



Challenge técnico - Aplicación de Gestión de Contactos

El candidato debe crear una aplicación web simple de Gestión de Contactos utilizando React para el Frontend y Express con PostgreSQL o MySQL como BBDD.

A continuación se te promoverá las tareas que debes completar:

Front-end:

- ☐ Ver una lista de contactos existentes.
- ☐ Agregar un nuevo contacto.
- ☐ Editar la información de un contacto.
- ☐ Eliminar un contacto.

Back-end:

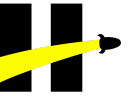
- ☐ Crear una base de datos PostgreSQL o MySQL para almacenar la información de los contactos.
- ☐ Crear rutas/APIs para listar, agregar, editar y eliminar contactos.
- ☐ Utilizar Express.js para manejar las rutas/APIs en el servidor.
- ☐ Conectar la base de datos con la aplicación utilizando el cliente de PostgreSQL.
- ☐ Utilizar Node.js para la lógica del servidor.

Funcionalidades EXTRA:

- ☐ Agregar la opción de enviar correos electrónicos directamente desde la plataforma. Esto puede lograrse integrando un servicio de envío de correos electrónicos (por ejemplo, Nodemailer).
- ☐ Implementar validación de datos en el servidor para prevenir entradas incorrectas.

El candidato debe proporcionar una documentación del proyecto que incluya lo siguiente:

- ☐ Una breve descripción de la aplicación y sus características.
- ☐ Instrucciones detalladas para configurar y ejecutar la aplicación en un entorno local.
- ☐ Descripción de las rutas/APIs creadas en el back-end, incluyendo cómo se utilizan y qué datos esperan/reciben.
- ☐ Descripción de las principales funciones y componentes utilizados en el front-end.
- ☐ Cualquier información adicional que ayude a comprender el proyecto.



Se evaluará la calidad del código, la funcionalidad, el cumplimiento de los requisitos y las buenas prácticas. Algunos aspectos a considerar para evaluar las buenas prácticas incluyen:

- Organización y estructura del código.
- Nombres descriptivos de variables y funciones.
- Comentarios explicativos en el código.
- Manejo de errores y excepciones.
- Seguridad en la aplicación (por ejemplo, protección contra inyección SQL).
- Uso de componentes reutilizables (en el caso de React).
- Eficiencia y rendimiento del código.

Mucha suerte!