

## Prueba técnica para front-end de Tagger



### Objetivo de la prueba

Conocer tus habilidades como programador basándonos en tareas a las que te enfrentarás a diario, así como tus conocimientos en tecnologías front-end que se utilizan en el proyecto.

### Ejercicio

Siéntete libre de proponer tu solución como creas conveniente, centrándote en las tareas y requisitos que se piden. Se recomienda leer detenidamente todo el ejercicio antes de proceder a su ejecución.

A continuación te detallamos los requisitos de la prueba que consiste en dos partes:

#### Primera Parte

1. Crea tu proyecto en Angular añadiendo enrutamiento (a ser posible la última versión, Angular 11). Tendrás que crear el siguiente componente con la siguiente firma:

```
<app-add-codes (codes)="checkArticleCodes (codes) "></app-add-codes>
```

Añadir artículos por código ⓘ

Ej.: 80265472, 98564528...

AÑADIR

Se trata de un buscador por código (input type="text") en el que insertas códigos de artículos y al pulsar el botón *Añadir* se realiza una validación y se procede a su búsqueda.

Este buscador aparecerá en la página principal. El icono de información (al lado del título) mostrará un 'tooltip' con información del buscador.

2. Tendrás que añadir validación ya que los códigos sólo pueden tener el siguiente formato:
  - 12345678: serie de 8 dígitos comprendidos del 0-9.
  - 000.000.00: serie de 8 dígitos comprendidos de 0-9 separados por puntos.

Se deben poder introducir un código o más códigos separados por comas.  
Ejemplos:

12345678, 111.222.33 formato válido  
ab123, 11.222.33, 55 formato inválido

3. Una vez validados, deberás parsearlos para que cada código tenga el formato 00000000 (si tiene puntos deberás quitarlos).
4. Mediante un evento deberás comunicarte con su componente padre y enviarle la lista con los códigos parseados:

```
(codes)="checkArticleCodes (codes) "
```

### Segunda Parte

5. Crea el mockup de un servicio API Rest que al realizar una petición GET devuelva el siguiente array:

```
["40424224", "00447350", "30349678", "60454955"]
```

Para ello puedes utilizar <https://www.mockable.io/>.

6. En tu proyecto Angular, crea el siguiente servicio Http para realizar la conexión con la API:

```
getArticleCodes(): Observable<HttpResponse<string[]>>
```

7. En el componente padre desarrolla la función que compruebe si los códigos que se han introducido existen en el array de códigos de la API y que genere un array con los artículos que sí se encuentran en el array del servicio y otro con los que no:

```
checkArticleCodes (codes: string[])
```

8. Añade otra página al proyecto con su correspondiente ruta.
9. Una vez tengamos los arrays anteriores, dentro del mismo método de *checkArticleCodes(codes: string[])*, se redirigirá a esta otra página mostrando los códigos de la siguiente manera:

ARTÍCULOS VÁLIDOS: XXXXXXXX,...  
ARTÍCULOS INVÁLIDOS: XXXXXXXX,...

Puedes recoger la información en el componente de la manera que creas más oportuna: mediante la ruta, un servicio, comunicación entre padre-hijo, etc.  
Para esta página puedes añadir los estilos como prefieras.

### **A tener en cuenta**

El ejercicio deberá ser entregado mediante repositorio de Github.  
Puedes utilizar librerías de Angular Material, Bootstrap o la que creas conveniente.

Se valorarán como extras:

- buenas prácticas.
- maquetación según el diseño aportado.
- correcto uso de tipado.
- documentación del código.