

Introducción

Bienvenido/a Estamos encantados de que estés aquí para demostrar tus habilidades y conocimientos. Esta prueba está diseñada para evaluar tu capacidad para resolver problemas, tu creatividad y tu forma de abordar los desafíos técnicos. Te deseamos mucho éxito en esta experiencia.

¡Estamos emocionados de ver de lo que eres capaz!

OBJETIVO DE LA PRUEBA

Desarrollar una mini aplicación de gestión de incidencias (Issue Tracker) que demuestre tus habilidades como desarrollador full stack

Para facilitar tu trabajo, proporcionamos un proyecto inicial con la configuración base de cada componente.

IMPORTANTE:

- ☐ Puedes modificar la estructura a tu antojo
- ☐ Se sumarán puntos por aplicar mejores prácticas y patrones de arquitectura que consideres apropiados.
- ☐ La estructura sugerida es sólo una guía
- ☐ Siéntete libre de reorganizar carpetas, agregar capas adicionales, o implementar patrones que prefieras.
- ☐ Cada servicio tiene su README
- ☐ Consulta los archivos README en cada carpeta para instrucciones de configuración y ejecución.

MÓDULO 1: BACKEND PRINCIPAL (Node.js)

- API REST que maneje usuarios, proyectos e incidencias
- Integración con servicio auxiliar en Python para clasificar issues
- Autenticación básica con JWT o sesión mock
- Base de datos: SQLite o archivo JSON

MÓDULO 2: SERVICIO AUXILIAR

- endpoint /classify
- Recibe title y description de un issue
- Retorna etiquetas (tags) automáticas basadas en reglas simples

Ejemplo: si el texto contiene "auth", "login", "token" → tag "security"

MÓDULO 3: FRONTEND (React)

- Pantallas requeridas:
- Login (mock)
- Lista de proyectos
- Listado de issues (vista tipo tabla o kanban simple)
- Formulario de creación/edición de issues (debe mostrar los tags generados)

Requisitos:

- Manejo de estados: loading, error, success
- Diseño simple pero ordenado (puedes usar Tailwind o CSS puro)

Stack tecnológico:

- Node.js: Express
- React: Vite o NextJS
- Base de datos: SQL (o JSON para simplicidad)

Documentación obligatoria:

- README.md: pasos para instalar y ejecutar el proyecto
- ARCHITECTURE.md: decisiones técnicas y arquitectura del proyecto
- GCP_PLAN.md: plan teórico de despliegue y respuestas al cuestionario

Opcional (pero recomendable):

- docker-compose.yml para levantar los tres servicios y una BD para pruebas

¡Éxito con tu prueba técnica!

Recuerda que no es obligatorio cumplir con todos los requisitos, haz lo que puedas hacer en el plazo planteado, la evaluación se hace sobre las buenas prácticas y lo que propones en tu desarrollo en general.

Estamos emocionados de ver tu solución.