



Desafío Técnico – React Native (Expo + TypeScript)

Objetivo

Construir una aplicación móvil sencilla de **hábitos diarios**, que permita marcarlos como completados y visualizar el progreso semanal en un gráfico.



Requerimientos

1. Pantalla principal – Lista de hábitos

- Mostrar una lista de hábitos (pueden estar hardcodeados, cargados desde un mock API o permitir agregar).
- Cada hábito se puede marcar como **completado/no completado** para el día actual.

2. Animaciones

- Al marcar un hábito como completado debe mostrarse una **animación de feedback** (ej: check que hace “bounce”, ícono que escala, etc.).
- Al desmarcar, otra animación sutil (ej: fade).

3. Estado

- Manejo de estado complejo:
 - Hábitos disponibles.
 - Progreso diario (qué hábitos se completaron en cada día).
- Persistencia local para que los datos no se pierdan al cerrar la app (ej: AsyncStorage).

4. Pantalla de estadísticas

- Mostrar un **gráfico semanal** con la cantidad de hábitos completados por día.
- Podés usar cualquier librería de gráficos (ej: `react-native-chart-kit`, `victory-native`, etc.).

5. Opcional (suma puntos)

- Cargar hábitos iniciales desde una API (ej: Fake API, JSONPlaceholder, MockAPI).
 - Testing con Jest y/o `@testing-library/react-native`
-

Requisitos técnicos

- **Expo + TypeScript** obligatorio.
 - Manejo de **estado complejo** (puede ser con Context, Zustand, Redux o `useReducer`).
 - Uso de **al menos una animación clara** (`react-native-reanimated`, `moti` o Animated API nativa).
 - UI **mínimamente cuidada** (colores consistentes, tipografía clara, padding, que se vea usable).
 - Código legible, modular y con tipado correcto.
-

Tiempo

- Se estima que el desafío debería llevar entre **1.5 y 2 horas**.
 - **Se espera y se recomienda el uso de inteligencia artificial como apoyo** (para boilerplate, dudas puntuales, ejemplos de código, etc.).
 - Si al llegar a las 2 horas no terminaste todas las features, **entregá lo que tengas hecho** para no gastar más tiempo.
-

Qué vamos a evaluar

- Correcto manejo del **estado** (hábitos + progreso).
 - Uso de **TypeScript** (tipos claros para hábitos, progreso, etc.).
 - Animación fluida y bien integrada.
 - Calidad del código y separación de componentes.
 - Diseño básico y funcional.
 - Creatividad y extras opcionales.
-

Entregables

1. **Grabación de pantalla** mostrando el resultado final de la app.
2. **Código fuente** publicado en un repositorio público de GitHub.
3. **Link o QR de Expo Go** para probar la app en un dispositivo real.