

Consigna Final: Aplicaciones web progresivas

Consignas para la entrega del parcial:

- Para la entrega del presente final, debe hacerse el desarrollo y entrega sobre un repositorio de github.
 - **pwa-final-dwm3bv-[apellido]-[nombre]**
 - TODO EN MINÚSCULAS Y UTILIZANDO EL GUIÓN MEDIO COMO SEPARADOR. ○
 - Ejemplos:
 - **pwa-final-dwm3bv-medina-sergio**
 - ***En caso de incumplir con el formato del nombre, se restará automáticamente dos puntos, pudiendo desaprobar por este error el parcial.***
- La entrega será mediante **DV Panel**.
- Fecha de entrega:
 - La fecha de entrega será hasta 24 hs de antelación a la fecha y hora del final.

Consigna del trabajo

Realizar una aplicación web que funciona completamente offline.

Obligatorio para llegar al 4 (verificar uno a uno):

- ☐ interfaz amigable con el usuario, con uso de Service Worker
- ☐ precaching todos los elementos necesarios para la carga inicial
- ☐ incluir manifiesto correctamente
- ☐ navegación principal (home)
- ☐ navegación secundaria (subpágina)
- ☐ listado de artículos
- ☐ vista ampliada o detalle de artículos (en subpágina o bien en modal)
- ☐ uso de persistencia (localStorage) que se vea reflejado en la interacción
- ☐ uso de async/await por lo menos una vez

- ☐ debe haber algún formulario que esté validado.
- ☐ al menos un consumo de datos a servidor por GET usando api externa, o api php propia (no es válido consumir un json pero se puede leer el json devolverlo desde el backend).
- ☐ al menos un envío de datos a servidor por POST (por ejemplo api php)
- ☐ botón de instalación (que se oculte post instalación)
- ☐ interacción funcional (si el envío o recepción HTTP falla debería saberlo el usuario, por ejemplo mostrando un mensaje)
- ☐ Uso combinado de promise y asyc-await
- ☐ Validación de datos en el back (con fines de testeo).
- ☐ uso de firebase auth

Para llegar al 10:

- Validación de datos del formulario (front) con el back (los datos son válidos para el envío, pero en la práctica son inválidos, por ejemplo un nombre que incluya letras o caracteres especiales como el @) .
- Debe haber interacciones que vinculen datos en la base de datos firebase firestore (por ejemplo, que se pueda actualizar la información del usuario, se pueda agregar datos a la base de datos, etc)
- Uso de algún tipo de convención para los commit empleados. Sugerido: [conventional commits](#)
- Uso apropiado de convenciones de nomenclatura js
- Uso de convenciones para estilos.
- Configuración de notificaciones usando fcm
- Se evaluarán las buenas prácticas en el desarrollo.

(sigue en la siguiente hoja)

Instancia de final:

- Aun cuando se pueda haber desarrollado en equipos, cada integrante debe estar en la mesa evaluadora, pudiendo presentarse individualmente.
- Restricciones de entrega:
 - Los cambios deben ser aplicados directamente sobre el repositorio
 - Debe haber cambios significativos en github hecho por quien presenta el final.
- El envío de la resolución del final es condición necesaria, pero no suficiente para aprobación, pudiendo desaprobar en caso de (por ejemplo):
 - No presentarse a la mesa evaluadora
 - No alcanzar el **Desarrollo mínimo y obligatorio** esperado en la entrega
 - No poder responder preguntas sobre la autoría de la aplicación.
- Se realizará un coloquio donde se presentará lo desarrollado.
 - Se podrá exigir cámara y micrófono encendido
 - Se podrá consultar sobre el desarrollo en su totalidad
- Se recomienda el uso apropiado de github, al resultar beneficioso para versionar los avances. El uso de commit bien distribuidos en tareas y tiempo permiten evidenciar el proceso de desarrollo y sirviendo en gran medida para evacuar dudas razonables sobre la autoría y avance de la aplicación.