

Enrutamiento

Módulo 2 – Unidad 3

Mini dashboard - Rutas públicas y protegidas



Objetivos

Practicar:

- Configuración básica con `BrowserRouter`, `Routes` y `Route`.
- Navegación con `<Link>` y `useNavigate`.
- **Rutas dinámicas** con `useParams`.
- **Query params** con `useSearchParams`.
- **Rutas anidadas** con `<Outlet>`.
- **Ruta protegida** con redirección *post-login* (usando `useLocation` y `Navigate`).



Consigna

Consigna

1. Crear el proyecto base.

- Iniciar un nuevo proyecto React.
- Envolver toda la app con el enrutador ([BrowserRouter](#)).

2. Definir rutas básicas.

- Crear tres páginas: **Inicio**, **Nosotros** y **Contacto**.
- Configurar el enrutamiento para que cada URL muestre su página correspondiente.
- Añadir enlaces de navegación para moverse entre ellas.

3. Agregar una ruta dinámica.

- Crear una página **Producto** que reciba un parámetro en la URL (por ejemplo, [/producto/:id](#)).
- Mostrar en la página el número o nombre del producto recibido como parámetro.

4. Explorar navegación programática.

- Crear un botón que, al hacer clic, lleve automáticamente al usuario a otra página utilizando navegación imperativa.

5. Opcional.

- Añadir un *layout* con navegación fija y aplicar rutas anidadas para que las páginas se muestren dentro de él.
- Simular una ruta protegida que muestre un mensaje si el usuario no está "logueado".

Formato de presentación

La entrega debe realizarse a través de un **repositorio en GitHub**, el cual debe incluir:

- Proyecto base creado en **React (Vite recomendado)**, con archivos innecesarios eliminados.
- Configuración del enrutamiento con **BrowserRouter, Routes y Route**.
- Páginas principales:
 - **Inicio.jsx**
 - **Nosotros.jsx**
 - **Contacto.jsx**
- Enlaces de navegación con **<Link>** o navegación programática con **useNavigate**.
- Página dinámica:
 - **Producto.jsx** mostrando el parámetro recibido vía **useParams**.
- Ejemplo de navegación imperativa con **useNavigate**.

- Ejemplo de **query params** usando `useSearchParams` (si aplica).
- Ejemplo de **rutas anidadas** utilizando `<Outlet>` y un *layout* de navegación fija (opcional).
- Ejemplo de **ruta protegida** con redirección *post-login* usando `useLocation` y `Navigate`.
- **Archivo README.md** con:
 - Descripción breve del proyecto.
 - Instrucciones para clonar, instalar dependencias y ejecutar (`npm install` / `npm run dev`).
 - Capturas de pantalla mostrando las diferentes rutas (página estática, dinámica, protegida).
 - Créditos del autor (nombre del estudiante, curso, unidad).
 - Citación de fuentes (bibliografía y créditos de imágenes o recursos si corresponde).

Criterios de evaluación

- Correcta configuración del enrutamiento con **BrowserRouter**, **Routes** y **Route**.
- Implementación de páginas estáticas (Inicio, Nosotros, Contacto).
- Uso de **Link** y **useNavigate** para navegación declarativa y programática.

- Implementación de una **ruta dinámica** con `useParams`.
- Uso correcto de **query params** con `useSearchParams`.
- Simulación de una **ruta protegida** con `Navigate` y `useLocation`.
- Organización del proyecto en componentes y páginas.
- Presentación limpia y prolífica del repositorio (estructura clara, README completo).
- Buenas prácticas de sintaxis y código indentado correctamente.



Bibliografía utilizada y sugerida

Libros y otros manuscritos

Abramov, D. y Clark, A. *Fullstack React: The Complete Guide to ReactJS and Friends*. 1^a ed. Accomazzo LLC; 2017.

Banks, A. y Porcello, E. *Learning React: Modern Patterns for Developing React Apps*. 2^a ed. O'Reilly Media; 2020.

Artículos y documentación en línea

MDN Web Docs. (s.f.). *URLSearchParams*. Mozilla Corporation.

<https://developer.mozilla.org/en-US/docs/Web/API/URLSearchParams>

React Router. (s.f.). *Tutorial*. <https://reactrouter.com/en/main/start/tutorial>

React Router API Reference. *Function useNavigate*.

<https://reactrouter.com/en/main/hooks/use-navigate>