

# Práctica Fundamentos de Android



## Enunciado

El **diseño de la aplicación es libre**, iconos, tamaños, colores, forma de representar los distintos elementos, como seleccionar personajes, cuando y como hacer las transiciones entre ventanas, etc. Siempre que cumpla con las siguientes condiciones.

Vamos a realizar un simulador de batallas de Dragon Ball utilizando la API del Bootcamp.

Ventana 1: Primero el usuario deberá loguearse con un usuario y contraseña.

Ventana 2: Tras el login deberán aparecer todos los personajes de Dragon Ball, permitiendo escoger uno de ellos. Deberás realizar una request para obtenerlos. Además de lo que te mande el servicio, tienes que añadir a todos los personajes estos nuevos atributos: "Vida máxima" y "Vida actual", ambos a valor 100 al inicio. Un personaje que tenga 0 puntos de vida no podrá ser seleccionado.

Ventana 3: Debe aparecer el personaje seleccionado por el usuario junto con una barra, un botón de "curarse" y otro de "recibir daño". La barra de vida debe mostrar sus puntos de vida actuales. En caso de pulsar el botón "recibir daño" debe recibir reducirse la vida entre 10 y 60 puntos. En caso de pulsar el botón "curarse" debe incrementarse la vida 20 puntos.

- Si un personaje tiene 0 puntos de vida, debes volver a la Ventana 2. Ese personaje deberá aparecer en la ventana con algún signo que indique que está herido.

## Requisitos Obligatorios

- La Ventana 1 es una **actividad**
- La Ventana 2 y 3 son **fragmentos** que comparten una **segunda actividad**
- La arquitectura aplicada es **MVWV**
- **Diseño coherente y eficaz.** Que sea diseño libre no significa que no vaya a evaluarse
- Debes tener 2 clases de personajes: El "PersonajeDTO", representando el Json que has descargado del servidor y el "Personaje" propio de tu app (donde le añades sus puntos de vida). Utiliza la función map para transformar de PersonajeDTO a Personaje y hacerle las modificaciones

## Requisitos Opcionales

- Añade un botón (u otro elemento pulsable) a la Ventana 2. Si el usuario lo pulsa, todos los puntos de vida de todos los personajes vuelven a 100. Por tanto, todos los personajes de la lista vuelven a estar habilitados
- Añade persistencia al juego. Si el usuario cierra de la aplicación, todos los personajes deben conservar sus puntos de vida.
- Añade un botón (u otro elemento pulsable) a la Ventana 3. Al pulsarlo, debe mostrarte cuantas veces ha sido seleccionado por el usuario ese personaje mediante un mensaje en pantalla estilo Toast.