

Prueba Técnica Full-Stack

Contexto

Eres un **Desarrollador Fullstack** en **Inlaze**, una compañía de marketing de afiliados especializada en apuestas deportivas en Colombia. Tu responsabilidad principal es garantizar el desarrollo y mantenimiento eficiente de las plataformas web de Inlaze, asegurando el cumplimiento de los estándares de diseño, rendimiento y usabilidad.

Objetivo de la prueba

Evaluar tus habilidades en el diseño e implementación de un **sistema de gestión de tareas** utilizando **NestJS** para el backend y **Next.js** para el frontend, aplicando una arquitectura de microservicios. Esta prueba comprueba tu capacidad para integrar ambas tecnologías y seguir buenas prácticas de desarrollo.

1. Descripción del Proyecto

Desarrolla una aplicación web para la gestión de tareas que permita a los usuarios colaborar eficientemente.

2. Requisitos Funcionales

Gestión de Proyectos

- Crear nuevos proyectos y asignar tareas a esos proyectos.
- Ver una lista de proyectos existentes.
- Acceder a la página de detalles de cada proyecto para ver sus tareas asociadas.

Gestión de Tareas

- Crear tareas dentro de un proyecto, especificando título, descripción, fecha límite, estado (por hacer, en progreso, completada) y responsable (usuario o equipo).
- Editar la información de una tarea existente.
- Asignar una tarea a un usuario o equipo.

- Marcar una tarea como completada y, si es necesario, restablecerla al estado anterior.
- Eliminar tareas.

Colaboración y Comentarios

- Comentar en las tareas para discutir detalles y actualizaciones.
- Recibir notificaciones sobre nuevos comentarios en las tareas en las que participa el usuario.

Filtros y Búsqueda

- Filtrar tareas por estado, fecha límite, responsable, etc.
- Buscar tareas por palabras clave.

Autenticación y Autorización

- Solo usuarios autenticados pueden acceder y administrar proyectos y tareas.
- Implementar roles (administrador, miembro de equipo) con distintos niveles de permisos.

3. Requisitos Técnicos

Autenticación y Autorización

- Implementar autenticación basada en tokens **JWT** para controlar acceso.

Interfaz de Usuario

- Diseñar una UI intuitiva y atractiva usando **NextUI**, **Headless UI**, **Tailwind** u otra librería de componentes.

6. Respóndenlos vía email

La prueba dura **48 horas**. Si necesitas más tiempo, solicita una extensión. Envía tu respuesta por email con un enlace en la nube al proyecto entregable. ¡Éxito!

Repositorio en GitHub

- Proporciona el **código fuente completo** de los microservicios, incluidos archivos de configuración.
- Incluye un **README** con instrucciones claras para configurar y ejecutar el proyecto.
- Puedes usar cualquier sistema de control de versiones y alojar el código en un repositorio público.
- Entrega el enlace al repositorio.
- Se valorará cualquier **mejora o funcionalidad extra** añadida.
- Documenta las decisiones técnicas relevantes y los desafíos superados durante el desarrollo.