Open Opened 5 days ago by Rubén Montero

Rutas anidadas (usando Outlet)

Resumen

- Crearemos un nuevo componente Container.js que mostrará ¡Hola!
- Container.js será el contenedor de todas las páginas, poniéndolo como <Route /> padre
- Además, en Container.js incluiremos <Outlet /> , que actuará como lienzo donde cada ruta hija podrá pintarse

Descripción

Hasta ahora, hemos escrito todas las <Route> al mismo nivel, como hermanas. Como en el siguiente ejemplo:

```
<Routes>
  <Route path='equipos' element={<Equipos/>} />
  <Route path='equipos/jugadores' element={<Jugadores/>} />
  </Routes>
```

¡Pero se pueden anidar! Observa:

```
<Routes>
     <Route path='equipos' element={<Equipos/>} >
          <Route path='jugadores' element={<Jugadores/>} />
          </Route>
          </Routes>
```

Esto será muy útil para implementar un panel lateral o un encabezado común que esté presente en todas las páginas.

Vamos a hacerlo, aprendiendo, en el camino, cómo funcionan las <u>rutas anidadas</u> y <Outlet /> en react-router-dom.



La tarea

Crea src/screens/container/Container.js:

Y luego **modifica** App.js para que las rutas sean así:

• A cada path (excepto wildcard *) le hemos eliminado el prefijo /, que ahora pertenece a la ruta padre (<Container />).

Ahora verifica, por ejemplo, http://localhost:3000/register

¡Vaya...! ¡Sólo se ve ¡Hola!! ¡Hemos roto nuestra web!

De acuerdo a la documentación:

https://raspi/francisco.gomez/dwec/issues/149

An Outlet should be used in parent route elements to render their child route elements

Añade lo siguiente a Container.js:

```
+import { Outlet } from "react-router-dom"
const Container = () => {
   return <div>
       <div data-cy="topBar">¡Hola!</div>
       <Outlet/>
     </div>
}
```

Observarás que casi todas las páginas funcionan ahora bien, y además muestran el ¡Hola! .

<u>index</u>

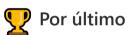
Sin embargo, la principal (http://localhost:3000/) aún no se ve bien.

Añade lo siguiente en App.js:

```
<Routes>
  <Route path='/' element={<Container/>}>
    <Route index element={<Dashboards/>}></Route>
    <Route path='register' element={<UserRegister/>}></Route>
</Routes>
```

De esta manera, indicas que <Dashboards /> es la principal (índice), y debe poblar el <Outlet /> en caso de que se visite la ruta principal (y no una subruta).

Verifica ahora http://localhost:3000/



Sube tus cambios al repositorio en un nuevo commit.



Rubén Montero @ruben.montero changed milestone to MSprint 5 5 days ago