

Desafío React / MongoDB

La presente prueba técnica tiene como objetivo evaluar tus conocimientos y habilidades en React y MongoDB. Reconocemos que no existe una única respuesta correcta para esta prueba, pero es fundamental que enfoques tu esfuerzo en cumplir únicamente con lo solicitado. Esto nos permitirá valorar tu capacidad para resolver problemas específicos, además de evaluar tu desempeño en términos de performance, estabilidad, diseño UI, y la calidad de tu código en cuanto a claridad y creatividad.

Tu tarea será desarrollar un proyecto utilizando React como Frontend, que interactúe con una API, como Backend, para consultar, mostrar y actualizar información en una base de datos MongoDB.

Te deseamos mucho éxito en esta prueba. Si tienes dudas o enfrentas algún problema, no dudes en contactarnos a través del correo electrónico: sergio@drenvio.com

Recursos

Clúster MongoDB

URL de conexión:

`mongodb+srv://drenviochallenge:m1jWly3uw42cBwp6@dr
enviochallenge.2efc0.mongodb.net/`

Base de datos: `tienda`

Colección inicial: `productos`

Criterios de evaluación:

Funcionabilidad: `La aplicación debe cumplir con los requerimientos planteados.`

Código Limpio: `Se valorarán las buenas prácticas en cuanto a formato, estructura y documentación.`

Creatividad: `Se evaluará la lógica y estructura que propongamos para la nueva colección en MongoDB.`

Requerimientos

Objetivo

Evaluar el conocimiento del participante en:

1. **JavaScript/TypeScript:** Deberás elegir uno de estos lenguajes y justificar tu elección en el archivo README.md.
2. **React:** Crear una interfaz sencilla que consuma, muestre y actualice datos mediante una API.
3. **Node.js:** Implementar una API básica para interactuar con MongoDB.
4. **MongoDB:** Realizar operaciones de lectura y escritura en el clúster proporcionado.

Desafío React / MongoDB

Instrucciones

Configura el entorno del proyecto con las siguientes herramientas:

- **Frontend:** React.
- **Backend:** Node.js.
- **Base de datos:** MongoDB (URL de conexión proporcionada).

Creación de una nueva colección:

- Crea una colección en el clúster de MongoDB con el siguiente formato para el nombre:
preciosEspeciales<tuPrimerApellido><dosNúmeros>
Por ejemplo, si tu apellido es Curiel y eliges el número 13, el nombre será:
preciosEspecialesCuriel13
- Diseña la estructura de la colección para que sea funcional y óptima en el contexto de la pantalla **Artículos**.
Considera incluir campos que permitan relacionar los datos de esta colección con los usuarios que tienen precios especiales, optimizando las consultas necesarias.

Crea un menú de navegación con dos opciones:

- **Artículos:** Muestra una tabla con datos de la colección productos, el precio cambiara si el usuario en cuestión se encuentra en tu colección para precios especiales.
- **Subida:** Muestra un formulario para agregar datos a la colección de precios especiales que creaste, deberás diseñar dicha colección en el clúster de mongoDB.

Servicios API:

- Obtener los datos de la colección productos.
- Consultar y actualizar tu colección de precios especiales.
- Validar la existencia de usuarios en la colección de precios especiales para la pantalla Artículos.

Requisitos técnicos adicionales:

- Documenta tu código para que sea fácil de entender.
- Explica en el archivo README.md como ejecutar el proyecto localmente
- Si lo prefieres, puedes utilizar algún gestor de estados.

Esquema sugerido para el archivo README.md:

- Introducción
- Pasos para ejecutar localmente
- Justificación de elecciones técnicas
- Descripción de la estructura del proyecto

Desafío React / MongoDB

Al finalizar

- Publica el proyecto en un repositorio público de GitHub.
- Opcional: Implementa un deploy funcional (por ejemplo, utilizando Vercel, Netlify, o cualquier servicio que prefieras).
- Envía un correo a sergio@drenvio.com con:
 - La URL del repositorio.
 - La URL del deploy (si aplica).
 - Comentarios sobre la prueba, dificultades enfrentadas y cualquier sugerencia adicional.