

Descripción

El reto consiste en crear una aplicación sencilla en react native, que tenga las siguientes funcionalidades:

- Una sección de “tasks”, que es un listado sencillo de tasks agregados por el usuario (solo tendrán descripción), y que dé la opción de agregar más tasks.
- Una sección de “listado”, que simplemente hará un request a una lista de datos remoto y les hará render en la pantalla

Debe tener una pantalla principal que contenga 2 botones, que se usarán para navegar a las pantallas de cada una de las secciones mencionadas.

Anotaciones generales:

- Todas las funcionalidades descritas deben llevar unit tests (transacciones en pantallas, redux)
- Debe ser desarrollado en typescript.

Sección de “tasks”

- Se debe crear un listado de “tasks” en un contexto de redux.
- La pantalla deberá leer la data presente en redux para mostrar la lista, y deberá agregar los datos a redux cuando el usuario intente crear un task nuevo.
- La pantalla debe contar con la lista de tasks, y con un botón de “agregar nuevo task”
 - El botón debe mostrar un modal con un formulario para crear el task nuevo
 - No debe permitir crear el task si el campo está vacío.

Nota: como la data se está guardando en redux, debemos poder ser capaces de navegar a la pantalla principal de vuelta, y al regresar a la pantalla de “tasks”, se deben mostrar los tasks que agregué anteriormente.

Sección de “listado”

El objetivo de esta sección es mostrar una lista que provenga de una fuente remota de datos. Para esta prueba debe utilizar esta fuente de datos: <https://6172cfe5110a740017222e2b.mockapi.io/elements>

- Se debe mostrar un listado con los elementos regresados por el endpoint proporcionado, que muestre por lo menos el nombre.
 - Si quiere, puede mostrar la imagen a la derecha estilo “lista de contactos”
- Debe hacer la petición al endpoint apenas entre a la pantalla
- Debe mostrar un layout de “loading” mientras se está cargando la data

Una vez completado, compartir el repositorio para poder revisar el código y para poder correr la aplicación de forma local (solo la probaremos en android)

