MPL Practical 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 JavaScript file that runs in the background and helps make a PWA work offline by caching important files. It also allows features like background sync, push notifications, and faster loading even with a slow or no internet connection.

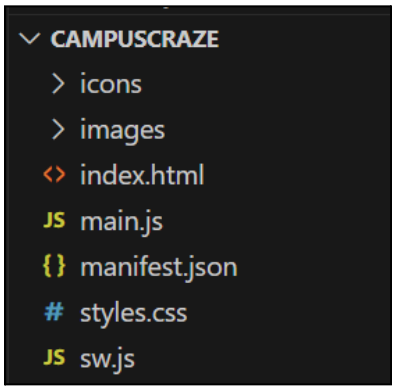
In this experiment, we created and registered a service worker for our e-commerce PWA CampusCraze. The service worker listens to two main events:

- The install event, where it caches essential files like HTML, CSS, images, icons, and fonts to make the app available offline.
- The activate event, where it removes old cached data to keep the app updated and clean.

We also wrote JavaScript in the index.html file to register the service worker when the page loads. Once registered, it caches all listed assets and allows the app to work smoothly even without an internet connection.

This setup helps improve the performance and offline capability of the CampusCraze PWA.

Folder Structure



index.html

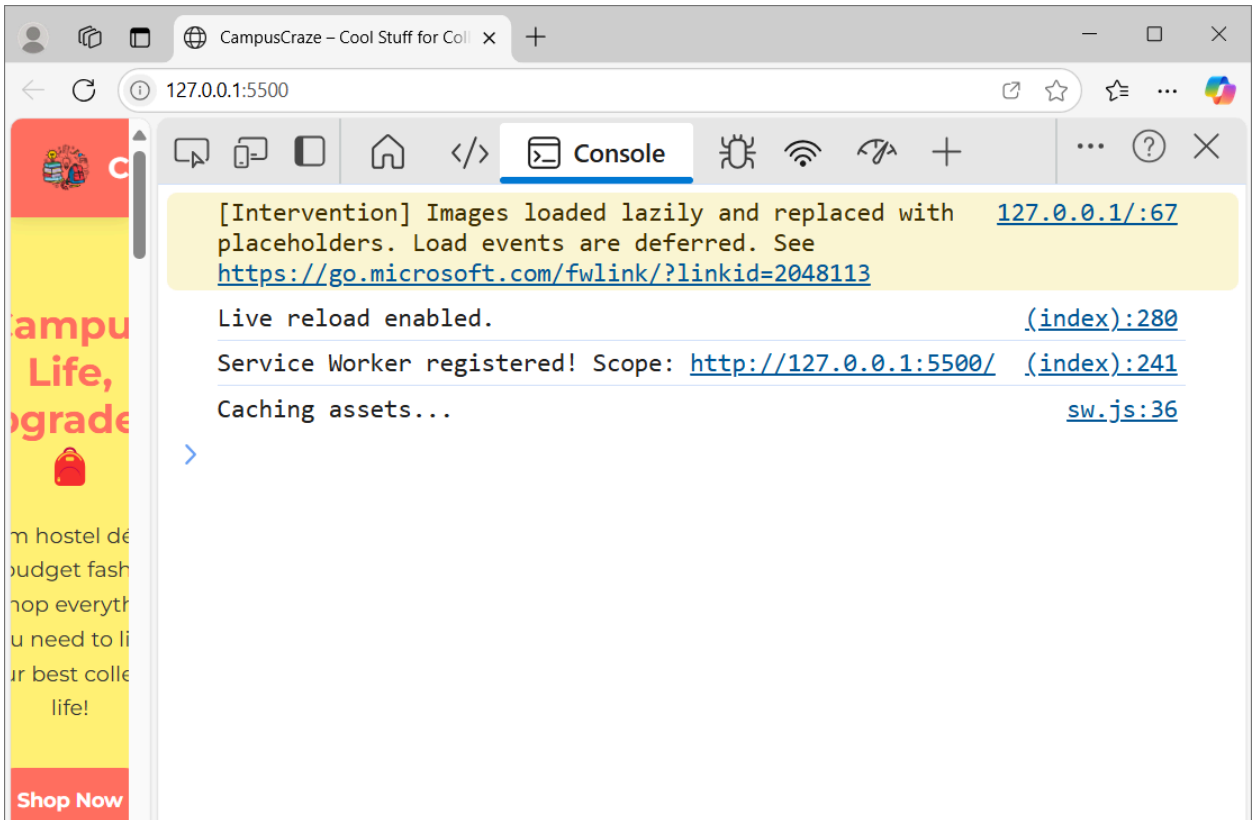
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta name="description" content="CampusCraze - Budget-friendly stationery, hostel decor and fashion for college students">
  <meta name="theme-color" content="#ff6f61">

  <title>CampusCraze – Cool Stuff for College Life</title>
```

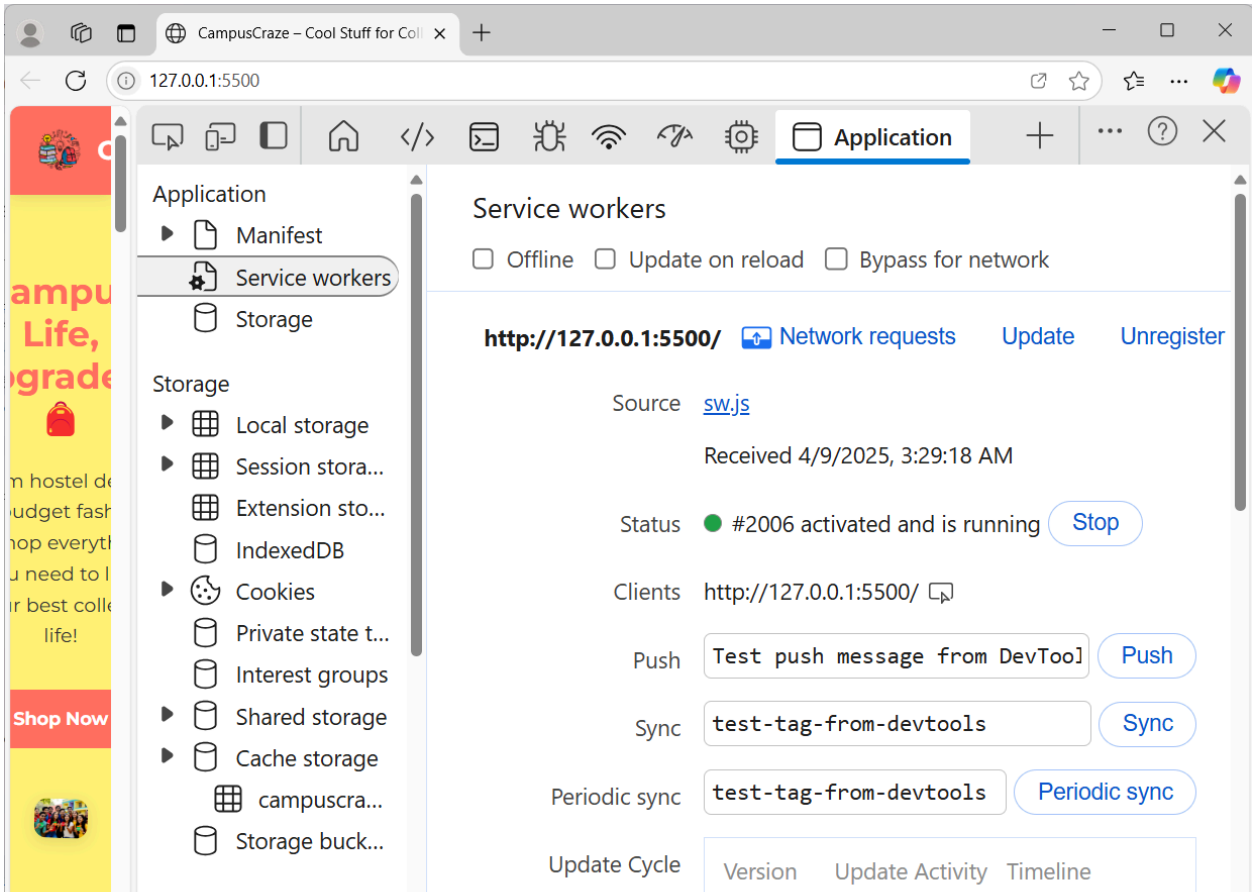
```
  <link rel="stylesheet" href="styles.css">
  <link rel="preconnect" href="https://fonts.googleapis.com">
  <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
  <link
href="https://fonts.googleapis.com/css2?family=Montserrat:wght@400;500;700&display=swap"
rel="stylesheet">

  <!-- PWA requirements -->
  <link rel="manifest" href="manifest.json">
  <link rel="apple-touch-icon" href="icons/icon-192x192.png">

  <!-- For better SEO and sharing -->
  <meta property="og:title" content="CampusCraze – Cool Stuff for College Life">
  <meta property="og:description" content="From hostel décor to budget fashion — shop
everything you need to live your best college life!">
  <meta property="og:image" content="images/og-image.jpg">
</head>
<body>

  <script>
    if ("serviceWorker" in navigator) {
      window.addEventListener("load", () => {
        navigator.serviceWorker
          .register("/sw.js")
          .then((registration) => {
            console.log(
              "Service Worker registered! Scope:",
              registration.scope
            );
          })
          .catch((error) => {
            console.log("Service Worker registration failed:", error);
          });
      });
    }
  </script>
</body>
</html>
```

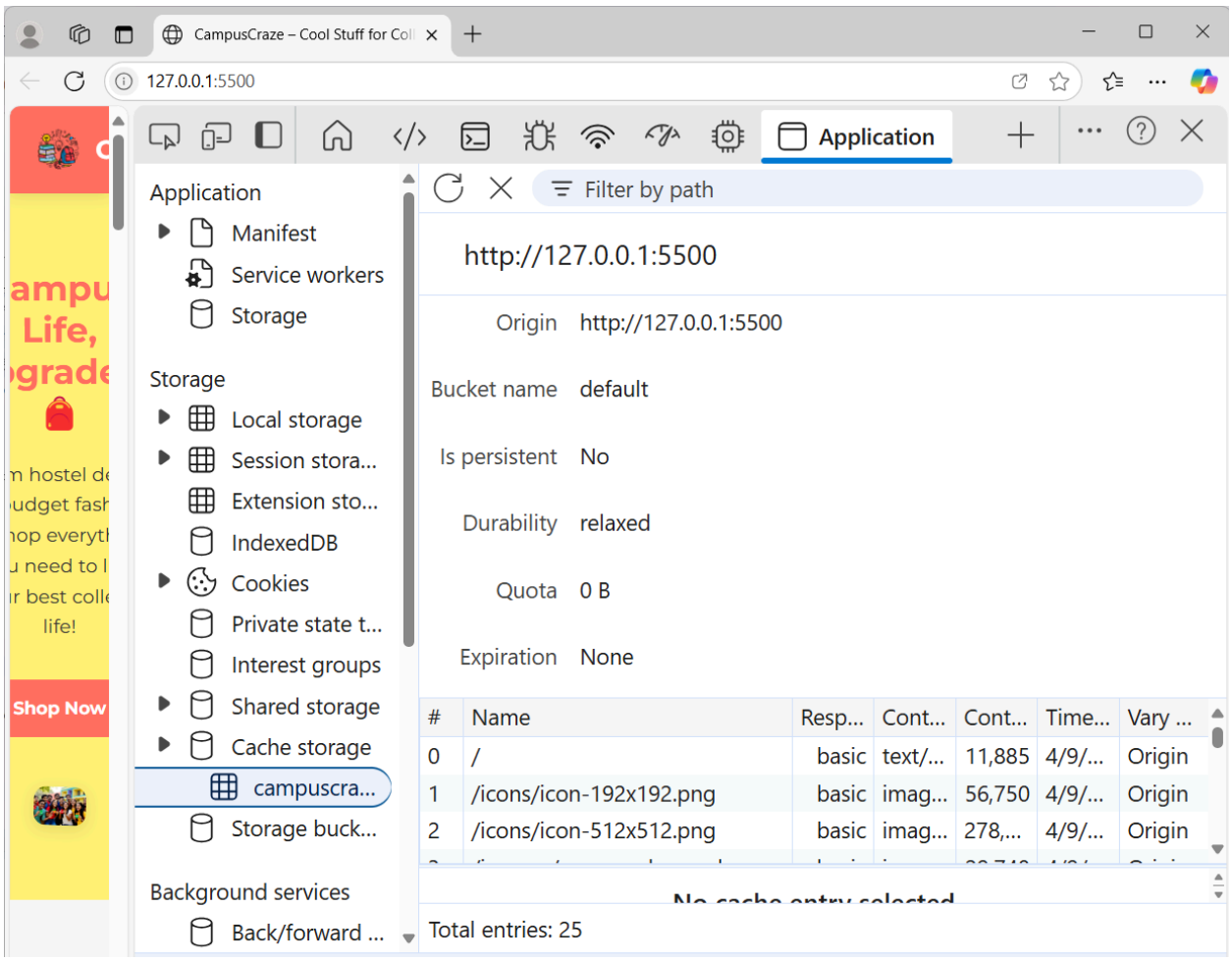
serviceworker.js

```
// CampusCraze Service Worker for PWA functionality
const CACHE_NAME = 'campuscraze-v1';
const ASSETS = [
  '/',
  '/index.html',
  '/styles.css',
  '/main.js',
  '/manifest.json',
  '/images/logo.svg',
  '/images/college-vibes.webp',
  '/images/stationery.webp',
  '/images/decor.webp',
  '/images/fashion.webp',
  '/images/tech.webp',
  '/images/study-bundle.webp',
  '/images/decor-bundle.webp',
  '/images/app-mockup.webp',
  '/images/google-play.svg',
  '/images/app-store.svg',
  '/images/avatar-simran.webp',
  '/images/avatar-rishi.webp',
  '/images/avatar-priya.webp',
  '/images/instagram.svg',
  '/images/youtube.svg',
  '/images/twitter.svg',
  '/icons/icon-192x192.png',
  '/icons/icon-512x512.png',
  'https://fonts.googleapis.com/css2?family=Montserrat:wght@400;500;700&display=swap'
];

// Install event - cache assets
self.addEventListener('install', event => {
  event.waitUntil(
    caches.open(CACHE_NAME)
      .then(cache => {
        console.log('Caching assets...');
        return cache.addAll(ASSETS);
      })
      .then(() => self.skipWaiting())
```

```
  );
});

// Activate event - clean up old caches
self.addEventListener('activate', event => {
  event.waitUntil(
    caches.keys().then(cacheNames => {
      return Promise.all(
        cacheNames.map(cache => {
          if (cache !== CACHE_NAME) {
            console.log('Clearing old cache:', cache);
            return caches.delete(cache);
          }
        })
      );
    })
    .then(() => self.clients.claim())
  );
});
```




CampusCraze – Cool Stuff for Col
127.0.0.1:5500
Application
Application
Manifest
Service workers
Storage
Storage
Local storage
Session stora...
Extension sto...
IndexedDB
Cookies
Private state t...
Interest groups
Shared storage
Cache storage
campuscra...
Storage buck...
Service workers
Offline
Update on reload
Bypass for network
http://127.0.0.1:5500/
Network requests
Update
Unregister
Source
sw.js
Received 4/9/2025, 3:29:18 AM
Status
#2006 activated and is running
Stop
Clients
http://127.0.0.1:5500/
Push
Test push message from DevTool
Push
Sync
test-tag-from-devtools
Sync
Periodic sync
test-tag-from-devtools
Periodic sync
Update Cycle
Version
Update Activity
Timeline


CampusCraze – Cool Stuff for Col
127.0.0.1:5500
Application
Application
Manifest
Service workers
Storage
Storage
Local storage
Session stora...
Extension sto...
IndexedDB
Cookies
Private state t...
Interest groups
Shared storage
Cache storage
campuscra...
Storage buck...
Background services
Back/forward ...
Filter by path
http://127.0.0.1:5500
Origin
http://127.0.0.1:5500
Bucket name
default
Is persistent
No
Durability
relaxed
Quota
0 B
Expiration
None
Name Resp... Cont... Cont... Time... Vary ...
0 / basic text/... 11,885 4/9/... Origin
1 /icons/icon-192x192.png basic imag... 56,750 4/9/... Origin
2 /icons/icon-512x512.png basic imag... 278,... 4/9/... Origin
Total entries: 25

Conclusion

In this experiment, we successfully registered a service worker and implemented the install and activate events to cache assets for offline use in CampusCraze. Initially, the service worker failed to load because the file was named incorrectly (sw.js vs serviceworker.js), which we fixed by renaming the file to match the registered path.