

Trabajo Final

1. Configuración inicial:

- Crear un nuevo proyecto React.js utilizando Vite.
- Organiza tu proyecto en componentes reutilizables para una mejor estructura.
- Configura las rutas utilizando React Router para navegar entre las diferentes secciones del tu e-commerce.

Aca tienen una propuesta inicial de las rutas que pueden utilizar en el proyecto:

- Ruta de inicio de sesión: `"/login"`
- Ruta de registro de usuario: `"/register"`
- Ruta de inicio (página principal): `"/"`
- Ruta de categorías: `"/categories"`
- Ruta de productos: `"/products"`
- Ruta de detalle de producto: `"/products/:id"`
- Ruta de creación de producto (solo para usuarios administradores): `"/products/create"`
- Ruta de edición de producto (solo para usuarios administradores): `"/products/edit/:id"` (donde `":id"` representa el ID del producto a editar)
- Ruta de detalle del carrito: `"/cart-detail"`

Estas son solo algunas rutas básicas para comenzar. Pueden ajustarlas según las necesidades que encuentren y agregar rutas adicionales si es necesario.

2. Vista de categorías

- Crea una vista para mostrar las categorías disponibles.
- Utiliza la API para obtener la lista de categorías.
- Muestra las categorías en una lista o en un diseño de tu elección.
- Al hacer click en una categoría se debería redirigir al listado de producto. Ese listado de producto debe tener activado el filtro por la categoría por la que se ingresó

3. Vista de productos

- Crea una vista para mostrar los productos disponibles.
- Utiliza la API para obtener la lista de productos.
- Muestra los productos en una lista o en un diseño de tu elección.
- Agrega opciones de filtrado, por ejemplo, título, categoría, precio y rango de precios. Puedo filtrar por una o mas de una condición.

4. Autenticación y registro de usuarios

- Crea componentes para el registro y el inicio de sesión de usuarios.
- Utiliza el formulario para recopilar los datos necesarios, como nombre, correo electrónico, contraseña, etc.
- Consume la API para registrar y autenticar usuarios.
- Implementa la funcionalidad de inicio de sesión y registro de usuarios para obtener tokens de autenticación.
- Guardar la sesión del usuario en el local storage para evitar tener que iniciar sesión al volver ingresar a la página. Dar la opción de cerrar la sesión y remover la información del usuario del local storage
- Tener en cuenta que si ya estoy loggeado, y me dirijo a la página de login o registro, debo redireccionar al usuario a la página principal.

5. Creación y edición de productos (solo para usuarios administradores):

- Crea formularios para agregar y editar productos.
- Implementa la lógica para enviar las solicitudes de creación y actualización de productos a la API.
- Asegúrate de que solo los usuarios con el rol de administrador puedan acceder a estas funcionalidades. Teniendo en cuenta que si ingreso a la página de creacion y edicion de productos, y no soy usuario admin, debo redirigir al usuario a la página principal

6. Creación y edición de categorías (solo para usuarios administradores):

- Crea formularios para agregar y editar categorías.
- Implementa la lógica para enviar las solicitudes de creación y actualización de categorías a la API.

- Asegúrate de que solo los usuarios con el rol de administrador puedan acceder a estas funcionalidades. Teniendo en cuenta que si ingreso a la página de creación y edición de categorías, y no soy usuario admin, debo redirigir al usuario a la página principal

7. Eliminación de productos y categorías (solo para usuarios administradores)

- Los usuarios administradores en el listado deben tener visible un botón para eliminar el ítem. Al hacer click abrir un modal para que el usuario pueda confirmar la acción que desea realizar.

8. Estado del carrito:

- Crea un estado global o contexto(React Context) para almacenar la información del carrito.
- El estado del carrito puede ser un arreglo de objetos, donde cada objeto representa un producto en el carrito y contiene información como el ID del producto, título, precio, cantidad, etc.

9. Agregar productos al carrito:

- Agrega un botón "Agregar al carrito" en la vista de productos.
- Al hacer clic en el botón, captura la información del producto seleccionado y agrégala al estado del carrito.
- Si el producto ya existe en el carrito, actualiza la cantidad en lugar de agregar un nuevo objeto.

10. Actualizar el precio del carrito y la cantidad de ítems:

- Calcula el precio total del carrito sumando el precio de cada producto multiplicado por la cantidad.
- Actualiza el estado del carrito con el nuevo precio total y la cantidad total de ítems.

11. Visualización del carrito:

- Crea una vista para mostrar el contenido del carrito.
- Muestra los productos agregados al carrito, junto con su título, precio, cantidad y subtotal.
- Proporciona opciones para aumentar o disminuir la cantidad de cada producto en el carrito.

- Actualiza automáticamente el precio total del carrito y la cantidad de ítems al cambiar la cantidad de un producto.

12. Persistencia del carrito (opcional):

- Utiliza el almacenamiento local (localStorage) para almacenar el estado del carrito.
- Al cargar la página, verifica si hay un carrito guardado en el almacenamiento local y cárgalo en el estado del carrito.
- Asegúrate de guardar y actualizar el estado del carrito en el almacenamiento local cada vez que se realicen cambios.

13. Finalización de la compra:

- En el detalle del carrito, agrega un botón para finalizar la compra. Luego de hacer click en el boton redirigir a una página indicando que la compra fue realizada con éxito, con la opción de volver a la página principal
- Limpia el carrito después de una compra exitosa.

Consideraciones

- Todas las pantallas que involucren llamado a una API deben contar con 3 estados: loading, error y success
- Agregar pantalla de 404 Not Found, para cuando quiero ingresar a una ruta que no tenga asociada ningún componente

Metodología

La entrega se dividirá en 5 etapas. Su proyecto lo deben subir a GitHub. Y deberán crear una rama y posteriormente un pull-request para cada etapa. Es decir, su proyecto al finalizar el trabajo deberá contar con 5 ramas (mínimo) y 5 pull-requests.

También deberán contar con su aplicación desplegada para que pueda ser utilizada.

Tengan en cuenta que si bien existe una fecha límite, es súper normal que no se llegue con todo el desarrollo que se propuso. No pasa nada, en ese caso creen el pr con lo que tengan. De todas formas traten siempre de cumplir con lo requerido para que al final del proyecto no les quede muy pesado y puedan llegar a desarrollar todo.

Etapas

Etapa 1

Involucra desarrollar:

- 1 - Configuración inicial
- 2 - Vista de categorías

Fecha límite para crear el pull request: 10/07

Etapas 2

Involucra desarrollar:

- 3 - Vista de productos
- 4 - Autenticación y registro de usuarios

Fecha límite para crear el pull request: 17/07

Etapas 3

Involucra desarrollar:

- 5 - Creación y edición de productos (solo para usuarios administradores)
- 6 - Creación y edición de categorías (solo para usuarios administradores)
- 7 - Eliminación de productos y categorías (solo para usuarios administradores)

Fecha límite para crear el pull request: 27/07

Etapas 4

Involucra desarrollar:

- 8 - Estado del carrito
- 9 - Agregar productos al carrito
- 10 - Actualizar el precio del carrito y la cantidad de ítems

Fecha límite para crear el pull request: 31/07

Etapas 5

Involucra desarrollar:

- 11 - Visualización del carrito
- 12 - Persistencia del carrito
- 13 - Finalización de la compra

Fecha límite para crear el pull request: 07/08

Links de recursos

La API a consumir es: <https://fakeapi.platzi.com/en/rest/introduction>

Pueden basarse el siguiente diseño, o en el diseño que mas deseen:

<https://www.figma.com/community/file/966016571279781800>