

## Parcial 2: Trabajo práctico domiciliario

### Consignas para la entrega del parcial:

- Para la entrega del presente parcial, la carpeta archivos debe ser comprimida en un único archivo de extensión .ZIP o .RAR, respetando el siguiente formato para asignarle el nombre de archivo:
  - **pwa-parcial-2-dwn3ap-[apellido]-[nombre].[zip o rar]**
- TODO EN MINÚSCULAS Y UTILIZANDO EL GUIÓN MEDIO COMO SEPARADOR.
- Ejemplos:
  - **pwa-parcial-2-dwn3ap-marcosgalban-camila.zip**
- ***En caso de incumplir con el formato del nombre, se restarán automáticamente puntos, pudiendo desaprobar por este error el parcial.***
- La entrega será mediante **Classroom**, en la actividad correspondiente que genere el docente.
- Fecha de entrega:
  - La fecha de entrega será la que indique el docente en la actividad del **Classroom**.

## Consigna del trabajo

Realizar una aplicación web, que contenga una interfaz HTML5, CSS, Imágenes, JavaScript. Esta aplicación debe registrar un Service Worker, el cual debe realizar un precaching de los elementos esenciales para la carga inicial de la aplicación. También debe contener un manifiesto con las propiedades correspondientes.

La aplicación debería tener una landing page, una navegación principal y opcionalmente una secundaria.

En la landing page se deberá dar información sobre de qué trata el sitio.

Una vista con listado de artículos (distinta de la vista anterior) y la posibilidad de realizar algún tipo de interacción ya sea de compra, selección, valoración o carga de comentarios (Esto nos permitirá realizar funciones de persistencia de datos y envío de datos al servidor.).

Vista de detalle de los productos.

Alguna funcionalidad del tipo filtrado, buscador, ordenamiento...

Consumo de datos de datos del servidor, se puede simular con archivos json locales, o realizar consumo de datos de APIs externas.

El usuario debe tener la posibilidad de instalar la pwa.

Se debe incluir notificaciones (Preguntarle al usuario si acepta recibir notificaciones e implementar al menos 1 notificación)

Implementar posibilidad de compartir contenidos.

Detectar el estado de la conexión y realizar algún cambio estético (Offline / Online).

Para el manejo de la persistencia de datos utilizar una de las siguientes:

- LocalStorage - Almacenamiento de datos como string (json).
- IndexedDB - Almacenamiento de datos como objetos.

Su diseño se debe adaptar a los diferentes dispositivos (Responsive).

Se deberá poder acceder al sitio desde internet.

## Consideraciones finales

- Complejidad de la tarea realizada.
- Correcto diseño de la base de datos o JSON.
- Consumo de APIs Externas
- Uso correcto de los verbos de HTTP para cada petición.
- Coherencia en los nombres de variables, clases, métodos, etc.
- **Uso correcto de las etiquetas semánticas de HTML.**
- Documentación correcta del código usando comentarios.
- Suman
  - Estilización del sitio.
  - Validación de los datos.
  - Prolijidad en el código.
  - Prolijidad en la organización de la carpeta del proyecto.