Open Opened 5 days ago by Rubén Montero

Paginación en APIs REST



- Comprenderemos la necesidad de paginación cuando se consumen datos en una aplicación
- Hablaremos de un tipo sencillo de paginación con offset y size
- Añadiremos page_size a la petición de detalle de dashboard (lista de preguntas) para empezar a trabajar

Descripción

Hasta ahora, cada vez que atacamos un endpoint y consumimos datos mediante un HTTP GET, se devuelven unos pocos.

Por ejemplo, el *endpoint* de *dashboards* <u>http://raspi:8082/api/v2/dashboards</u> devuelve 5 datos. ¡Qué pocos!

¿Y si hubiera más? ¿Y si hubiera 1.000.000.000?

Cuando se diseña e implementa un endpoint REST que previsiblemente accede a muchos datos, se suele implementar paginación.

Paginación

Imagina que un endpoint ficticio:

• GET http://servidorDeEjemplo:8081/mensajes

...devuelve los siguientes datos:

Posición	Contenido	Fecha
1	¡Empezamos la semana!	Lunes (9:00)
2	Ni te cases ni te embarques	Martes (9:00)
3	Media semana, ¡qué bien!	Miércoles (9:00)
4	El día de Júpiter	Jueves (9:00)
5	Hoy empieza el fin de semana 😇	Viernes (9:00)
6	¡Un sábado de deporte!	Sábado (9:00)
7	¡Bonito domingo de callos!	Domingo (9:00)

Si el API REST lo soporta podemos implementar una versión sencilla de paginación empleando dos parámetros de query ó query params¹:

- offset: Posicion desde donde extraer
- size : Número de datos máximo a devolver

Así, por ejemplo, podríamos consumir los datos en páginas de 3 en 3:

• GET http://servidorDeEjemplo:8081/mensajes?offset=0&size=3 (página 1)

Posición	Contenido	Fecha
1	¡Empezamos la semana!	Lunes (9:00)
2	Ni te cases ni te embarques	Martes (9:00)
3	Media semana, ¡qué bien!	Miércoles (9:00)

• GET http://servidorDeEjemplo:8081/mensajes?offset=3&size=3 (página 2)

Posición	Contenido	Fecha
4	El día de Júpiter	Jueves (9:00)

https://raspi/francisco.gomez/dwec/issues/133

Posición	Contenido	Fecha
5	Hoy empieza el fin de semana 😈	Viernes (9:00)
6	¡Un sábado de deporte!	Sábado (9:00)

GET http://servidorDeEjemplo:8081/mensajes?offset=6&size=3 (página 3)

Posición	Contenido	Fecha
7	¡Bonito domingo de callos!	Domingo (9:00)



La tarea

Las siguientes peticiones en nuestro API REST soportan paginación:

- /api/v2/dashboards/<dashboardId>
- /api/v2/dashboards/<dashboardId>/questions/<questionId>

Visita http://localhost:3000/dashboards/2. Se ven 7 elementos, ¿verdad?

Ahora, modifica DashboardDetail.js para que en la petición al API REST se envíe el parámetro ?page_size=5.

```
useEffect(() => {
 axios.get('http://raspi:8082/api/v2/dashboards/' + params.dashboardId).then(response => {
 axios.get('http://raspi:8082/api/v2/dashboards/' + params.dashboardId + '?page_size=5').then(response => {
    setDashboardTitle(response.data.title);
    setDashboardDescription(response.data.description);
    setQuestions(response.data.questions);
 })
}, [])
```

Visita de nuevo http://localhost:3000/dashboards/2. Ahora sólo se ven 5 elementos, ¿verdad?

Continuaremos implementando paginación en la siguiente tarea.

¡Spoiler! No será mediante un parámetro offset , pues presenta sus problemillas 🤢



Por último

Sube tus cambios al repositorio en un nuevo commit.

1. Los query params son parte del estándar HTTP. Viajan en la URL tras un signo de interrogación (?key=value). Si hay más de uno, se anexan con un ampersand (?key1=value1&key2=value2&key3=value3)



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