

Trabajo Práctico de React:
Desarrollo de Frontend en React para el
Sistema de Gestión de Productos

Diseñar y desarrollar una aplicación web utilizando React. El sitio deberá ser visualmente atractivo y funcional.

Descripción del Proyecto: La empresa de manufactura que desarrolló el sistema backend para gestionar productos, fabricantes y componentes desea ahora un frontend en React que permita a los usuarios visualizar la información almacenada en la API de gestión de productos. El sistema solo necesita mostrar información, sin opciones de modificación ni eliminación, enfocándose en la navegación y presentación del catálogo de productos.

Objetivo: Construir una aplicación web en React que consuma el backend de la API de gestión de productos. La aplicación permitirá a los usuarios navegar por un catálogo de productos, visualizar detalles de productos específicos y sus fabricantes y componentes asociados.

Páginas:

Para cumplir con el trabajo, deben generar 4 páginas Web con las siguientes características (como mínimo esto, si agregan más no hay problema)

Página Principal: Inicio

La página inicio deberá contener lo siguiente:

- Presentación de la empresa, incluyendo su historia y filosofía.
- Un menú de navegación que permita acceder a todas las secciones de la página.
- Imágenes representativas del local y demás, con texto alternativo adecuado.
- Links en imágenes. Estos links deben abrir en pestañas nuevas y deben ser videos de Youtube sobre lo que muestre la imagen.
- Por lo menos un video tipo iframe de Youtube.
- Párrafos y títulos descriptivos.
- Información de contacto, como dirección, teléfono y horario de apertura.

- Un mapa interactivo (puede ser un iframe de Google Maps) mostrando la ubicación de la empresa.
- En el pie de página deben estar los datos de los integrantes del grupo.

Página de Catálogo de Productos:

- Mostrar una lista de productos disponibles, con nombre, descripción, precio y la imagen en cada tarjeta.
- Permitir al usuario hacer clic en un producto para ver más detalles.
- Utilizar **fetch** para obtener datos desde la API.
- **(Opcional)** Deberá ser dinámica esta sección, es decir, dado una lista de productos, el usuario va a poder seleccionar los productos que desee y su cantidad. Además, deberá tener un botón para calcular el precio total de los productos que seleccionó el usuario del catálogo.

Página de Detalle de Producto:

- Mostrar detalles completos de un producto seleccionado, incluyendo información como descripción, precio y componentes asociados.
- Listar los fabricantes y componentes del producto con enlaces a sus respectivas páginas de detalles.

Página de Fabricantes y Componentes:

- Página dedicada para mostrar la lista completa de fabricantes y componentes, accesibles desde la página principal.

Requisitos Técnicos

Componentes y Props:

- Crear componentes reutilizables para listar productos, fabricantes y componentes.
- Usar props para pasar los datos a cada componente de detalle y así mostrar la información de cada producto, fabricante y componente en una tarjeta o lista.

Framework CSS:

- Pueden utilizar un framework CSS (como Bootstrap, Material UI o Tailwind) para estilizar la aplicación.
- Aplicar estilos personalizados para resaltar el diseño de un catálogo de productos.

Estado y Ciclo de Vida:

- Utilizar el estado en los componentes para almacenar la información obtenida desde la API.

- Implementar métodos de ciclo de vida o hooks como useEffect para cargar la lista de productos, fabricantes y componentes cuando sea necesario.

Eventos:

- Agregar manejadores de eventos para acciones simples, como hacer clic en un producto para ver más detalles o seleccionar un fabricante.
- Asegurarse de que los eventos actualicen la visualización en la interfaz sin modificar los datos del backend.

Rutas y Navegación:

- Configurar rutas con react-router-dom para navegar entre las diferentes secciones de la aplicación: una página principal de catálogo de productos y una página de detalles para cada producto.
- Permitir navegación entre las páginas, como una ruta específica para los detalles de un producto (/productos/:id) que muestre la información completa del producto seleccionado.

Fetch y Consumo de APIs:

- Usar fetch o una librería como axios para obtener datos de la API mediante los métodos GET.
- Cargar y mostrar datos de productos, fabricantes y componentes, y manejar errores en caso de fallos en la conexión o en la carga de datos.

Backend:

Pueden usar la aplicación que armaron en la materia de Estrategia de Persistencia. En caso de usarla, no se olviden de usar el paquete "cors".

Para los que no están cursando persistencia pueden usar esta aplicación de Backend:

<https://github.com/CIU-UnaHur/tp-backend>

Acuerdense para correr la aplicación de Backend hacer:

- npm i
- npm run dev

A continuación se detallan las rutas disponibles en la API:

http://localhost:5000/productos	Obtener todos los productos
http://localhost:5000/productos/:id/	Obtener un producto en particular
http://localhost:5000/productos/:id/fabricantes	Obtener todos los fabricantes de un producto

http://localhost:5000/productos/:id/componentes	Obtener todos los componentes de un producto
http://localhost:5000/fabricantes	Obtener todos los fabricantes
http://localhost:5000/componentes	Obtener todos los componentes

Instrucciones

La aplicación en React deberá tener al menos los siguientes componentes:

- **Header:** Deberá mostrar el nombre de la empresa y un menú de navegación.
- **Footer:** Deberá incluir los datos de los integrantes del grupo.

Organizar los componentes en una carpeta llamada components.

Pueden utilizar bootstrap, react-bootstrap, tailwind o la herramienta que deseen.

Entrega

1. Subir el código del proyecto a un repositorio en GitHub.
2. Incluir un archivo README.md con una descripción del proyecto y las instrucciones para correr la aplicación.
3. Deberán mandar el link al repo (recordar de hacerlo público) al siguiente mail:
lucas.figarola@unahur.edu.ar (incluir los datos de los integrantes del grupo)

La fecha y hora límite para la entrega del parcial es el domingo 28/11 a las 18:00hs.