

## NodeJS: API de tablero Kanban– #1

Implemente un servicio REST que exponga el extremo de /boards.

Cada producto es una entrada JSON con las siguientes claves:

- id: El ID de recordatorio único. (Entero)
- title: El título del elemento TODO. (Cuerda)
- stage: La etapa actual del elemento (Entero)

Este es un ejemplo de un objeto JSON de recordatorio:

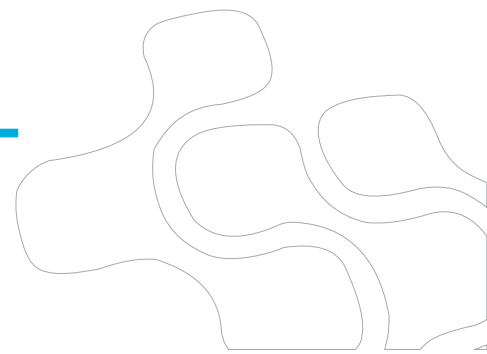
```
{  
  "id": 2,  
  "stage": 1,  
  "title": "Create a new project"  
}
```

Se proporciona la implementación del modelo y es de solo lectura.

Debe permitir administrar la colección de artículos en un Tablero Kanban en la etapa actual de la siguiente manera:

PUBLICAR solicitud a /boards:

- crea un nuevo elemento en el tablero
- espera un objeto de elemento JSON sin la propiedad id y stage como la carga útil del cuerpo. Puede suponer que el objeto dado siempre es válido.
- agrega el objeto dado que contiene el elemento al tablero y le asigna una identificación entera única. El primer elemento creado debe tener una identificación de 1, el segundo, 2, y así sucesivamente.
- Cada elemento recién creado debe tener una etapa de 1.
- Puede suponer que la propiedad del escenario nunca se pasará.
- El código de respuesta es 201 y el cuerpo de la respuesta es el objeto del elemento del tablero kanban creado.



PUT request to `/boards/:id`

- acepta la propiedad del escenario que contiene el nuevo escenario como la carga útil del cuerpo
- actualiza la etapa del artículo para la identificación del artículo pasado
- Si el valor de etapa pasado no es 1, 2 o 3, devuelve el código de estado 400 sin ningún requisito en el cuerpo de la respuesta.
- Si el valor de etapa es 1, 2 o 3, el código de respuesta es 200, con el elemento actualizado como cuerpo de respuesta.
- Puede suponer que el ID del elemento pasado siempre existe.

Debe completar el proyecto dado para que pase todos los casos de prueba al ejecutar las pruebas *unitarias proporcionadas*. El proyecto por defecto soporta el uso de la base de datos SQLite3.

## Ejemplos de solicitudes y respuestas

POST **solicitud de** `/boards`

Cuerpo de la solicitud:

```
{
  "title": "Create a new project"
}
```

El código de respuesta es 201 y, cuando se convierte a JSON, el cuerpo de la respuesta es:

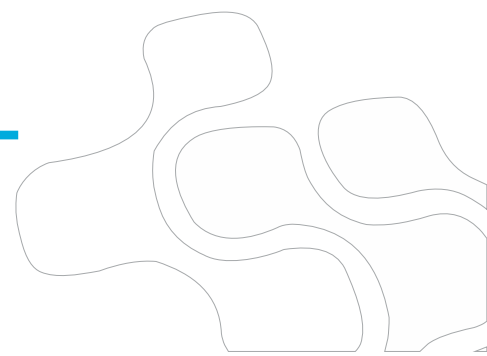
```
{
  "id": 1,
  "stage": 1,
  "title": "Create a new project"
}
```

Esto agrega un nuevo objeto a la colección con las propiedades dadas, id 1, con el escenario establecido en 1.

PUT **solicitud de** `/boards/1`

Cuerpo de la solicitud:

```
{
```



```
"stage": 2  
}
```

El código de respuesta es 200, la etapa del elemento 1 se actualizó a 2 y el objeto del elemento actualizado se envía en el cuerpo de la respuesta:

```
{  
  "id": 1,  
  "stage": 2,  
  "title": "Create a new project"  
}
```

PUT **solicitud de**/boards/1

Cuerpo de la solicitud:

```
{  
  "stage": 4  
}
```

El código de respuesta es 400 sin requisitos para el cuerpo de respuesta.

