

IISI-2 IS: Simulacro de examen de laboratorio

Enunciado

Una vez se ha puesto en marcha la primera versión de DeliverUS, los inversores han solicitado la inclusión de una nueva funcionalidad que consiste en ofrecer a los propietarios la posibilidad de promocionar sus restaurantes. Cada propietario sólo podrá promocionar uno de sus restaurantes.

(Validaciones)

Un propietario podrá promocionar un restaurante de dos maneras distintas:

(Requisiciones)

- ① • En el formulario de creación de restaurante. Por defecto, se seleccionará la opción de no promocionado. Si el propietario indica que el nuevo restaurante debe estar promocionado, pero ya existían restaurantes promocionados del mismo propietario, al pulsar el botón Save se mostrará un error y no se creará el restaurante.
switch →
- ② • En la pantalla de "Mis restaurantes", mediante un botón mostrado junto a cada restaurante, que permitirá mediante su pulsación promocionar el restaurante en cuestión. Si el propietario pulsa el botón para promocionar un nuevo restaurante y ya existían otros restaurantes promocionados del mismo dueño, se procederá a promocionar el restaurante indicado y se marcará como "no promocionado" el restaurante que lo fuese anteriormente. La aplicación debe pedir confirmación al propietario cuando se pulse el botón; utilice para ello el componente suministrado ConfirmationModal, similar al componente DeleteModal utilizado en clase.

Además, los restaurantes promocionados aparecerán siempre al principio de los listados de restaurantes que se le presentan tanto a los propietarios como a los clientes. Además de presentarse al principio, los restaurantes promocionados deben destacarse visualmente, por lo que aparecerá una etiqueta de texto ¡En promoción! con el color principal de la marca.

Ejercicio 1

Realice todos los cambios necesarios en el proyecto de backend para implementar el nuevo requisito.

Ejercicio 2

Realice todos los cambios necesarios en el proyecto de frontend para implementar el nuevo requisito.

Pantallas

My Restaurants

- 100 MONTADITOS**
100 montaditos
Una forma divertida y variada de disfrutar de la comida.
Un lugar para compartir experiencias y dejarse llevar por...
Shipping: 1.50€ ¡En promoción!
Edit **Delete** **Promote**
- Casa Félix**
Cocina Tradicional
Shipping: 2.50€
Edit **Delete** **Promote**
- 0 productos**
0 productos
Shipping: 1.50€
Edit **Delete** **Promote**
- 30 productos**
1000 productos
Shipping: 1.50€
Edit **Delete** **Promote**
- 50 productos**
1000 productos
Shipping: 1.50€
Edit **Delete** **Promote**

My Restaurants **Control Panel** **Profile**

My Restaurants

Casa Félix
Cocina Tradicional

Please confirm action
Other promoted restaurant, if any, will be unpromoted

X Cancel **✓ Confirm**

- 100 MONTADITOS**
100 montaditos
Una forma divertida y variada de disfrutar de la comida.
Un lugar para compartir experiencias y dejarse llevar por...
Shipping: 1.50€ ¡En promoción!
Edit **Delete** **Promote**
- 30 productos**
1000 productos
Shipping: 1.50€
Edit **Delete** **Promote**
- 50 productos**
1000 productos
Shipping: 1.50€
Edit **Delete** **Promote**

My Restaurants **Control Panel** **Profile**

Create Restaurant

http://prueba.es
Shipping costs:
10
Email:
prueba@prueba.es
Phone:
123456789
Spanish
Logo:

Hero image:

Is it promoted?
 promoted - You can only promote one restaurant at a time
Save

My Restaurants **Control Panel** **Profile**

BACKEND
 Migrations y modelo Restaurant → Añadir atributo promoted, con valor por defecto FALSE
 Validations Restaurant → Crear método que comprueba si hay un solo restaurante promocionado y lo incluimos en los checks de create y update.
 Restaurant Controller → Añadimos el ordenamiento pedido tanto a index como a indexOwner (vde TaF)
 Añadimos un método para actualizar la promoción → promote
 Lo que hace este método es comprobar si hay algún restaurante promocionado, si lo hay, le quita la promoción y se lo pone al restaurante que queremos promocionar.
 Restaurant Routes → Añadimos una ruta específica para la promoción → /restaurants/:restaurantId/promote.patch()

FRONTEED
 Restaurant Endpoints → Añadimos el endpoint de la nueva ruta → promote
 Configuration Model → Tenemos que crear una ventana de confirmación para el momento de promocionar
 ① Create Restaurant Screen
 ↳ ① En los valores iniciales añadimos promoted: false, como valor por defecto.
 ↳ ② En el return final → Añadimos el switch para seleccionar la promoción del restaurante.
 ↳ ③ Al final añadimos el estilo que va a tener el switch.

② Restaurant Screen
 ↳ ① Añadimos constante restaurantToBePromoted → useState(null)
 ↳ ② Añadimos el texto !En promoción! en render Restaurant, solo si está en promoción
 ↳ ③ Añadimos botón promote en render Product
 ↳ ④ Añadimos un método para promocionar un restaurante que utilice el Endpoint creado (que está conectado con el Backend a través de la ruta)
 ↳ ⑤ Añadimos la ventana de promoción en el return final
 ↳ ⑥ Geamos el estilo del texto !En promoción! (badge)

Introducción

Este repositorio incluye el backend completo (carpeta `DeliverUS-Backend`) y el frontend de `owner` (carpeta `DeliverUS-Frontend-Owner`). Servirá como base para realizar el examen de laboratorio de la asignatura.

Preparación del entorno

a) Windows

- Abra un terminal y ejecute el comando `npm run install:all:win`.

b) Linux/MacOS

- Abra un terminal y ejecute el comando `npm run install:all:bash`.

Ejecución

Backend

- Para **rehacer las migraciones y seeders**, abra un terminal y ejecute el comando

```
npm run migrate:backend
```

- Para **ejecutarlo**, abra un terminal y ejecute el comando

```
npm run start:backend
```

Frontend

- Para **ejecutar la aplicación frontend de owner**, abra un nuevo terminal y ejecute el comando

```
npm run start:frontend:owner
```

Depuración

- Para **depurar el backend**, asegúrese de que **NO** existe una instancia en ejecución, pulse en el botón `Run and Debug` de la barra lateral, seleccione `Debug Backend` en la lista desplegable, y pulse el botón de `Play`.
- Para **depurar el frontend**, asegúrese de que **EXISTE** una instancia en ejecución del frontend que deseé depurar, pulse en el botón `Run and Debug` de la barra lateral, seleccione `Debug Frontend` en la lista desplegable, y pulse el botón de `Play`.

Test

- Para comprobar el correcto funcionamiento de backend puede ejecutar el conjunto de tests incluido a tal efecto. Para ello ejecute el siguiente comando:

```
npm run test:backend
```

Advertencia: Los tests no pueden ser modificados.

BACKEND

① MODELS Y MIGRATIONS

↳ Restaurant y create_restaurant → Añadimos la propiedad 'promoted' que será valor BOOLEANO

② VALIDATION

↳ RestaurantValidation → creamos check que comprueba si hay algún restaurante promocionado. Además, añadimos el check anterior de 'promoted'

③ CONTROLLERS

↳ RestaurantController → Añadimos el método promoted para promocionar el restaurante.

Modificamos los métodos index e indexOwner para realizar el ordenamiento pedido.

④ ROUTES

↳ RestaurantRoutes → Añadimos una nueva ruta para el método create

FRONTEND

① ENDPOINTS

↳ RestaurantEndpoint → Añadimos el endpoint de la nueva ruta.

② SCREENS

↳ RestaurantsScreen → Añadimos el botón para promocionar junto con la lógica correspondiente.

También, el texto que nos indica el restaurante se promociona.

Por último, la lógica para la asignación del modelo de confirmación.

↳ CreateRestaurant → Añadimos el switch correspondiente.