

IISSI-2 IS: Simulacro de frontend 2025

DeliverUS

La documentación de DeliverUS se puede encontrar en: <https://github.com/IISSI2-IS-2025>

Introducción

Este repositorio incluye el backend completo (carpeta `DeliverUS-Backend`) y el frontend de owner (carpeta `DeliverUS-Frontend-Owner`).

Fijar restaurantes en listas. Enunciado

Tras el lanzamiento de DeliverUS, los inversores han pedido que se incluya una nueva característica, de manera que permita a los propietarios que se puedan fijar uno o más restaurantes en las listas de restaurantes.

Un propietario puede fijar (English: *to pin*) un restaurant de dos maneras diferentes:

- En el formulario para crear un nuevo restaurante. Por defecto, el restaurant no estará fijado., pero el propietario puede cambiar la propiedad. Para ello, se debe utilizar un control `Switch` para trabajar con una propiedad llamada `pinned`. Si el `Switch` está activo, el restaurant debe crearse como fijado. El backend espera que la propiedad `pinned` sea un booleano y opcional. Si la propiedad no está presente, el restaurant se entiende que no está fijado.
- En la lista de "My Restaurants" habrá un icono que actuará como un botón y se visualizará a la derecha de cada restaurant. Al pulsar este ícono, el restaurant podrá pasar de fijado a no fijado y viceversa.. La aplicación debe preguntar por una confirmación cuando el ícono sea pulsado. Utilice el componente `ConfirmationModal` proporcionado, similar al componente `DeleteModal` de la misma pantalla. El sistema debe informar al usuario que el restaurant ha sido fijado o no.

Finalmente, los restaurantes fijados deben aparecer al inicio de las listas presentadas al propietario, ordenadas por la fecha en las que fueron fijadas (las más antiguas, primero), seguidos por los restaurantes que no están fijados.

Tareas a realizar sobre el frontend

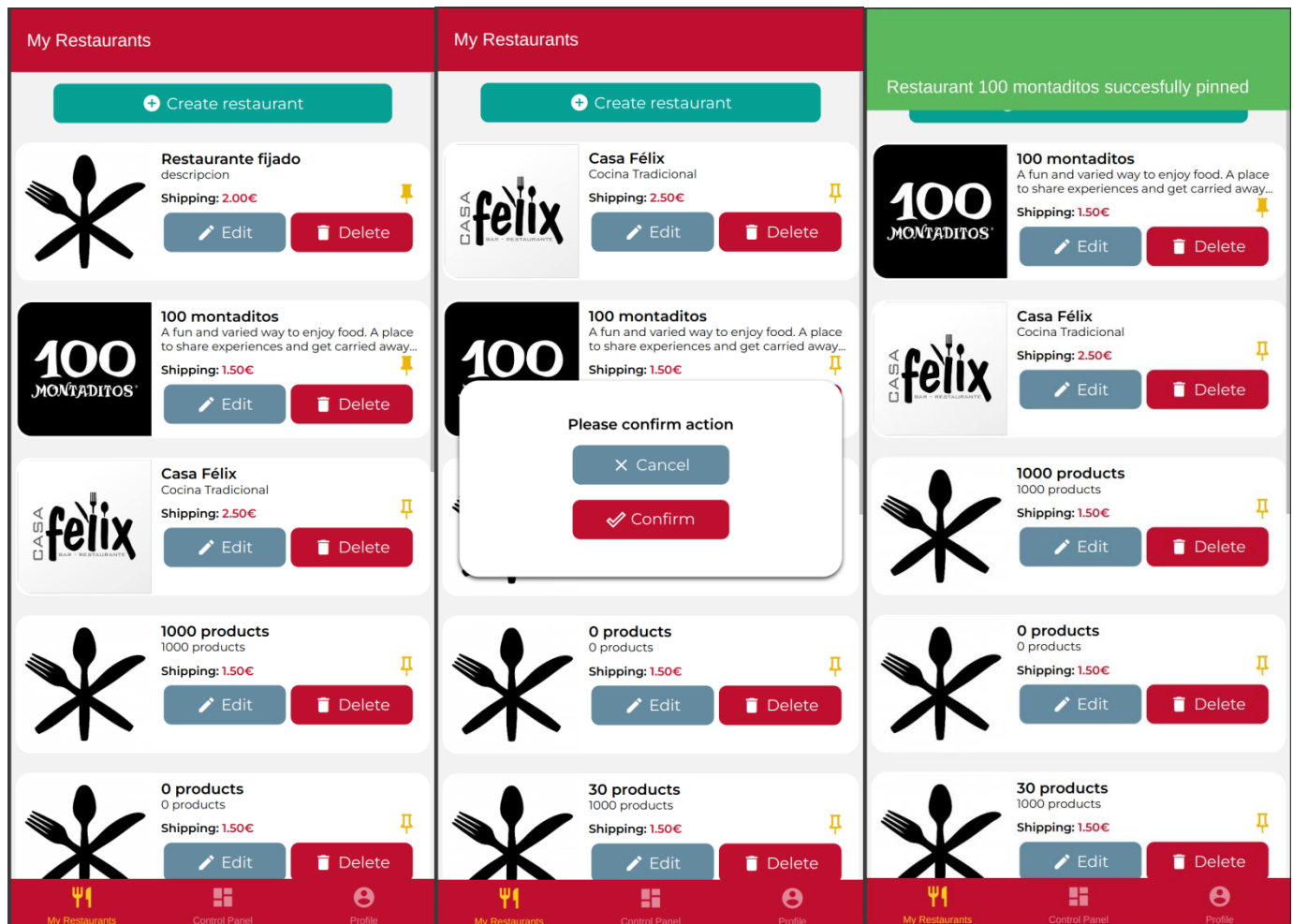
Sobre el backend: el backend se ha adaptado para implementar el nuevo requisito, de manera que ofrece la ruta: `PATCH /restaurants/:restaurantId/togglePinned` y los restaurantes tienen una nueva propiedad llamada `pinnedAt`. Además, el cuerpo de `POST /restaurants/` incluye una propiedad `pinned` que puede ser `true` o `false`, y los controladores a cargo de los listados se han adaptado al nuevo requisito.

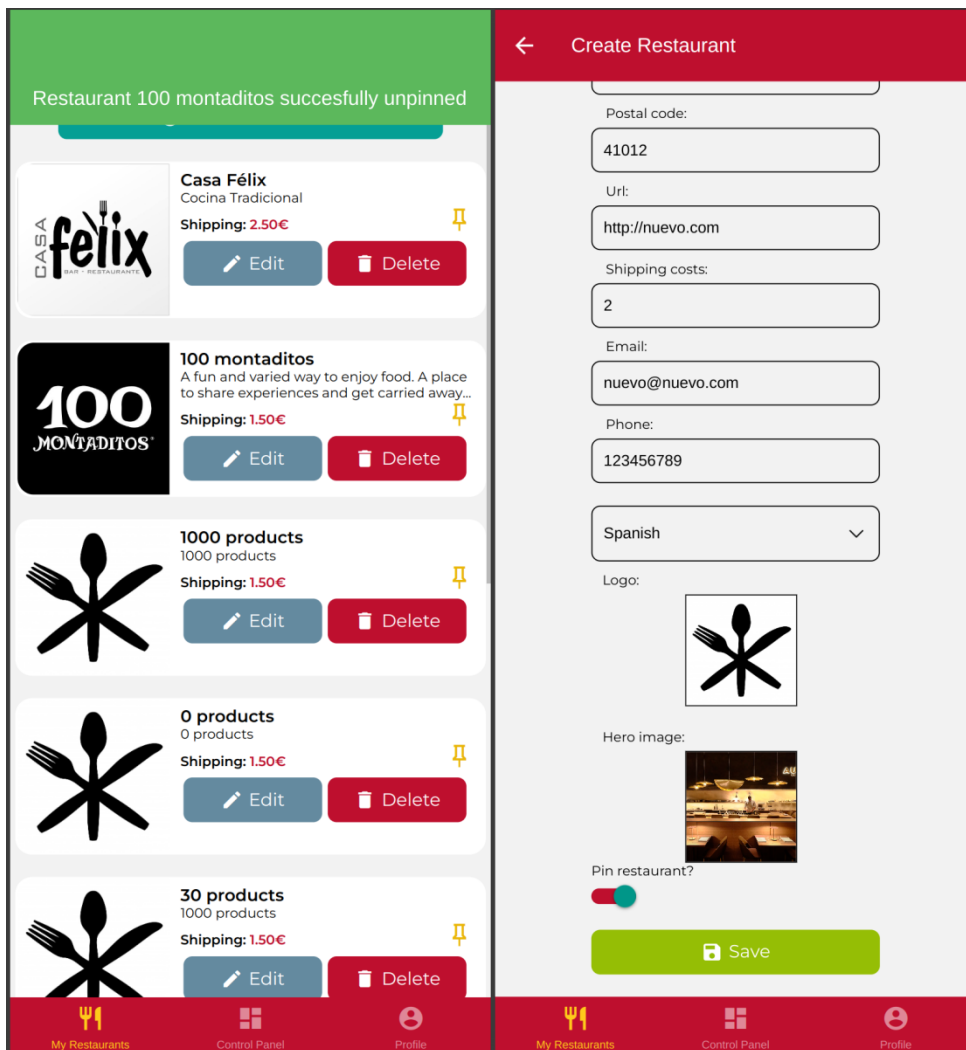
Tarea: Realiza todos los cambios necesarios en el proyecto de frontend para implementar el nuevo requisito.

Puedes visualizar el ícono propuesto de “pin” con:

```
<MaterialCommunityIcons  
  name={item.pinnedAt ? 'pin' : 'pin-outline'}  
  color={GlobalStyles.brandSecondaryTap}  
  size={24}  
/>
```

La interfaz de usuario debe diseñarse de acuerdo a las siguientes pantallas:





Resumen de los requisitos

- RF1. Capacidad para crear un nuevo restaurante, fijado o no.
- RF2. Capacidad para cambiar dicha propiedad de un restaurant que ya está creado.
- RF3. Listar los restaurantes en el orden descrito: primero los restaurantes fijados ordenados por fecha (los restaurantes que fueron fijados antes, primero), seguidos de los restaurantes no fijados.

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 desee 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.

Problemas con los puertos

En ocasiones, los procesos de backend o frontend, con o sin depuración, pueden quedarse bloqueados sin liberar los puertos utilizados, impidiendo que puedan ejecutarse otros procesos. Se recomienda cerrar y volver a iniciar VSC para cerrar dichos procesos.

Procedimiento de entrega

1. Borrar las carpetas **node_modules** de backend y frontend y **.expo** del frontend.
2. Crear un ZIP que incluya todo el proyecto. **Importante: Comprueba que el ZIP no es el mismo que te has descargado e incluye tu solución**
3. Avisa al profesor antes de entregar.
4. Cuando el profesor te dé el visto bueno, puedes subir el ZIP a la plataforma de Enseñanza Virtual. Espera a que la plataforma te muestre un enlace al ZIP antes de pulsar el botón de aceptar.