

# **MAD and PWA Lab**

**Name: Anuj Chitari    Class: D15A    Roll no:11**

## **Experiment - 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.

Things to note about Service Worker:

A service worker is a programmable network proxy that lets you control how network requests from your page are handled.

Service workers only run over HTTPS. Because service workers can intercept network requests and modify responses, "man-in-the-middle" attacks could be very bad. The service worker becomes idle when not in use and restarts when it's next needed. You cannot rely on a global state persisting between events. If there is information that you need to persist and reuse across restarts, you can use IndexedDB databases.

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.

What can't we do with Service Workers?

You can't access the Window

You can't access the window, therefore, You can't manipulate DOM elements. But, you can communicate to the window through post Message and manage processes that you want.

You can't work it on 80 Port

Service Worker just can work on HTTPS protocol. But you can work on localhost during development.

Registration

To install a service worker, you need to register it in your main JavaScript code. Registration tells the browser where your service worker is located, and to start installing it in the background. Let's look at an example:

```
if ('serviceWorker' in navigator) { navigator.serviceWorker.register('/service-worker.js')
.then(function(registration) {
console.log('Registration successful, scope is:', registration.scope);
})
.catch(function(error) {
console.log('Service worker registration failed, error:', error);
});
}
```

This code starts by checking for browser support by examining `navigator.serviceWorker`. The service worker is then registered with `navigator.serviceWorker.register`, which returns a promise that resolves when the service worker has been successfully registered. The scope of the service worker is then logged with `registration.scope`. If the service worker is already installed, `navigator.serviceWorker.register` returns the registration object of the currently active service worker.

The scope of the service worker determines which files the service worker controls, in other words, from which path the service worker will intercept requests. The default scope is the location of the service worker file, and extends to all directories below. So if `service-worker.js` is located in the root directory, the service worker will control requests from all files at this domain.

You can also set an arbitrary scope by passing in an additional parameter when registering. For example:

`main.js`

```
navigator.serviceWorker.register('/service-worker.js', { scope: '/app/'
});
```

In this case we are setting the scope of the service worker to `/app/`, which means the service worker will control requests from pages like `/app/`, `/app/lower/` and `/app/lower/lower`, but not from pages like `/app` or `/`, which are higher.

If you want the service worker to control higher pages e.g. `/app` (without the trailing slash) you can indeed change the scope option, but you'll also need to set the `Service-Worker-Allowed` HTTP Header in your server config for the request serving the service worker script.

`main.js`

```
navigator.serviceWorker.register('/app/service-worker.js', { scope: '/app'
});
```

## Installation

Once the browser registers a service worker, installation can be attempted. This occurs if the service worker is considered to be new by the browser, either because the site currently doesn't have a registered service worker, or because there is a byte difference between the new service worker and the previously installed one.

A service worker installation triggers an install event in the installing service worker. We can include an install event listener in the service worker to perform some task when the service worker installs. For instance, during the install, service workers can precache parts of a web app so that it loads instantly the next time a user opens it (see caching the application shell). So, after that first load, you're going to benefit from instant repeat loads and your time to interactivity is going to be even better in those cases. An example of an installation event listener looks like this:

service-worker.js

```
// Listen for install event, set callback
self.addEventListener('install', function(event) {
// Perform some task
});
```

## Activation

Once a service worker has successfully installed, it transitions into the activation stage. If there are any open pages controlled by the previous service worker, the new service worker enters a waiting state. The new service worker only activates when there are no longer any pages loaded that are still using the old service worker. This ensures that only one version of the service worker is running at any given time.

When the new service worker activates, an activate event is triggered in the activating service worker. This event listener is a good place to clean up outdated caches (see the Offline Cookbook for an example).

**Code:**

**index.html**

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>E-commerce PWA</title>
  <link rel="stylesheet" href="main.css">
</head>
<body>
  <h1>Welcome to Our E-commerce Store</h1>
  <!-- E-commerce content goes here -->
  <script src="app.js"></script>
</body>
</html>

```

```

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

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Xplore Skill - Sagar Developer</title>
  <link rel="stylesheet" href="main.css">
  <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/css/all.min.css">
  <link rel="stylesheet"

```

```

href="https://fonts.googleapis.com/css2?family=Poppins:wght@100;200;300;400;600;700&display=swap">
</head>
<!-- Google tag (gtag.js) -->
<script async src="https://www.googletagmanager.com/gtag/js?id=G-F8NS5KXZLN"></script>
<script>
  window.dataLayer = window.dataLayer || [];
  function gtag(){dataLayer.push(arguments);}
  gtag('js', new Date());

  gtag('config', 'G-F8NS5KXZLN');
</script>
<body>
  <script src="app.js"></script>
  <!-- Created BY - Sagar Developer -->
  <section class="header">
    <nav>
      <a href="index.html" class="logo">Xplore
        <i class="fab fa-staylinked"></i>kill
      </a>
      <div class="nav-links" id="navLinks">
        <!-- Reposnive bar open and close -->
        <i class="fa fa-times" onclick="hideMenu()"></i>

```

```

        <ul>
            <li><a href="index.html">Home</a></li>
            <li><a href="course.html">Course</a></li>
            <li><a href="blog.html">Blog</a></li>
            <li><a href="about.html">About</a></li>
            <li><a href="contact.html">Contact</a></li>
        </ul>
    </div>
    <i class="fa fa-bars" onclick="showMenu()"></i>

</nav>

<div class="text_box">
    <h2>GET READY</h2>
    <p id="headtext">TO DISCOVER CAMPUS</p>
    <p>Lorem ipsum dolor sit, amet consectetur adipisicing elit. Velit consequuntur corrupti sapiente
aut porro
        <br> esse blanditiis in quae perspiciatis quo.
    </p>
    <a href="#" class="hero_btn">Visit Us To Know More</a>
</div>
</section>

<!-- Course Section Start -->
<section class="course">
    <div class="PageBlock">
        <div class="verticalLine"></div>
        <div class="Clear"></div>
    </div>
    <h1>EXPLORE OUR 60+ <br> MAJOR PROGRAMS</h1>
    <p>Lorem ipsum dolor, sit amet consectetur adipisicing elit.</p>

    <div class="row">
        <div class="course-col">
            <h3>Undergraduate Programs</h3>
            <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Similique dolor corporis, commodi
nihil quas
                soluta labore quisquam impedi distinctio explicabo aut minima quos pariat unde aliquam
earum
                laborum velit non.</p>
        </div>
        <div class="course-col">
            <h3>Graduate Programs</h3>
            <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Similique dolor corporis, commodi
nihil quas
                soluta labore quisquam impedit distinctio explicabo aut minima quos pariat unde aliquam
earum
                laborum velit non.</p>
        </div>
        <div class="course-col">
            <h3>Adult Learning & Degree Completion</h3>

```

<p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Similique dolor corporis, commodi nihil quas  
soluta labore quisquam impedit distinctio explicabo aut minima quos pariatur unde aliquam  
earum

laborum velit non.</p>

</div>

</div>

</section>

<!-- Course Section End -->

<!-- Campus Section Start-->

<section class="campus">

<div class="PageBlock">

<div class="verticalLine"></div>

<div class="Clear"></div>

</div>

<h1>TAKE OUR VIRTUAL TOUR</h1>

<p>Lorem ipsum dolor, sit amet consectetur adipisicing elit.</p>

<div class="row">

<div class="campus-col">



<div class="layer">

<h3>DELHI</h3>

</div>

</div>

<div class="campus-col">



<div class="layer">

<h3>HYDERABAD</h3>

</div>

</div>

<div class="campus-col">



<div class="layer">

<h3>MUMBAI</h3>

</div>

</div>

</div>

</section>

<!-- Campus Section End -->

<!-- Facilities Section Start -->

<section class="facilities">

<div class="PageBlock">

<div class="verticalLine"></div>

<div class="Clear"></div>

</div>

<h1>Our Facilities</h1>

<p>Lorem ipsum dolor, sit amet consectetur adipisicing elit.</p>

```

<div class="row">
  <div class="facilities-col">
    
    <h3>Best Library</h3>
    <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Distinctio omnis asperiores atque
aperiam.
    </p>
  </div>
  <div class="facilities-col">
    
    <h3>Athletics</h3>
    <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Distinctio omnis asperiores atque
aperiam.
    </p>
  </div>
  <div class="facilities-col">
    
    <h3>Tasty and Healthy Food</h3>
    <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Distinctio omnis asperiores atque
aperiam.
    </p>
  </div>
</div>
</section>
<!-- Facilities Section End -->

<!-- Testimonials Section Start -->
<section class="testimonials">
  <div class="PageBlock">
    <div class="verticalLine"></div>
    <div class="Clear"></div>
  </div>
  <h1>What Our Student Says</h1>
  <p>Lorem ipsum dolor, sit amet consectetur adipisicing elit.</p>

  <div class="row">
    <div class="testimonials-col">
      
      <div>
        <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Modi maiores in nostrum rerum
voluptatem
        praesentium veritatis alias omnis voluptate nisi aperiam, voluptatum quibusdam itaque a
deserunt. In quia repellat maxime.</p>
        <h3>Student Name</h3>
        <i class="fa fa-star"></i>
        <i class="fa fa-star"></i>
        <i class="fa fa-star"></i>
        <i class="fa fa-star"></i>
        <i class="fa fa-star"></i>
      </div>
    </div>
  </div>

```



```

</div>
<div class="testimonials-col">
  
  <div>
    <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Modi maiores in nostrum rerum
voluptatem
    praesentium veritatis alias omnis voluptate nisi aperiam, voluptatum quibusdam itaque a
    deserunt. In quia repellat maxime.</p>
    <h3>Student Name</h3>
    <i class="fa fa-star"></i>
    <i class="fa fa-star"></i>
    <i class="fa fa-star"></i>
    <i class="fa fa-star"></i>
    <i class="fa fa-star-half-alt"></i>
  </div>
</div>
</div>
</div>
</section>
<!-- Testimonials Section End -->

<!-- Call To Action Section Start -->
<section class="cta">
  <h1>GET READY FOR A BRIGHT FUTURE</h1>
  <a href="contact.html" class="hero_btn">CONTACT US</a>
</section>
<!-- Call To Action Section End -->

<!-- Footer Section Start -->
<section class="footer">
  <hr>
  <h4>About Us</h4>
  <p>Lorem, ipsum dolor sit amet consectetur adipisicing elit. Esse minima incidunt odio nam facilis,
eligendi
  Itaque fugiat cupiditate consectetur. Aliquid ab deserunt exercitationem, illum molestiae dolorem.
  </p>

  <div class="icons">
    <i class="fab fa-facebook-f"></i>
    <i class="fab fa-instagram"></i>
    <i class="fab fa-twitter"></i>
    <i class="fab fa-linkedin"></i>
  </div>
  <p>Made with <i class="fas fa-heart"></i> by <a href="index.html">Sagar Developer</a></p>
  <p>Copyright © 2021 <a href="index.html">Xplore Skill</a>. All Rights Reserved</p>
</section>
<!-- Footer Section End -->

<script src="script.js"></script>
</body>

</html>

```

## main.css

```
html, body {  
  margin: 0;  
  font-family: arial,sans-serif;  
  min-width: 900px;  
  line-height: 14px;  
  font-size: 14px;  
}  
  
* {  
  box-sizing: border-box;  
}  
  
a {  
  color: #0066c0;  
}  
  
a:hover {  
  color: #c45500;  
}  
  
#banner {  
  background: #F6F6F6 url('../img/banner.jpg') no-repeat top left;  
  height: 55px;  
  background-size: 1920px;  
  min-width: 1000px;  
  display: block;  
}  
  
header {  
  background-color: #232f3e;  
  height: 99px;  
}
```

## app.js

```
if ('serviceWorker' in navigator) {  
  window.addEventListener('load', () => {  
    navigator.serviceWorker.register('/service-worker.js')  
      .then(registration => {  
        console.log('Service Worker registered with scope:', registration.scope);  
      })  
      .catch(error => {  
        console.error('Service Worker registration failed:', error);  
      });  
  });  
}
```

## **service-worker.js**

```
// service-worker.js
```

```
const cacheName = 'ecommerce-pwa-v1';
```

```
const assetsToCache = [
```

```
  '/',
```

```
  '/index.html',
```

```
  '/main.css',
```

```
  '/app.js'
```

```
  // Add more files and assets here as needed
```

```
];
```

```
self.addEventListener('install', event => {
```

```
  event.waitUntil(
```

```
    caches.open(cacheName)
```

```
    .then(cache => {
```

```
      return cache.addAll(assetsToCache);
```

```

    })

    );

});

self.addEventListener('activate', event => {

    event.waitUntil(

        caches.keys().then(cacheNames => {

            return Promise.all(

                cacheNames.filter(name => {

                    return name !== cacheName;

                }).map(name => {

                    return caches.delete(name);

                })

            );

        })

    );

```

### **Steps for Execution**

**Create a folder and put all 4 files main.css , service-worker.js, app.js, index.html**  
**open visual studio**  
**install extension Live server**  
**open folder in visual studio open index.html**  
**on bottom right corner click go Live**  
**it will open html page in browser**  
**go to developer tools**

## Output:

The first screenshot shows the 'Application' tab in Chrome DevTools. The 'Storage' section is expanded, showing 'eCommerce-pwa-v1' with a list of files: /, /app.js, /index.html, and /main.css. The 'Background services' section is also visible, showing 'Back/forward cache', 'Background fetch', 'Background sync', 'Bounce tracking mitigations', 'Notifications', 'Payment handler', 'Periodic background sync', and 'Speculative loads'.

The second screenshot shows the 'Service workers' tab in Chrome DevTools. It displays the registration details for the service worker 'http://127.0.0.1:5500/'. The status is '#93 activated and is running stop'. The 'Update Cycle' section shows a timeline with three events: 'Install' (Version #93, Update Activity), 'Wait' (Version #93, Update Activity), and 'Activate' (Version #93, Update Activity).

## Conclusion:

In this experiment, we have registered a service worker, and completed the install and activation process for a new service worker for the E-commerce PWA.