

Ejercicios Prácticos



TalentLab – Proyecto Final

¡Felicitaciones nuevamente! 🎉 Ahora que has dominado la primera etapa de tu tarea en **TalentLab**, el equipo ha decidido asignarte una responsabilidad más avanzada.

Ejercicio Práctico

Obligatorio



Silvia

Product Owner

Descripción de tu tarea:

Tu próximo objetivo será conectar la aplicación a una API que provea información sobre los productos. Deberás mostrar los datos en tiempo real, manejar estados de carga y errores, y continuar utilizando un diseño atractivo para el eCommerce.

Ejercicio Práctico

Obligatorio

Requisitos del Proyecto:

Integración con una API:

- Usa **MockAPI** para obtener una lista de productos desde un endpoint.
- Asegúrate de configurar correctamente la URL del recurso.

Gestión del estado con `useState`:

- Almacena los productos obtenidos de la API en un estado local.
- Usa otro estado para gestionar si la aplicación está en proceso de carga.

Manejo de efectos secundarios con `useEffect`:

- Realiza la petición a la API cuando el componente principal de productos se monte.
- Gestiona posibles errores y muestra un mensaje adecuado si ocurre algún problema.

Ejercicio Práctico

Obligatorio

Requisitos del Proyecto:

Estado de carga y errores:

- Mientras los productos están cargando, muestra un mensaje o un spinner con "Cargando productos...".
- Si ocurre un error, muestra un mensaje como: "Error al cargar productos. Inténtalo más tarde."

Actualizar el diseño del eCommerce:

- Crea una sección de productos dinámicos que se cargue desde la API.
- Asegúrate de mantener el diseño atractivo y responsivo.

Ampliación del carrito:

- Permite agregar al carrito productos que ahora se obtendrán desde la API.
- Si el carrito está vacío, muestra el mensaje "El carrito está vacío", como en el ejercicio anterior.

Ejercicio Práctico

Obligatorio

Pautas generales del proyecto:

1 Estructura

Continuar con el proyecto de la clase anterior, agregando nueva funcionalidad.

2 Componentes

Mantener componentes organizados y reutilizables.

3 Código limpio

Evitar redundancias y organizar estados y efectos claramente.

4 Diseño

Usar Bootstrap o alternativa para mantener un diseño profesional.