

# Prueba Técnica – Finova

## Parte 1 – Preguntas escritas a mano

Responde estas preguntas a mano en una hoja, toma una foto o captura y envíala.

### ***Generales***

1. ¿Cómo aprendes un nuevo framework, proyecto o lenguaje de programación?
2. ¿Cómo encuentras o buscas un bug?
3. ¿Cómo afrontas los desacuerdos en un grupo aun sabiendo que tienes la razón?

### ***Frontend***

4. ¿Conoces buenas prácticas de desarrollo del front-end?
5. ¿Para qué sirve un useEffect y un useRef en React?

### ***Backend***

6. ¿Cómo diseñarías una API RESTful?
7. ¿Cómo se mejora el desempeño de una base de datos?
8. ¿En qué se diferencia SQL de NoSQL y de NewSQL?
9. ¿Sabes qué es DevOps?
10. ¿Sabes qué es polimorfismo?

## Parte 2 – Ejercicio práctico

El ejercicio consiste en crear un backend y un frontend simples:

***Backend (Node.js preferiblemente con TypeScript)***

- Crea un servidor que consulte una API externa.
- Puedes elegir entre:
  - Thrones API: <https://thronesapi.com/>
  - PokéAPI: <https://pokeapi.co/>
  - Rick and Morty API: <https://rickandmortyapi.com/>

## ***Frontend***

- El frontend debe consumir tu propia API (la que creaste en el backend).
- El objetivo es mostrar una card de personaje o personajes obtenidos desde el backend.
- Esto nos dará una idea de si sabes:
  - Mapear correctamente las respuestas del backend.
  - Conectar frontend con backend, que es lo que más nos interesa.

## **Entregables en GitHub**

- Sube tu código a GitHub con el commit: "prueba técnica".
- Puedes crear un repositorio para el backend y otro para el frontend (opcional pero recomendado).
- El objetivo de esto es confirmar que sabes lo básico de Git/GitHub y que podamos revisar tu código.

## **Lo que valoramos**

- Proceso lógico: cómo llegas a la solución.
- Escalabilidad y modularización: si tu diseño se puede ampliar y mantener en el futuro.
- Consistencia en el código: claridad, orden y buenas prácticas.
- Responsabilidad: entender lo que construyes con o sin ayuda de IA.

## **Nota importante**

En Finova, este tipo de trabajo hace parte del día a día como programadores.

Aunque hoy en día la IA nos ayuda mucho a programar, el objetivo es que seas responsable y entiendas el código que generas, con o sin IA.

Por eso:

- Te preguntaremos periódicamente cómo vas con la prueba.
- Puedes preguntarnos cualquier duda que tengas durante el proceso.
- Lo más importante no es solo entregar el código, sino mostrar consistencia y claridad en tu trabajo.