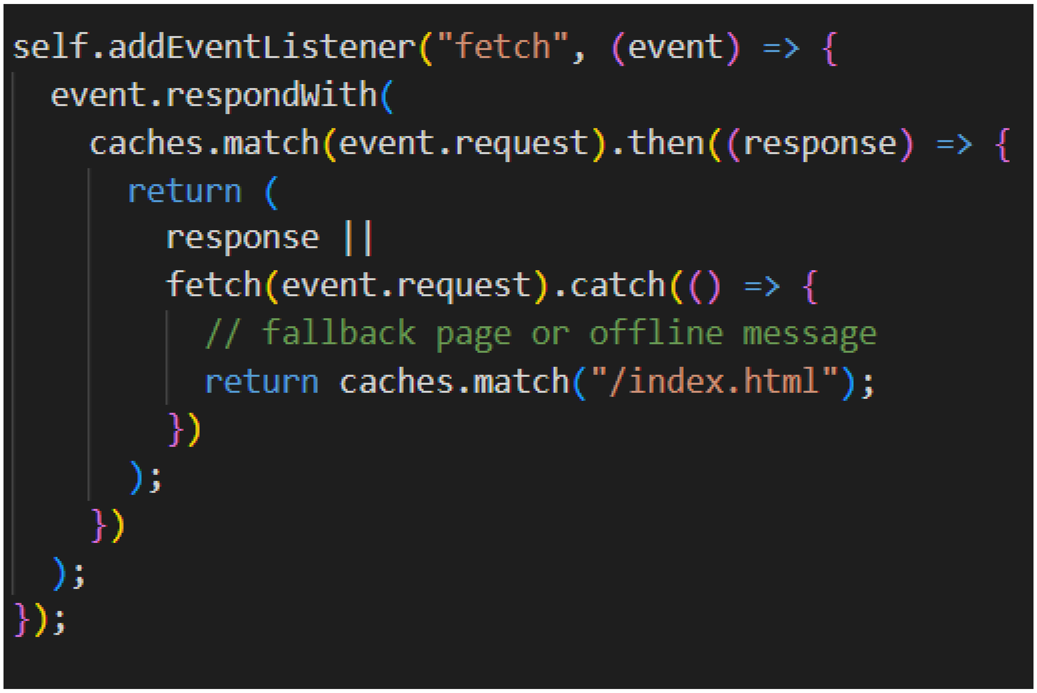
In service-worker.js:

javascript

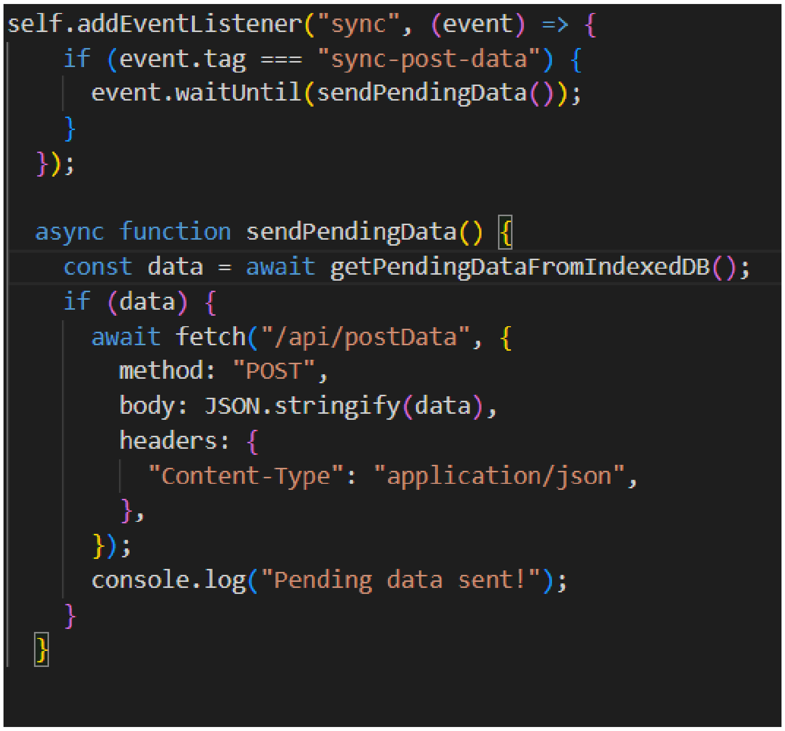
```
self.addEventListener('push', event => {
  const data = event.data.json();
  self.registration.showNotification(data.title, {
    body: data.body,
    icon: 'icons/icon-192x192.png',
  });
});
```

Key Features:

- **Smart Caching: Improve performance and offline support with fetch control.**
- **Background Reliability: Retry failed tasks when online using sync.**
- **User Engagement: Push notifications improve return rates.**
- **Seamless UX: Users enjoy uninterrupted service even with bad networks.**

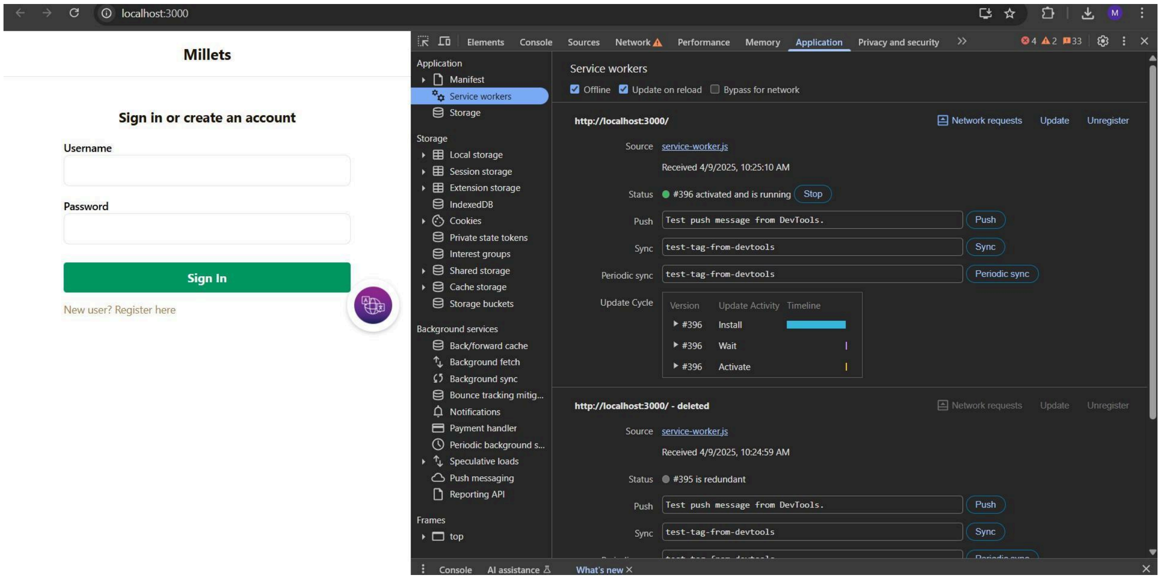


```
self.addEventListener("fetch", (event) => {
  event.respondWith(
    caches.match(event.request).then((response) => {
      return (
        response ||
        fetch(event.request).catch(() => {
          // fallback page or offline message
          return caches.match("/index.html");
        })
      );
    })
  );
});
```



```
self.addEventListener("sync", (event) => {
    if (event.tag === "sync-post-data") {
      event.waitUntil(sendPendingData());
    }
  });

  async function sendPendingData() {
    const data = await getPendingDataFromIndexedDB();
    if (data) {
      await fetch("/api/postData", {
        method: "POST",
        body: JSON.stringify(data),
        headers: {
          "Content-Type": "application/json",
        },
      });
      console.log("Pending data sent!");
    }
  }
```



Millets

Sign in or create an account

Username

Password

New user? Register here

Service workers

☑ Offline ☑ Update on reload □ Bypass for network

http://localhost:3000/

Source service-worker.js

Received 4/9/2025, 10:25:10 AM

Status ● #396 activated and is running

Push `Test push message from DevTools.`

Sync `test-tag-from-devtools`

Periodic sync `test-tag-from-devtools`

Update Cycle

Version	Update Activity	Timeline
#396	Install	
#396	Wait	
#396	Activate	

http://localhost:3000/ - deleted

Source service-worker.js

Received 4/9/2025, 10:24:59 AM

Status ● #395 is redundant

Push `Test push message from DevTools.`

Sync `test-tag-from-devtools`