

Prueba Técnica de FontEnd

Desarrollar una **aplicación web de una sola página (SPA)** utilizando tecnologías modernas y buenas prácticas para gestionar un conjunto de **posts** provenientes de una API REST externa (<https://jsonplaceholder.typicode.com/guide/>). La aplicación permitirá a los usuarios:

- **Listar** los posts de forma paginada en una tabla con estilo datatable.
- **Crear** nuevos posts a través de un formulario modal con validación de campos.
- **Editar** posts existentes.
- **Eliminar** posts.
- **Notificar** al usuario sobre el resultado de cada acción realizada.

Tecnologías Propuestas:

- **React:** Framework de JavaScript para construir interfaces de usuario.
- **Next.js:** Framework basado en React para crear aplicaciones web y SSR/SSG.
- **React Hook Form:** Librería para gestionar formularios de manera eficiente.
- **Axios:** Cliente HTTP para realizar peticiones a la API.
- **Material UI:** Framework de componentes UI basado en Material Design (opcional).
- **Bootstrap:** Framework de componentes CSS (opcional).
- **Redux:** Librería de gestión de estado global (opcional).
- **TypeScript:** Lenguaje de tipado estático para JavaScript (opcional).

Estructura y Funcionalidades:

1. Página Principal:

- Tabla paginada con los posts, incluyendo:
 - Título del post

- Contenido (resumen)
- Botones de edición y eliminación
- Barra de búsqueda para filtrar los posts.
- Botón para crear un nuevo post.

2. **Formulario Modal:**

- Campos para ingresar el título y contenido del post.
- Validación de campos requeridos.
- Botones para guardar o cancelar.
- Mensajes de éxito o error.

3. **Gestión de Estado:**

- Utilizar un estado global (con Redux o Context API) para gestionar la lista de posts, el post seleccionado y los mensajes de notificación.

4. **Interacción con la API:**

- Utilizar Axios para realizar las peticiones HTTP a la API externa.
- Implementar la lógica para crear, leer, actualizar y eliminar posts.
- Manejar los errores de la API de forma adecuada.

5. **Estilos:**

- Aplicar un diseño limpio y moderno utilizando Material UI , Bootstrap , NextUI .
- Asegurar la responsividad de la aplicación para diferentes dispositivos.

Consideraciones Adicionales:

- **Accesibilidad:** Asegurar que la aplicación sea accesible para todos los usuarios.

Entrega del Proyecto:

- **Repositorio de GitHub:** Crear un repositorio público en GitHub para alojar el código fuente.

- **Despliegue en Vercel:** Configurar el despliegue continuo en Vercel (opcional).

Evaluación:

La evaluación del proyecto se basará en los siguientes criterios:

- **Funcionalidad:** Completitud de las funcionalidades requeridas.
- **Calidad del código:** Estructura, legibilidad, mantenibilidad y buenas prácticas.
- **Diseño:** Usabilidad, estética y responsividad.
- **Documentación:** Claridad y completitud de la documentación.

Sugerencias Adicionales:

- **Personalización:** Agregar funcionalidades adicionales como:
 - Ordenamiento de los posts por diferentes criterios.
- **Mejoras en la UI:**
 - Utilizar un editor de texto enriquecido para el contenido de los posts.
 - Implementar un sistema de notificaciones persistentes.