

## 1. Backend: Desarrolla una solución para el siguiente problema con NodeJS / NET Core.

### Control Forestal

El equipo de control de incendios del parque forestal de Montserrat, quiere lanzar una serie de drones para sobrevolar el territorio y detectar fuentes de calor.

Para poder controlar el área donde se moverán, el número de drones que se quieren lanzar, y el recorrido que realizará cada uno de ellos, el equipo ha diseñado el siguiente sistema:

- Enviaremos todas las instrucciones a la vez.
- En la primera línea definiremos el área a sobrevolar. Para ello especificamos un rectángulo, por donde se podrán desplazar los drones.
- En la siguiente línea especificaremos el punto de inicio del dron y su dirección: "N", "E", "S" o "O". Por ejemplo especificando "0 0 N" mandaremos volar al dron abajo a la izquierda del rectángulo.
- En la siguiente línea programaremos las acciones que realizará el dron volando mediante una codificación por letras. Las letras posibles serán "L", "R" y "M". Mediante "L" y "R" giraremos el dron 90° hacia la izquierda y derecha respectivamente. Mediante la letra "M" mandaremos al dron desplazarse una posición hacia delante. Asumimos que avanzar una posición al norte de (x, y) se movería a (x, y+1).
- Volveremos a poner tantas parejas de líneas como queramos, siguiendo la especificación de las dos anteriores.
- Cada dron al finalizar registrará como salida de información su posición y dirección, del mismo modo que se especifica para su punto de inicio.
- Los drones acabarán volviendo a la base sin daño alguno.

Ejemplo:

Input:

```
5 5
3 3 E
L
3 3 E
MMRMMRMRRM
1 2 N
LMLMLMLMMLMLMLMLMM
```

Output:

```
3 3 N
5 1 E
1 4 N
```

## 2. Frontend:

Queremos crear una sencilla aplicación en js/react que nos permita añadir, editar o eliminar “tarjetas” con una información muy básica (título, descripción e imagen).

Dicha aplicación ha de cumplir los siguientes requisitos:

- Botón que muestra un modal con un sencillo formulario para introducir los datos de la nueva tarjeta:
  - Título (obligatorio)
  - Descripción (obligatorio)
  - Imagen (opcional y solamente la url, no es necesario que permita subir imágenes)
- Cada tarjeta a mostrar ha de visualizar la imagen en la parte superior, en caso de no haber indicado ninguna, ha de mostrar una imagen predeterminada por nosotros, justo debajo el título y descripción y al pasar el ratón sobre la tarjeta ha de mostrar unos botones que permitan editar o borrar la etiqueta.
- Tal como vayamos añadiendo tarjetas se han de ir visualizando directamente en la aplicación.

Ejemplo:



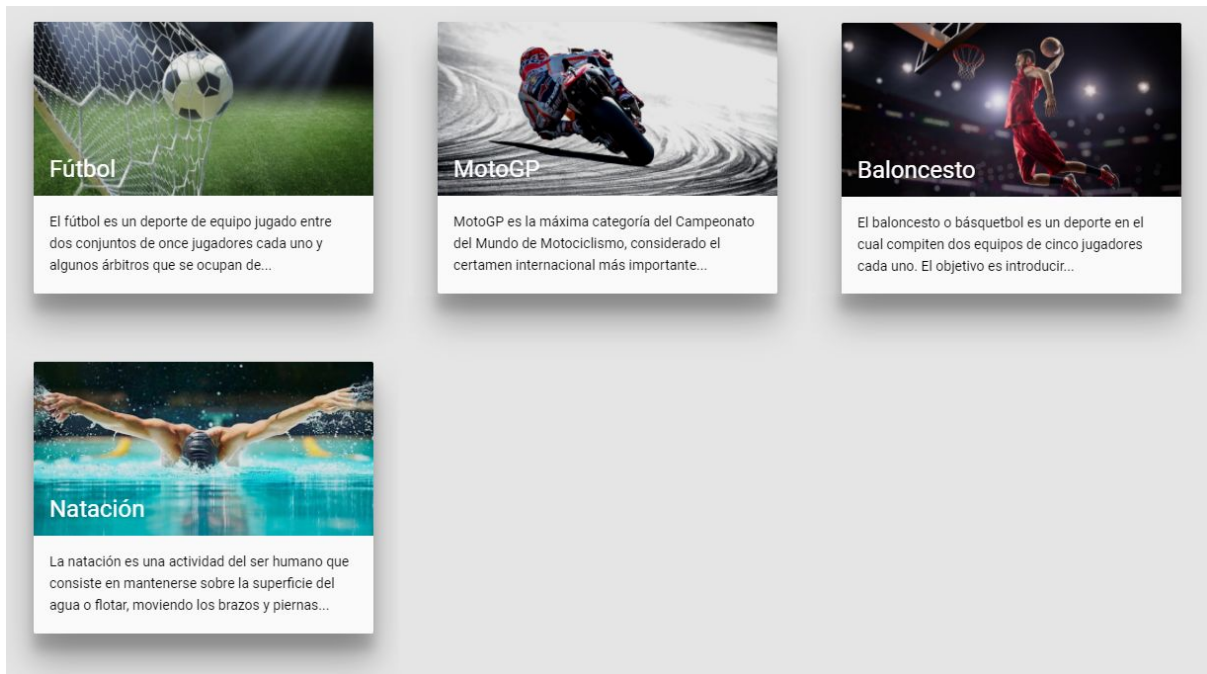
**Nueva tarjeta**

Título

Descripción

Imagen (Url)

Añadir



#### Extras:

- Almacenar las tarjetas en localStorage para que no se pierdan si cerramos o recargamos la aplicación.
- Añadir botones que nos permitan ordenar las tarjetas por título asc/desc o bien por creación asc/desc
- Que se visualice correctamente en mobile y desktop.

#### Nota:

Además del correcto funcionamiento, se valorará la calidad y legibilidad del código, la estructura de ficheros, así como las pruebas realizadas e instrucciones claras para la ejecución de las aplicaciones (readme.md...).