Open Opened 3 weeks ago by Rubén Montero

useNavigate



- Modificaremos DashboardListItem.js para que navegue a la página http://localhost:3000/dashboard/:id
- El :id será variable, en función de qué celda cliquemos (1, 2,...)
- De momento, en http://localhost:3000/dashboard/id no habrá ninguna ruta preparada

Descripción

El API REST no sólo ofrece una lista de dashboards. ¡También datos específicos para cada dashboard!

Vamos a implementar navegación desde cada celda en Main.js.

Usaremos el hook <u>useNavigate</u>



La tarea

Añade este import a DashboardListItem.js:

```
import { useNavigate } from 'react-router-dom';
```

Luego, invoca useNavigate para obtener un componente que te permitirá navegar a otra página.

```
const DashboardListItem = (props) => {
+ const navigate = useNavigate();
 const onClick = () => {
   console.log('clicked')
 }
```

Y, finalmente, en el onClick, añade la siguiente línea para navegar a /dashboards/1, /dashboards/2,... (según qué celda fue clicada)

```
const onClick = () => {
 navigate('/dashboards/'+props.dashboard.id)
  console.log('clicked')
}
```

¡Enhorabuena! Has empleado una poderosa alternativa a Link . Ahora, si clicas en cada celda de http://localhost:3000 te llevará a:

- http://localhost:3000/dashboards/1
- http://localhost:3000/dashboards/2
- http://localhost:3000/dashboards/3
- http://localhost:3000/dashboards/4
- http://localhost:3000/dashboards/5

De momento, verás la página de Not Found. ¡Está bien!

En la siguiente tarea implementaremos el detalle de un dashboard (!)





Por último

Sube tus cambios al repositorio en un nuevo commit.

1. ¿Se te ocurre cómo implementar esta navegación usando un <Link> en vez de un <button> con un navigate?



1/2



Rubén Montero @ruben.montero changed milestone to MSprint 4 3 weeks ago

https://raspi/francisco.gomez/dwec/issues/114