Open Opened 3 weeks ago by Rubén Montero

## Subir (POST) una pregunta (IV)



• Añadiremos un nuevo estado questionsPublishedRightNow que servirá para que React auto-refresque la página tras publicar una pregunta

## Descripción

En la tarea anterior dijimos:

```
si publicas una nueva pregunta, tras refrescar (F5) la página... ¡Allí está!
```

¿No es un poco raro que el usuario tenga que actualizar manualmente?

Mejorémoslo.

Vamos a aprovechar que, en DashboardDetailPage.js ya tenemos implementado un useEffect que pide inicialmente la lista de preguntas:

```
useEffect(() => {
  axios.get('http://raspi:8081/api/v1/dashboards/' + params.idDashboard).then(response => {
    setTitle(response.data.title);
    setDescription(response.data.description);
    setQuestions(response.data.questions);
  })
}, [])
```

...y modificándolo un poquito, conseguiremos que se lance nuevamente cada vez que el usuario añada una nueva pregunta.



## La tarea

**Añade** el siguiente estado a DashboardDetailPage.js:

```
const [questionsPublishedRightNow, setQuestionsPublishedRightNow] = useState(0)
```

...y **modifica** el **useEffect** ya existente para que sea afectado por dicho estado:

```
useEffect(() => {
   axios.get('http://raspi:8081/api/v1/dashboards/' + params.idDashboard).then(response => {
      setTitle(response.data.title);
      setDescription(response.data.description);
      setQuestions(response.data.questions);
   })
- }, [])
+ }, [questionsPublishedRightNow])
```

Ya sólo falta alterar este estado para que React reaccione y, mediante el efecto, pida nuevamente las preguntas. Por lo tanto, aparecerá la pregunta recién añadida.

En handleSubmit, implementa .then para la petición así:

```
const handleSubmit = (e) => {
    e.preventDefault();
    if ((formTitle.length === 0) || (formDescription.length == 0)) {
        return;
    }

    console.log("Quieres publicar la pregunta con título:")
    console.log(formTitle);
    console.log(formTitle);
    console.log("...y descripción:");
    console.log(formDescription);

    const httpRequestBody = { title: formTitle, description: formDescription }
    axios.post('http://raspi:8081/api/v1/dashboards/' + params.idDashboard + '/questions', httpRequestBody)
```

¡Listo! Ya puedes comprobar que, al subir una pregunta, la página muestra la pregunta recién añadida automáticamente.

## Unas últimas consideraciones...

- ¿Qué pasaría si invocásemos setQuestionsPublishedRightNow(old => { return old + 1 }) en un lugar distinto? ¿Funcionaría igual de bien si estuviera fuera del .then?
- ¿Qué te parece la elección de questionsPublishedRightNow (número) para que React reaccione? ¿Podríamos incurrir en problemas de bucles infinitos de carga de datos si tomásemos otra elección?



Sube tus cambios al repositorio en un nuevo commit.



Rubén Montero @ruben.montero changed milestone to <a href="MSprint 4">MSprint 4</a> 3 weeks ago