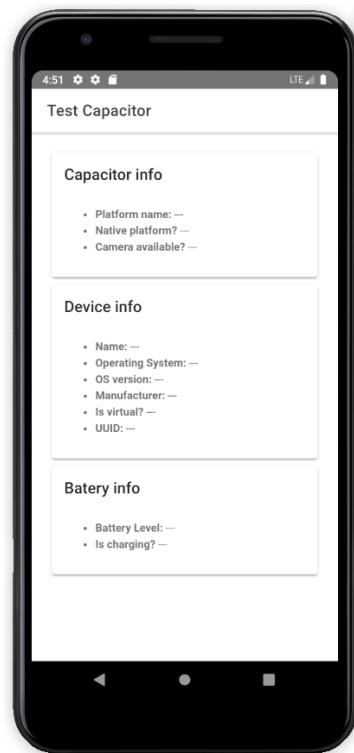


Ejercicios

Ejercicio 1 – Crear aplicación y probarla (2 puntos)

Crea una **nueva** aplicación de Ionic usando Angular con la plantilla “*blank*” y llámala “**TestCapacitor**”. Esta aplicación contendrá una única pantalla (puedes utilizar la página “*home*” creada por defecto) con el siguiente aspecto:



Para completar el ejercicio realiza los siguientes pasos:

- Completa la apariencia de la aplicación según la captura anterior. Puedes utilizar las tarjetas de Ionic e insertar listas dentro de los bloques de contenido.
- Prueba la aplicación en el navegador (con la opción “*serve*” de Ionic).
- Integra Capacitor y añade la plataforma “*android*”.
- Compila y prueba esta aplicación en un emulador Android o en un dispositivo real.
- Pulsa el botón “*back*”, ¿puedes salir de la aplicación? Modifica el código para que al pulsar este botón se cierre la aplicación.

Ejercicio 2 – Uso de API nativa (6 puntos + 2 puntos parte opcional)

En este segundo ejercicio vamos a continuar con la aplicación anterior y, haciendo uso de la API de Capacitor, completaremos la información de las tarjetas a partir de información nativa obtenida del dispositivo móvil. A continuación se incluye una captura con la apariencia final de la aplicación:

Test Capacitor

Capacitor info

- Platform name: android
- Native platform? Yes
- Camera available? No

Device info

- Name: AOSP on IA Emulator
- Operating System: android
- OS version: 9
- Manufacturer: Google
- Is virtual? Yes
- UUID: 649e196417c5ba35

Batery info

- Battery Level: 100%
- Is charging? No

Network info

- Is connected? Yes
- Connection Type: cellular

Events log

- onPause
- onStop
- onStart
- onResume
- Cambio tipo conexión a wifi
- Cambio tipo conexión a celular

Para completar la aplicación se pide:

- Para la tarjeta “*Capacitor info*” haz uso del *plugin* “Capacitor” y completa la información mostrada.
- Las tarjetas “*Device info*” y “*Battery info*” la puedes completar usando el *plugin* “Device”.
- Añade dos nuevas tarjetas: “*Network info*” y “*Events logs*”.
- Para la tarjeta “*Network info*” necesitarás el *plugin* “Network” y hacer uso de su método “getStatus”.
- Por último, en la tarjeta “*Events logs*” se tienen que ir añadiendo los eventos sucedidos: cuando se salga o entre de la aplicación y cuando se cambie el tipo de conexión.
 - Para los eventos de conexión usa el *plugin* “Network” capturando el evento “networkStatusChange” y añade una línea al *log* cada vez que cambie la conexión (por ejemplo: “Cambio tipo conexión a wifi”).
 - Para determinar si se entra o sale de la aplicación usa el *plugin* “App” y captura los eventos “resume”, “pause” y “appStateChange”, añadiendo al log los textos “onResume”, “onPause”, para los dos primeros eventos, y “onStart” o “onStop” para el evento “appStateChange”, en función del parámetro “isActive” que devuelve este evento.

Como parte **opcional** se pide incluir información de un *plugin* adicional, como, por ejemplo, del acelerómetro o del GPS.

Nota: Recuerda que la respuesta de los eventos de Capacitor tiene que gestionarse dentro de una NgZone para que Angular detecte el cambio.