

Name : Abhay Gupta

MPL Practical 09

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

Theory:

In Progressive Web Apps (PWAs), service workers allow us to handle advanced background tasks that improve performance, reliability, and user engagement. Three important events supported by service workers are fetch, sync, and push.

- The fetch event lets us intercept network requests and serve cached content when offline, helping the app work even without the internet.
- The sync event allows us to delay data synchronization with the server until the user is back online, ensuring reliability.
- The push event enables the app to receive push notifications in the background, even when the web page isn't open.

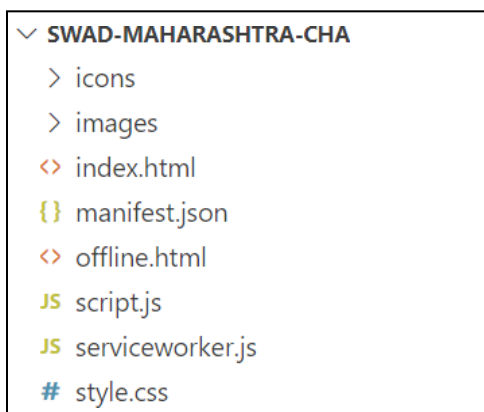
What We Implemented

In our "Swad Maharashtra Cha" PWA, we enhanced the service worker to support these three events:

- Fetch Event: We used a cache-first strategy for same-origin requests and a network-first approach for others. If a fetch fails, we show a custom [offline.html](#) page.
- Sync Event: We registered a background sync with the tag "sync-data" to simulate background syncing once connectivity is restored.
- Push Event: We implemented push notification handling, so the service worker can display alerts (like offers or updates) when a message is received.

This makes our eCommerce app more reliable, responsive, and user-friendly — even with unstable or no internet connection.

Folder Structure:



Code:

index.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
```

```

<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<!-- PWA Theme Color -->
<meta name="theme-color" content="#b22222" />

<!-- SEO Meta Tags -->
<meta
  name="description"
  content="Authentic Maharashtrian Spices & Food – Buy famous spices, masalas, and traditional items
like Puran Poli, Bhakarwadi, Modaks, and Chitale Bandhu snacks online."
/>
<meta
  name="keywords"
  content="Maharashtrian food, Goda masala, Bhakarwadi, Modaks, Puran Poli, Chitale Bandhu
snacks, Malvani masala, online spice store"
/>
<meta name="author" content="Swad Maharashtra Cha" />

<title>Swad Maharashtra Cha - Authentic Maharashtrian Food</title>

<!-- Manifest File for PWA -->
<link rel="manifest" href="manifest.json" />

<!-- Stylesheet -->
<link rel="stylesheet" href="style.css" />

<!-- Google Fonts -->
<link
href="https://fonts.googleapis.com/css2?family=Playfair+Display:wght@400;700&family=Poppins:wght
@300;400;500;600&display=swap"
rel="stylesheet"
/>
</head>
<body>

<script>
  if ("serviceWorker" in navigator) {
    window.addEventListener("load", () => {
      navigator.serviceWorker
        .register("/serviceworker.js")
        .then((registration) => {
          console.log("✓ Service Worker registered! Scope:", registration.scope);

          // ⓘ Request Push Notification Permission if
          ("PushManager" in window) {
            Notification.requestPermission().then((permission) => {
              if (permission === "granted") {

```

```

        console.log("🔔 Push notifications granted.");
    } else {
        console.log("🔔 Push notifications denied.");
    }
});
}

// 🔔 Register Background Sync if
("SyncManager" in window) {
    navigator.serviceWorker.ready.then((swReg) => {
        swReg.sync.register("sync-data").then(() => {
            console.log("🔔 Sync registered");
        }).catch((err) => {
            console.log("❌ Sync registration failed:", err);
        });
    });
}
}).catch((error) => {
    console.log("❌ Service Worker registration failed:", error);
});
});
}
</script>
</body>
</html>

```

offline.html

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />

    <!-- PWA Theme Color -->
    <meta name="theme-color" content="#8c0303" />

    <title>Offline - Swad Maharashtra Cha</title>
</head>
<body>

    <h1>You're Currently Offline</h1>

    <p>Looks like your connection to the flavorful world of Maharashtra has been temporarily lost. Please
    check your internet connection and try again.</p>

    <p>Don't worry, our delicious spices and treats will be waiting for you when you're back online!</p>

    <button class="btn" onclick="window.location.reload()">Try Again</button>

```

```
<div class="decorative-line"></div>
</div>
</body>
</html>
```

serviceworker.js

```
const CACHE_NAME = "swad-maha-cache-v1";
const FILES_TO_CACHE = [
  "index.html",
  "style.css",
  "script.js",
  "serviceworker.js",
  "manifest.json",
  "images/maharashtra-feast.webp",
  "images/cart.svg",
  "images/goda-masala.webp",
  "images/malvani-masala.webp",
  "images/kanda-lasun.webp",
  "images/puran-poli.webp",
  "images/modak.webp",
  "images/shrikhand.webp",
  "images/bhakarwadi.webp",
  "images/chivda.webp",
  "images/chakli.webp",
  "images/border-pattern.webp",
  "images/maharashtra-pattern.webp",
  "images/paithani-pattern.webp",
  "icons/icon-192x192.png",
  "icons/icon-512x512.png",
  "offline.html"
];

// Install Event
self.addEventListener("install", (event) => {
  console.log("[ServiceWorker] Install");
  event.waitUntil(
    caches.open(CACHE_NAME).then((cache) => {
      console.log("[ServiceWorker] Caching files");
      return cache.addAll(FILES_TO_CACHE);
    })
  );
});
```

// Activate Event

```
self.addEventListener("activate", (event) => {
  console.log("[ServiceWorker] Activate");
  event.waitUntil(
```

```

caches.keys().then((keyList) =>
  Promise.all(
    keyList.map((key) => {
      if (key !== CACHE_NAME) {
        console.log("[ServiceWorker] Removing old cache", key);
        return caches.delete(key);
      }
    })
  )
);
return self.clients.claim();
});

// Enhanced Fetch Event
self.addEventListener("fetch", (event) => {
  console.log("[ServiceWorker] Fetch", event.request.url);
  const requestURL = new URL(event.request.url);

  // If request is same-origin, use Cache First
  if (requestURL.origin === location.origin) {
    event.respondWith(
      caches.match(event.request).then((cachedResponse) => {
        return (
          cachedResponse ||
          fetch(event.request).catch(() => caches.match("offline.html"))
        );
      })
    );
  } else {
    // Else, use Network First
    event.respondWith(
      fetch(event.request)
        .then((response) => {
          return response;
        })
        .catch(() =>
          caches.match(event.request).then((res) => {
            return res || caches.match("offline.html");
          })
        )
    );
  }
});

// Sync Event (simulation)
self.addEventListener("sync", (event) => {
  if (event.tag === "sync-data") {
    event.waitUntil(

```

```

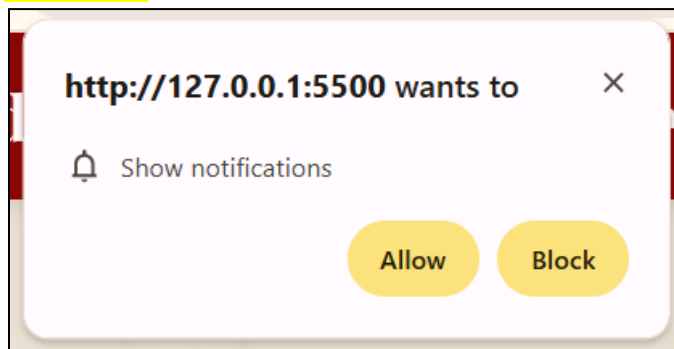
(async () => {
  console.log("Sync event triggered: 'sync-data'");
  // Here you can sync data with server when online
})();
);
}
});

// Push Event
self.addEventListener("push", function (event) {
  if (event && event.data) {
    let data = {};
    try {
      data = event.data.json();
    } catch (e) {
      data = {
        method: "pushMessage",
        message: event.data.text(),
      };
    }

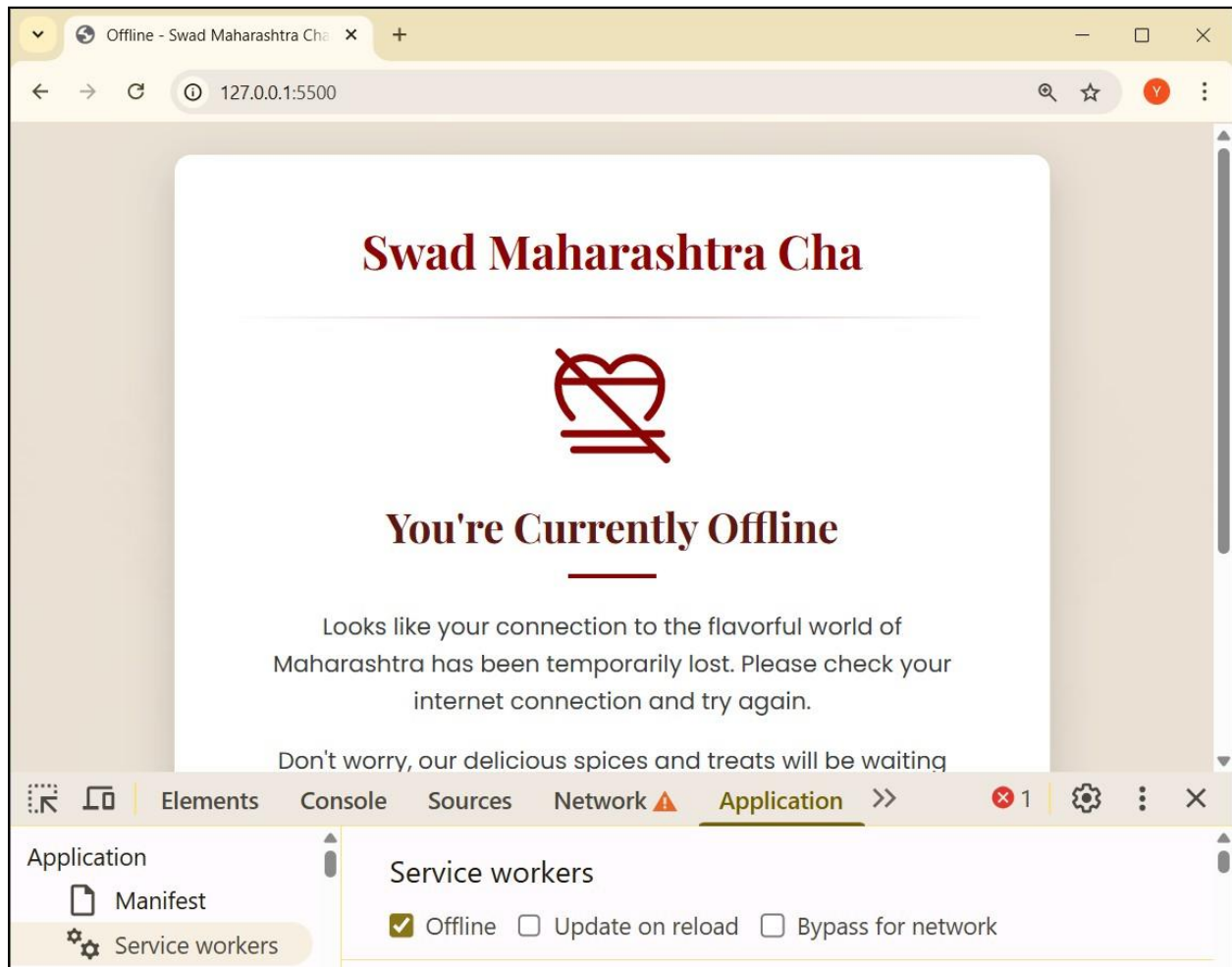
    if (data.method === "pushMessage") {
      console.log("Push notification sent");
      event.waitUntil(
        self.registration.showNotification("Swad Maharashtra Cha", {
          body: data.message,
        })
      );
    }
  }
});

```

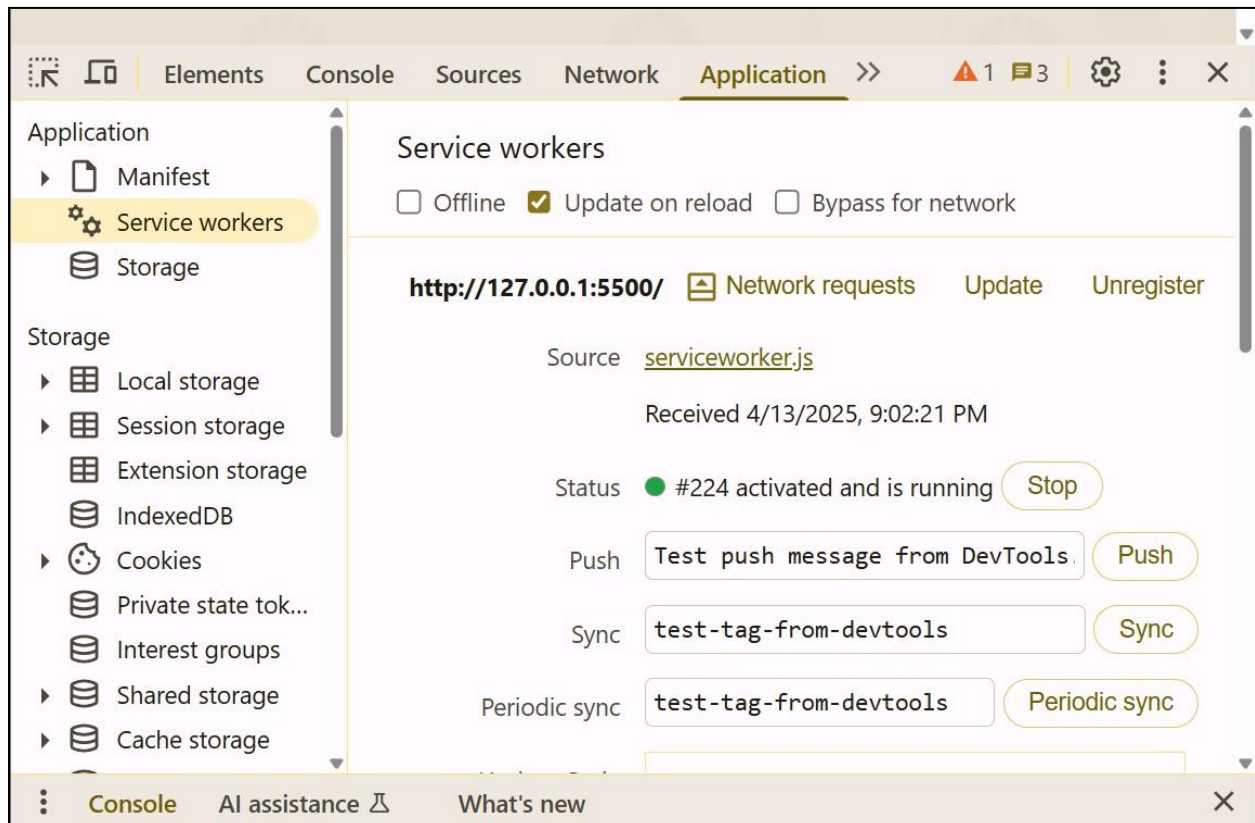
Screenshot:

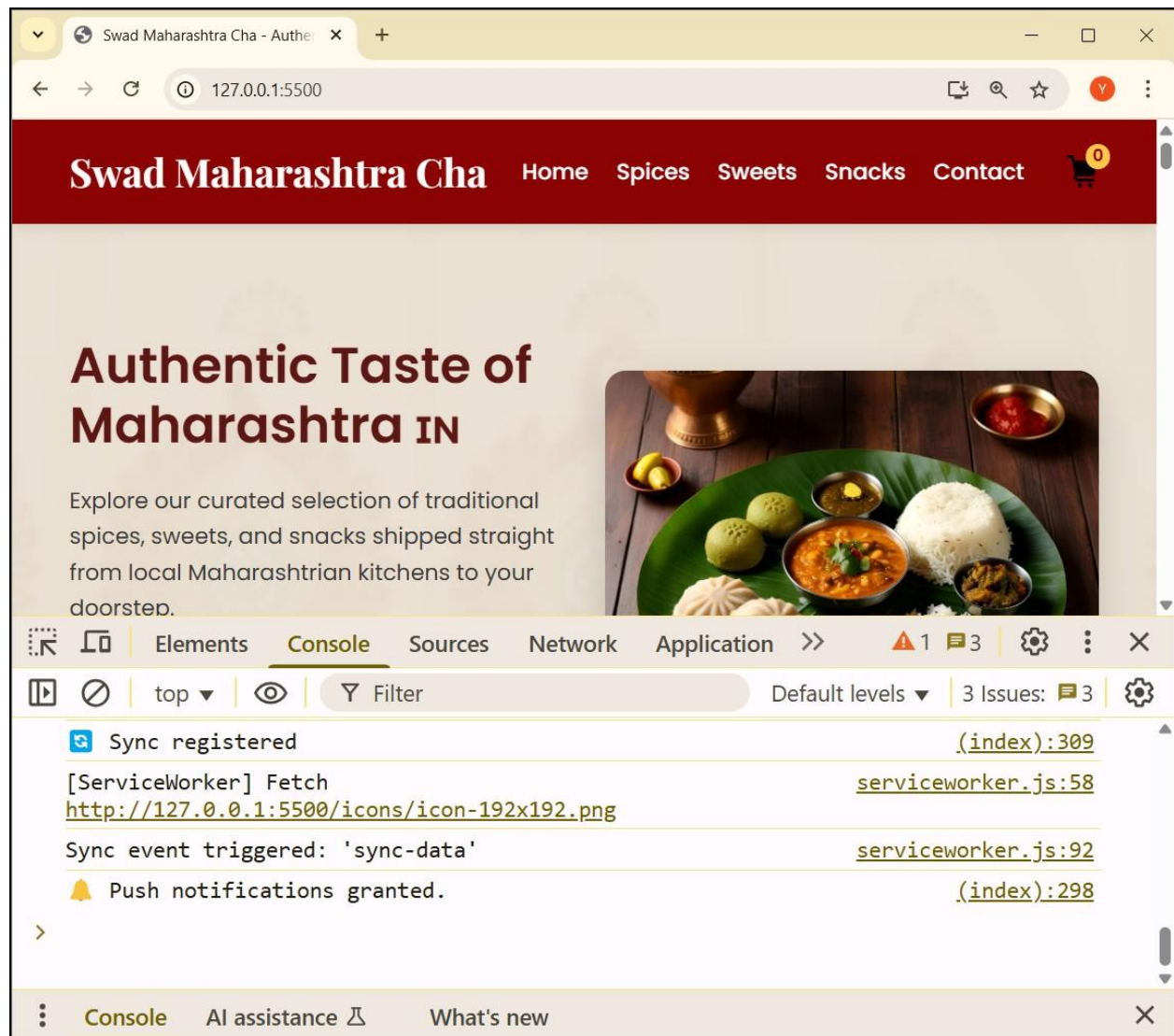


1. Test: Fetch Event (Offline Support)

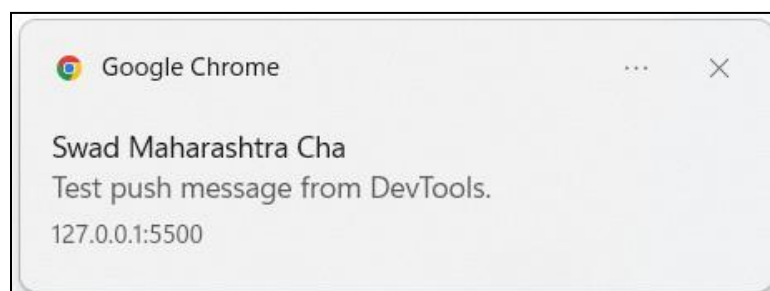


2. Test: Background Sync Event





3. Test: Push Notification Event



The screenshot shows a web browser window with the address bar displaying '127.0.0.1:5500'. The website, 'Swad Maharashtra Cha', has a dark red header with navigation links: Home, Spices, Sweets, Snacks, and Contact. A shopping cart icon with a '0' is in the top right. The main content area features the heading 'Authentic Taste of Maharashtra IN' and a paragraph: 'Explore our curated selection of traditional spices, sweets, and snacks shipped straight from local Maharashtrian kitchens to your doorstep.' Below the text is an image of a traditional Maharashtrian meal (Pongal) served on a banana leaf. The Chrome DevTools console is open, showing the following log entries:

- Sync registered .(index):309
- [ServiceWorker] Fetch serviceworker.js:58
<http://127.0.0.1:5500/icons/icon-192x192.png>
- Sync event triggered: 'sync-data' serviceworker.js:92
- Push notifications granted. .(index):298

The bottom of the console shows tabs for 'Console', 'AI assistance', and 'What's new'.

Conclusion:

In this experiment, we implemented fetch, sync, and push events in our Swad Maharashtra Cha PWA to enable offline support, background syncing, and push notifications. Initially, push notifications were not showing due to missing permission handling in the browser, which we resolved by adding a proper `Notification.requestPermission()` block during service worker registration.