

Otra vez... ¡un contador!

Resumen

- Añadiremos un `<div data-cy='issue3body'>` en `Main.js` y `About.js`
- Añadiremos un componente sencillo que consista en un contador
- Lo usaremos en la página `/examples`

Descripción

Hasta ahora hemos empleado `useState` de React para almacenar un estado en un componente.

`useState` es un [hook](#). Existe en React desde la versión 16.8 de [febrero de 2019](#). Antes de ese día... ¡los componentes se escribían como *clases*!

Hay dos *hooks* principales: `useState` y `useEffect`. Entenderemos y usaremos `useEffect` en futuras tareas.



La tarea

Añade un `<div data-cy='issue3body'>` en `Main.js` y `About.js`, de tal manera que englobe el `<h2>` ya existente.

Luego, crea una carpeta `src/components/`, y, dentro de una subcarpeta, el crea siguiente componente:

- `src/components/simple_counter/SimpleCounter.js`

```
import { useState } from "react"

const SimpleCounter = () => {
  const [number, setNumber] = useState(0);

  return <div>
    <p>Número: {number}</p>
    <button data-cy='issue3button' onClick={ () => {setNumber(number+1) }}>Incrementar</button>
  </div>
}
```

Para finalizar, en `Examples.js`, añade un `<div data-cy='issue3div'>` debajo (*a continuación*) del `<h2>` del ejercicio anterior. Dentro de ese `<div>`, incluye un `<h1>` y `<SimpleCounter />` así:

```
<div data-cy='issue3div'>
  <h1>Ejercicio 3</h1>
  <SimpleCounter />
</div>
```

¿Funciona correctamente <http://localhost:3000/examples>?



Por último

Sube tus cambios al repositorio en un nuevo *commit*.



Rubén Montero @ruben.montero changed milestone to %Sprint 3 3 hours ago