

Ingeniería en Desarrollo y Gestión de Software

TEMA: Introducción al Service Worker
PROFESOR: Parra Galaviz Ray Brunnet
MATERIA: PWA

ELABORADO POR: García González Christian Andrés
FECHA: 28 de febrero de 2024

1. Service Worker

Los Service workers actúan esencialmente como proxy servers asentados entre las aplicaciones web, el navegador y la red (cuando está accesible). Están destinados, entre otras cosas, a permitir la creación de experiencias offline efectivas, interceptando peticiones de red y realizando la acción apropiada si la conexión de red está disponible y hay disponibles contenidos actualizados en el servidor. También permitirán el acceso a notificaciones tipo push y APIs de sincronización en segundo plano. Un service worker es una secuencia de comandos ejecutados por el navegador en segundo plano. Se trata de un fichero JavaScript que continúa ejecutándose aunque el sitio web esté cerrado. Estas son las características más destacadas de un service worker:

- No puede acceder directamente al DOM, sino que se comunica con las páginas que controla mediante la interfaz PostMessage.
- Al ser un proxy de red programable, permite controlar el modo en que se manejan las solicitudes de red de la página.
- Son capaces de mantener información mediante la API de IndexedDB.
- Pueden implementar diferentes sistemas de cacheo.

2. ¿Qué necesito para usar service workers?

- Soporte del navegador. Para poder hacer uso del service worker necesitamos un navegador compatible. Hoy en día, la gran mayoría de navegadores ya lo han implementado (como Chrome o Firefox) o están en proceso para conseguirlo (como Safari o Edge).
- HTTPS. Durante la fase de desarrollo, se puede utilizar service worker a través del “localhost”, ya que se considera un servidor seguro, aunque no lo sea. Sin embargo, no se considera una forma segura de entrar al

servidor mediante la IP de un servidor, o la IP de loop. Es importante tener en cuenta que para poder implementar el service worker, necesitaremos configurar HTTPS en el servidor.

3. Conceptos y uso de service worker

Un service worker es un worker manejado por eventos registrado para una fuente y una ruta. Consiste en un fichero JavaScript que controla la página web (o el sitio) con el que está asociado, interceptando y modificando la navegación y las peticiones de recursos, y cacheando los recursos de manera muy granular para ofrecer un control completo sobre cómo la aplicación debe comportarse en ciertas situaciones (la más obvia es cuando la red no está disponible). Un service worker se ejecuta en un contexto worker: por lo tanto no tiene acceso al DOM, y se ejecuta en un hilo distinto al JavaScript principal de la aplicación, de manera que no es bloqueante. Está diseñado para ser completamente asíncrono, por lo que APIs como el XHR asíncrono y localStorage no se pueden usar dentro de un service worker. Los service workers solo funcionan sobre HTTPS, por razones de seguridad. Modificar las peticiones de red en abierto permitiría ataques man in the middle realmente peligrosos. En Firefox, las APIs de service worker se ocultan y no pueden ser empleadas cuando el usuario está en modo de navegación en privado.

4. Arquitectura básica

1. La URL del service worker se obtiene y registra a través de `serviceWorkerContainer.register()`.
2. Si tiene éxito, el service worker se ejecuta en `ServiceWorkerGlobalScope`; esto es básicamente un tipo especial de contexto de trabajo, que se ejecuta fuera del hilo principal de ejecución del script, sin acceso al DOM.
3. El service worker ahora está listo para procesar eventos.
4. Se intenta la instalación del worker cuando se accede posteriormente a las páginas controladas por el service worker. Un evento de instalación siempre es el primero que se envía a un service worker (esto se puede usar para iniciar el proceso de completar una IndexedDB «base de datos indexada» y almacenar en caché los activos del sitio). Este es realmente el mismo tipo de procedimiento que instalar una aplicación nativa o Firefox OS: hace que todo esté disponible para usar sin conexión.
5. Cuando se completa el controlador oninstall, se considera que el service worker está instalado.
6. Lo siguiente es la activación. Cuando se instala el service worker, recibe un evento de activación. El uso principal de onactivate es para la limpieza de los recursos utilizados en versiones anteriores de un script del service worker.

7. El service worker ahora controlará las páginas, pero solo aquellas que se abran después de que `register()` tenga éxito. En otras palabras, los documentos se deberán volver a cargar para controlarlos realmente, porque un documento comienza con o sin un service worker y lo mantiene durante toda su vida.

5. Ciclo de vida

El ciclo de vida del service worker es totalmente independiente de la página web y su fin es conseguir la mejor experiencia de usuario, así que cuenta con diferentes etapas, como vemos en el gráfico y explicamos a continuación con más detalle centrándonos en las más importantes.

- **Instalación:** Para poder instalar un service worker debemos registrarlo, en el lenguaje JavaScript de la página. Una vez registrado, el navegador comienza la etapa de instalación en segundo plano. La instalación solamente se realiza una vez por service worker y para que sea correcta los archivos deben almacenarse adecuadamente en caché.
- **Activación:** Cuando logramos que el service worker se instale correctamente, podrá controlar el cliente y pasará al estado *Activate* o activo, permitiéndonos manejar eventos.
- **Terminated y Fetch:** Tras la activación, el service worker controlará todas las páginas posibles, pudiendo aparecer dos estados diferentes: *Terminated* o modo de ahorro de memoria y *Fetch*, que indica que está manejando las peticiones de red.

6. Referencias

1. Service Worker API - Referencia de la API Web — MDN. (2023, July 24). MDN Web Docs. https://developer.mozilla.org/es/docs/Web/API/Service_Worker_API
2. Usar Service Workers - Referencia de la API Web — MDN. (2023, July 24). MDN Web Docs. https://developer.mozilla.org/es/docs/Web/API/Service_Worker_API/Using_Service_Workers
3. De Zúñiga, F. G. (2023, February 20). Service workers, servicios web más allá del navegador. Blog De arsys.es. <https://www.arsys.es/blog/service-worker>