

EXERCICI AJAX RESTAURANTE

CONTEXTUALIZACIÓ

Se nos pide que realicemos una aplicación para la gestión de pedidos de un restaurante, donde el camarero irá con un teléfono cogiendo los pedidos en las mesas y enviará esos pedidos a la barra para bebidas y a la cocina para los platos a cocinar.

En la barra y la cocina habrá una cola de bebidas y platos a preparar cuando el barman o cocinero termine de preparar el producto, lo marcará como servido.

Una vez creados los ficheros maestros de Mesas, bebidas y Platos. Vamos a crear una comanda nueva que tendrá la siguiente vista.

- **Alta Comandas:**

Restaurante

Nueva Comanda

Nombre

Comensales

Mesas

1

3

4

5

6

Bebidas

Borrar

Bebida

Cantidad

Agua Mineral Peq.

Agua Mineral Gran

Cassalla Tranbolica

Mojito

Xupito de leja

Platos

Borrar

Plato

Cantidad

Primeros

Gazpacho

Macarrones a la boloñesa

Ensalada Cesar

Patatas Bravas

Ensalada mixta

Segundos

Entrecote a la pimienta

Foie gras de pato al Romerilla

Confit de pato al Romerier

Postres

Fruta del Tiempo

Profiteroles

Helado 2 bolas

Notas

Confirmar

Se debe de tener en cuenta las siguientes consideraciones:

- Se debe de validar el nombre, los comensales, pero las notas no son obligatorias.
- Las mesas se mostrar la numeración que se ha introducido, pero se guardará el id del registro de la mesa. Cuando se seleccione el una mesa, esta cambiará de color a rojo con las clases “`mt-2 btn btn-danger p-3`”
- Cuando se seleccione una mesa se obtendrá el numero de comensales y las descripción asociada y se mostrará en la parte inferior.
- Igual en bebidas como en platos, los mostraremos con botones cada una de ellos, si se selecciona un botón este subirá a la lista de peticiones y se guardaran la información en un array que se mantendrá con los mismos datos que la tabla que se puesta en pantalla. Si se vuelve a seleccionar el mismo boton no se creará una nueva linea sino se aumentará la cantidad pedida en 1.
- Los platos se dividirán en el orden de: Primeros, Segundos y Postre.
- Una vez realizado toda la comanda se enviará al servidor mediante la siguiente estructura de datos.
- Hay que tener en cuenta que no es necesario rellenar el usuario, ya que por defeto el sistema ya lo reconoce, los estados deben de ser “Pendiente” y la fecha Entrada y la de Salida no se informarán en esta petición.
- Para el correcto funcionamiento no se debe de refrescar la página en ningún caso, utilizar el DOM, capturar los errores que nos pueda dar el API y una vez grabada la comanda enviar el flujo a el listado de comanda.

```
const bebidasSchema = mongoose.Schema({
  nombre: {
    type: String,
    required: true,
    min: 4,
    max: 60,
    trim: true
  },
  cantidad: {
    type: Number,
    required: true,
    min: 0
  },
  precio: {
    type: Number,
    required: true,
    min: 0
  },
  // estado : Pendiente, Servido.
  estado: {
    type: String,
    required: true,
  }
})
```

```
const platosSchema = mongoose.Schema({
  nombre: {
    type: String,
    required: true,
    minlength: 4,
    maxlength: 60
  },
  cantidad: {
    type: Number,
    required: true,
    min: 0
  },
  precio: {
    type: Number,
    required: true,
    min: 0
  },
  // estado : Pendiente, Servido.
  estado: {
    type: String,
    required: true,
  },
})
```

```
const comandasSchema = mongoose.Schema({
  nombre: {
    type: String,
    required: true,
    minlength: 4,
    maxlength: 60,
    trim: true
  },
  comensales: {
    type: Number,
    required: true,
    min: 1
  },
  // estado : Pendiente, Servido, Cobrado.
  estado: {
    type: String,
    required: true,
  },
  fechaEntrada: {
    type: Date,
    default: Date.now
  },
  fechaSalida: {
    type: Date,
  },
  // Vinculación de comanda con mesa
  mesa: {
    type: mongoose.Schema.Types.ObjectId,
    required: true,
    ref: 'mesa'
  },
  // Vinculación de comanda con usuario
  user: {
    type: mongoose.Schema.Types.ObjectId,
    required: true,
    ref: 'user'
  },
  // bebidas
  bebidas: [bebidasSchema],
  // Comidas
  platos: [platosSchema],
  notas: [
    type: String
  ]
})
```

Los ENDPOINTS serán :

- GET → ["https://restaurante.serverred.es/api/mesas"](https://restaurante.serverred.es/api/mesas)
- GET → ["https://restaurante.serverred.es/api/bebidas"](https://restaurante.serverred.es/api/bebidas)
- GET → ["https://restaurante.serverred.es/api/platos"](https://restaurante.serverred.es/api/platos)
- POST → <https://restaurante.serverred.es/api/comandas>

- **Listado de Comandas:**

en este apartado se mostrarán solo las comandas pendiente, una vez que el cliente pida la cuenta la comanda no estará en el listado.

En el listado el camarero solo podrá acceder para la bebida o los platos pedidos y la forma de la página será la siguiente:

		Nombre	Mesa	Com.	Camarero	H Entrada
+Bebidas	+Platos	Pepe	1	4	David Crespo Jorda	18:38
+Bebidas	+Platos	Maria	4	2	David Crespo Jorda	23:27

DWEC - Restaurante 2022 - David Crespo

Los ENDPOINTS serán los siguientes:

- GET → <https://restaurante.serverred.es/api/camareros>
- GET → ["https://restaurante.serverred.es/api/mesas"](https://restaurante.serverred.es/api/mesas)
- GET → ["https://restaurante.serverred.es/api/comandas"](https://restaurante.serverred.es/api/comandas)

En comandas tendremos dos opciones, la de editar la comanda de bebida o la de platos.

- **Comanda de Bebidas** , tendrán la siguiente vista:

Añadir Bebida

Nombre: **Pepe** Comensales: **4** Mesa: **1** Camarero: **David Crespo Jorda**
H.entrada: **18:38**

Bebidas

Borrar	Bebida	Cantidad
Borrar	Caña	2
Borrar	Tanque 750 cl	1
Borrar	Agua Mineral Gran.	1
Borrar	Protos Reserva 17 70 cc	1

Agua Mineral Peq. Agua Mineral Gran. Cassalla Tranbolica Mojito Xupito de leja

Notas

los postres los pedirán mas tarde

Confirmar

Se podrán añadir nuevas bebidas o eliminar las que hay siempre que en su estado no sea “Servido”

- **Comanda de Bebidas** , tendrán la siguiente vista:

Restaurante Home Cocina Barra Sala Gestión David Crespo Jorda

Añadir Platos

Nombre: **Maria** Comensales: **2** Mesa: **4** Camarero: **David Crespo Jorda**
H.entrada: **23:27**

Platos

Borrar	Plato	Orden	Cantidad
<button>Borrar</button>	Gazpacho	Primero	1
<button>Borrar</button>	Ensalada Cesar	Primero	1
<button>Borrar</button>	Patatas Bravas	Primero	1
<button>Borrar</button>	Foie gras de pato al Romerilla	Segundo	1

Primeros

Gazpacho Macarrones a la boloñesa Ensalada Cesar Patatas Bravas Ensalada mixta

Segundos

Entrecote a la pimienta Foie gras de pato al Romerilla Confit de pato al Romerier

Postres

Fruta del Tiempo Profiteroles Helado 2 bolas

Notas

Confirmar

DWEC - Restaurante 2022 - David Crespo

ENDPOINTS necesarios :

- GET → <https://restaurante.serverred.es/api/camareros>
- GET → <https://restaurante.serverred.es/api/mesas>
- GET → <https://restaurante.serverred.es/api/platos>
- PUT → ["https://restaurante.serverred.es/api/comandas"](https://restaurante.serverred.es/api/comandas)

Cabe destacar que para pasar la comanda del listado de la comanda se puede guardar toda la comanda en el LocalStorage y luego recuperar esa información en añadir bebida o platos.