

|  |
| --- |
| **[PROYECTO FIN DE CICLO]** |
| Juan Pedro Martínez Granados |

Contenido

[1. AMBITO DE PROYECTO 3](#_Toc200906816)

[1.1 ENUNCIADO DEL PROBLEMA 3](#_Toc200906817)

[1.2 IDEA DE SOLUCIÓN 3](#_Toc200906818)

[1.3 CARACTERÍSTICAS PRINCIPALES 3](#_Toc200906819)

[1.3 TECNOLOGÍAS UTILIZADAS 4](#_Toc200906820)

[1.4 LIMITACIONES Y ALCANCE 4](#_Toc200906821)

[1.5 REQUISITOS TÉCNICOS 4](#_Toc200906822)

[1.6 IMPACTO ESPERADO 5](#_Toc200906823)

[2. ESTUDIO DE VIABILIDAD 5](#_Toc200906824)

[2.1 DISEÑO Y PROGRAMACIÓN DE LA APLICACIÓN 5](#_Toc200906825)

[2.1.1 TÉCNICO 5](#_Toc200906826)

[2.1.2 ECONÓMICO 5](#_Toc200906827)

[HP 250 G10 AD1K3ET 6](#_Toc200906828)

[Lenovo V15 G4 IRU 83A100BKSP 7](#_Toc200906829)

[MSI Thin 15 B13VE-2446ES 8](#_Toc200906830)

[2.2 DESPLIEGUE DE LA APLICACIÓN 8](#_Toc200906831)

[2.2.1 TÉCNICO 8](#_Toc200906832)

[2.2.2 ECONÓMICO 9](#_Toc200906833)

[Fujitsu PRIMERGY TX1310 M5 9](#_Toc200906834)

[Conexión a internet 13](#_Toc200906835)

[Conexión a internet 14](#_Toc200906836)

[Conexión a internet 14](#_Toc200906837)

[2.3 DIAGRAMA DE GANTT 15](#_Toc200906838)

[3. ANÁLISIS 15](#_Toc200906839)

[3.1 ENTREVISTA AL CLIENTE 15](#_Toc200906840)

[3.1 .1 PRIMERA ENTREVISTA: TOMA DE CONTACTO CON DON PEDRO SANCHEZ 16](#_Toc200906841)

[3.1 .2 SEGUNDA ENTREVISTA. FUNCIONALIDAD DE OLVIDO DE CONTRASEÑA Y QUE LOS CLIENTES PUEDAN REGISTRARSE 19](#_Toc200906842)

[3.1 .3 TERCERA ENTREVISTA (PRESENTACIÓN DE PROTOTIPOS) 19](#_Toc200906843)

[3.1 .4 CUARTA ENTREVISTA (DISEÑO Y FUNCIONALIDADES) 27](#_Toc200906844)

[3.2 DIAGRAMA DE FLUJO DE DATOS 30](#_Toc200906845)

[4. DISEÑO 30](#_Toc200906846)

[4.1 DIAGRAMA ENTIDAD RELACIÓN 30](#_Toc200906847)

[4.2 DISEÑO RELACIONAL 31](#_Toc200906848)

[4.3 PASO A TABLAS 31](#_Toc200906849)

[4.4 PROCESOS 32](#_Toc200906850)

[5. CODIFICACIÓN 36](#_Toc200906851)

[5.1 ARCHIVOS MÁS IMPORTANTES 36](#_Toc200906852)

[5.2 WEBSOCKET 37](#_Toc200906853)

[6. PRUEBAS 39](#_Toc200906854)

[7. MANUALES 40](#_Toc200906855)

[7.1 MANUAL DEL ADMINISTRADOR 41](#_Toc200906856)

[7.2 MANUAL DEL COCINERO 54](#_Toc200906857)

[7.4 MANUAL DEL CAMARERO 57](#_Toc200906858)

[7.4 MANUAL DEL USUARIO 58](#_Toc200906859)

# 

# 1. AMBITO DE PROYECTO

## 1.1 ENUNCIADO DEL PROBLEMA

A lo largo de los años he visto como la tecnología en empresas hosteleras ha dado pasos agigantados. A día de hoy hay aplicaciones web que tienen por finalidad ahorrarle tiempo al camarero a la hora de coger la comanda en una mesa. Mi objetivo es crear una aplicación para que el camarero que te atienda no tenga necesidad de ir a la mesa y los clientes puedan pedir su comanda más rápido sin necesidad de esperar al camarero. Con esta aplicación nos ahorraríamos tiempo en llamar al camarero.

## 1.2 IDEA DE SOLUCIÓN

La idea principal de mi proyecto es crear ÑAMQR, una aplicación que se encargará de la gestión de las comandas de un restaurante. El cliente cuando escanee el código QR entrará en el login de la página. Si el cliente está verificado podrá poner sus credenciales para saber los pedidos que ha hecho en el restaurante y también podrá ver la carta para hacer su pedido. Habrá otra opción que será la de pedir sin necesidad de registrarnos o de iniciar sesión.

Gracias a esto habrá más facilidad a la hora de efectuar la comanda, ya que la comanda de la bebida irá directamente a la barra y la comanda de la comida irá directamente a la cocina.

Con esta aplicación tampoco habría errores de apuntado de comandas en las mesas puesto que los clientes son los que eligen que quieren de la carta.

## 1.3 CARACTERÍSTICAS PRINCIPALES

Estas serán las funcionalidades más importantes de nuestra aplicación:

* El cliente podrá seleccionar de la carta el número de unidades que quiere de cada producto.
* Los pedidos serán en tiempo real.
* Se dará la posibilidad de pagar a través de la página web o directamente en barra.
* Gestionar si una mesa se cambia a otra mesa.
* La barra tendrá la opción de ver las comandas de cada mesa.
* La cocina verá la comida que ha pedido cada mesa.
* Gestión de pedidos: Añadiré un sistema para gestionar el estado del pedido utilizando websockets (debo informarme de ello).

Las futuras actualizaciones serán:

* El cliente podrá valorar la comida y la atención recibida.

## 1.3 TECNOLOGÍAS UTILIZADAS

Para el frontend utilizaré:

* Interfaz de usuario: **Tailwind CSS.**
* **JavaScript.**
* Generación de QR: Librerías **qrcode.react** o **QRCode.js**

Para el backend utilizaré:

* **Php.**
* Base de datos: **MysQl.**

Escaneo de códigos QR y pedidos en tiempo real:

* Comunicación en tiempo real: Utilizaré **websocket** para el estado de los pedidos.

## 1.4 LIMITACIONES Y ALCANCE

Esta aplicación tendrá las siguientes limitaciones:

* Los pedidos solo se podrán realizar hasta que la cocina se encuentre abierta.
* Aplicación no integrada en pedidos en barra.

## 1.5 REQUISITOS TÉCNICOS

* La aplicación deberá de ser accesible tanto para móviles como para tablets.
* Es necesario conexión a internet y un dispositivo para escanear el código QR de la mesa.

## 1.6 IMPACTO ESPERADO

* Reducción de espera de los clientes.
* Mayor eficiencia en la toma de comandas.
* Mayor organización en el negocio.
* Reducción de errores en la toma de comandas.

# 2. ESTUDIO DE VIABILIDAD

## 2.1 DISEÑO Y PROGRAMACIÓN DE LA APLICACIÓN

### 2.1.1 TÉCNICO

Para poder realizar la aplicación mencionada en el primer punto necesitaré un equipo que tenga mínimo estas características:

* Procesador: Intel core i7-11800 2.30 GHz.
* Memoria Ram: Mínimo 8 GB pero con 16GB será mejor.
* Disco duro: Para un solo negocio podrá valer un disco duro con poco almacenaje como por ejemplo 10Gb pero pondremos un disco duro con mayor almacenaje por si en un futuro la aplicación sigue hacia adelante.
* Para el sistema operativo utilizaremos Windows 10 u 11.
* El editor de código que utilizaré será Visual Studio Code.

### 2.1.2 ECONÓMICO

Las siguientes tablas muestran algunos presupuestos de mayor a menor preferencia, que reúnen todos los requisitos para poder realizar la aplicación.

|  |  |
| --- | --- |
| PRESUPUESTO 1 | |
| Producto | Precio |
| HP 250 G10 AD1K3ET   <https://pcgaming365.com/products/portatil-hp-250-g10-ad1k3et-intel-core-i7-1355u-16gb-1tb-ssd-15-6-win11> | 573,00€ |
| Sistema operativo | Incluido en el equipo de trabajo |
| Visual Studio | Gratis |

|  |  |
| --- | --- |
| PRESUPUESTO 2 | |
| Producto | Precio |
| Lenovo V15 G4 IRU 83A100BKSP   [https://pcgaming365.com/products/portatil-lenovo-v15-g4-iru-83a100bksp-intel-core-i7-1355u-16gb-512gb-ssd-15-6-win11](https://pcgaming365.com/products/portatil-lenovo-v15-g4-iru-83a100bksp-intel-core-i7-1355u-16gb-512gb-ssd-15-6-win11?variant=50986788585736&currency=EUR&utm_source=google&utm_medium=cpc&utm_campaign=google%2Bshopping&gad_source=1&gclid=Cj0KCQjwy46_BhDOARIsAIvmcwNV5X20qslf2QJTqxyus-GYQdVFrKKpoeACp-sOk3hpOUO5bLmiJ3IaAl70EALw_wcB) | 536,00€ |
| Sistema operativo | Incluido en el equipo de trabajo |
| Visual Studio | Gratis |

|  |  |
| --- | --- |
| PRESUPUESTO 3 | |
| Producto | Precio |
| MSI Thin 15 B13VE-2446ES   <https://www.mediamarkt.es/es/product/_portatil-gaming-msi-thin-15-b13ve-2446es-156-full-hd-intelr-coretm-i7-13620h-16gb-ram-512gb-ssd-geforce-rtxtm-4050-windows-11-home-1581109.html> | 949,00€ |
| Sistema operativo | Incluido en el equipo de trabajo |
| Visual Studio | Gratis |

## 2.2 DESPLIEGUE DE LA APLICACIÓN

### 2.2.1 TÉCNICO

Para realizar un buen despliegue de la aplicación y que pueda funcionar correctamente podríamos utilizar un servidor con estas características:

<https://pclocura.com/servidores/280800-lenovo-7d8ja045ea-thinksystem-st50-v2-servidor-2-tb-torre-intel-xeon-e-e-2324g-3-1-ghz-16-gb-ddr-0889488787258.html?gad_source=1&gclid=Cj0KCQjwy46_BhDOARIsAIvmcwN8laGUgaXmSQILL4CEy3liMueYtM_Va26-mpD_I5P9VWuz4u4ShqQaAinLEALw_wcB>

Si por el contrario queremos echar mano de un servidor en la nube podemos recurrir a varias páginas como <https://www.netlify.com/> , <https://vercel.com/> o <https://railway.com/> que nos ofrecen un plan gratuito hasta cierto punto dependiendo de las necesidades de la aplicación.

### 2.2.2 ECONÓMICO

Las siguientes tablas nos muestran presupuestos de mayor a menor preferencia, que reúnen los requisitos para poder hacer el despliegue y el correcto funcionamiento de la aplicación.

|  |  |
| --- | --- |
| PRESUPUESTO 1 | |
| Producto | Precio |
| Fujitsu PRIMERGY TX1310 M5 <https://www.senetic.es/product/VFYT1315SC010PL> | 923,90€ |

|  |  |
| --- | --- |
| PRESUPUESTO 2 | |
| Producto | Precio |
| Contratar Netlify para utilizar un servidor en la nube. | Dependiendo del plan ahí unos precios u otros. También dispone de una versión gratuita.  <https://www.netlify.com/pricing/> |

|  |  |
| --- | --- |
| PRESUPUESTO 3 | |
| Producto | Precio |
| Contratar Railwy para utilizar un servidor en la nube. | Dependiendo del plan ahí unos precios u otros. También dispone de una versión gratuita.  <https://railway.com/pricing> |

Aparte del presupuesto escogido, tendremos que usar un dominio web (a no ser que usemos un servidor local como XAMPP). Las mejores opciones que he encontrado son:

|  |
| --- |
| OPCIÓN 1 |
| Fuente de información: <https://www.ionos.es/domainshop/search> |

|  |
| --- |
| OPCIÓN 2 |
| Fuente de información: <https://www.arsys.es/dominios/buscar/restaurantebarato> |

|  |
| --- |
| OPCIÓN 3 |
| Fuente de información: <https://dinahosting.com/dominios/resultados?busquedaSimple=restaurantebarato> |

Hay que tener en cuenta en contratar una compañía para la conexión a internet para todos los dispositivos de la compañía. Precio (20€ de fibra + 6€ por cada camarero y 6€ para la tablet que habrá en cocina con las comandas).

También hay que añadir los costes del personal.

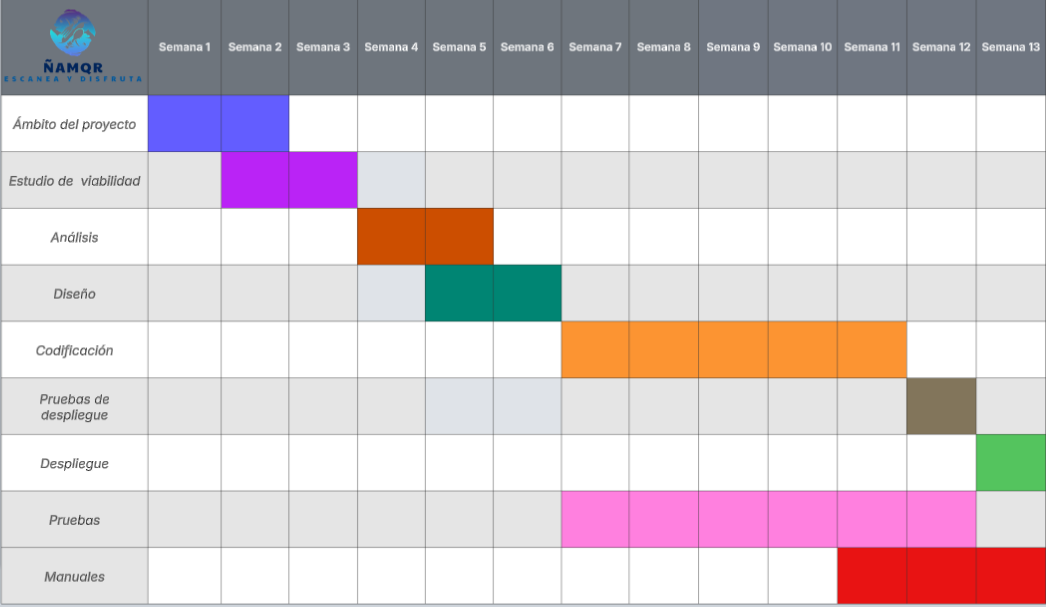
|  |  |
| --- | --- |
| PRESUPESTO 1 PARA JEFE DE COCINA | |
| Producto | Precio |
| XIOAOMI REDMI PAD SE  C:\Users\Juan Pedro\AppData\Local\Packages\MicrosoftWindows.Client.Core_cw5n1h2txyewy\TempState\ScreenClip\{5EE66D8C-460A-49FC-87DD-68A67B3ABA8D}.png  <https://allzone.es/tablets/782636-xiaomi-redmi-pad-se-11-8gb-ram-256gb-wifi-gris-6941812756447.html> | 157,43€ |
| Conexión a internet | Incluido en la compañía |

|  |  |
| --- | --- |
| PRESUPESTO 2 PARA JEFE DE COCINA | |
| Producto | Precio |
| SAMSUNG GALAXY TAB A9  C:\Users\Juan Pedro\AppData\Local\Packages\MicrosoftWindows.Client.Core_cw5n1h2txyewy\TempState\ScreenClip\{44BBAFE3-5787-4BFD-B4F3-94DBFFC57C45}.png  <https://www.pccomponentes.com/samsung-galaxy-tab-a9-wifi-11-8-128gb-gris> | 199,00€ |
| Conexión a internet | Incluido en la compañía |

|  |  |
| --- | --- |
| PRESUPESTO 1 PARA CAMAREROS | |
| Producto | Precio |
| XIAOMI REDMI NOTE 14  C:\Users\Juan Pedro\AppData\Local\Packages\MicrosoftWindows.Client.Core_cw5n1h2txyewy\TempState\ScreenClip\{300D6DF0-7F22-4A8F-83E4-FB3D8FC334AA}.png  <https://www.pccomponentes.com/movil-xiaomi-redmi-note-14-4g-8-256gb-negro-medianoche-libre> | 189,98€ |
| Conexión a internet | Incluido en la compañía |

## 2.3 DIAGRAMA DE GANTT

Con el diagrama de Gantt planificaré y gestionaré mi proyecto. Esto es muy útil para saber el tiempo estimado que debería de tardar en realizar las diferentes fases de mí proyecto.



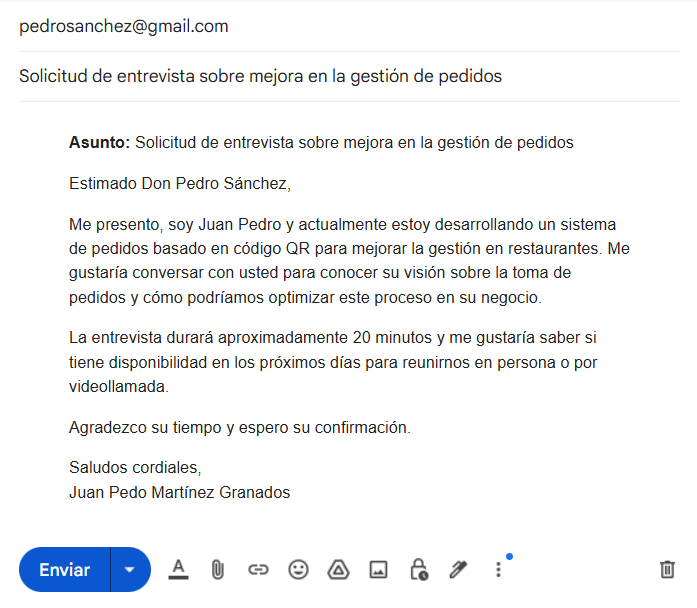
# 3. ANÁLISIS

# 3.1 ENTREVISTA AL CLIENTE

Las entrevistas que voy a realizar a continuación se la haré a Don Pedro Sánchez, que es el dueño de una empresa conocida mundialmente en el mundo de la hostelería. Este señor me pidió una entrevista porque necesita que su servicio vaya lo más rápido y eficiente posible.

A continuación muestro las entrevistas que he tenido con mi cliente en el transcurso y la realización de la aplicación.

MENSAJE DE SOLICITUD QUE MANDAMOS A NUESTRO CLIENTE

––

## 3.1 .1 PRIMERA ENTREVISTA: TOMA DE CONTACTO CON DON PEDRO SANCHEZ

Juan Pedro: Buenos días, don Pedro. Muchas gracias por su tiempo. Estoy desarrollando un sistema de pedidos basado en código QR para mejorar la eficiencia en mi restaurante, y su experiencia nos ayudará a realizar mejor la aplicación.

Pedro Sánchez: Buenos días, el placer es mío. Le doy las gracias por querer participar en este ambicioso proyecto.

Juan Pedro: Empecemos. Actualmente, ¿Cómo gestionáis los pedidos en su restaurante?

Pedro Sánchez: Los camareros de mi empresa toman la comandas a los clientes manualmente y llevan la bebida a la barra y la comida a la cocina. En horas donde la carga de trabajo es mayor suelen aparecer retrasos y errores.

Juan Pedro: ¿Cuáles son los principales problemas que enfrenta realizando las comandas de forma manual?

Pedro Sánchez: Los tiempos de espera de los clientes son altos, a veces hay confusiones en la toma de comandas y los camareros no dan abasto.

Juan Pedro: ¿Piensas que una aplicación basada en códigos QR que se pongan en las mesas podría ser una buena solución?

Pedro Sánchez: Si los clientes pudieran hacer su pedido desde la mesa, los camareros tendrían más tiempo para enfocarse en la atención y la comida y la bebida podrían llegar a la barra y a la cocina sin ningún tipo de error.

Juan Pedro: ¿Qué funciones considera usted imprescindibles para la realización de la aplicación?

Pedro Sánchez: Lo más importante en nuestra empresa es que los clientes puedan ver el menú actualizado, hacer los pedidos desde su móvil o tablet y que estos lleguen directamente a cocina y barra.

Juan Pedro: ¿Le gustaría recibir reportes sobre ventas consumo?

Pedro Sánchez: Sí, me ayudaría a tomar mejores decisiones sobre el menú y la gestión del stock.

Juan Pedro: ¿Usted cree que la aplicación debería de incluir notificaciones en tiempo real para avisar a la cocina y a la barra sobre los pedidos que han hecho los clientes?

Pedro Sánchez: Absolutamente, eso evitaría que nuestros camareros tengan que estar preguntando constantemente a la cocina o a la barra.

Juan Pedro: De acuerdo, ahora le preguntaré acerca del impacto que dará esta aplicación a su personal de trabajo y sus clientes. ¿Cómo cree que reaccionará su equipo ante este cambio?

Pedro Sánchez: Creo que al principio mis trabajadores se liarán pero si el sistema es fácil de usar yo pienso que se adaptarán rápido al cambio.

Juan Pedro: ¿Cómo cree que recibirán los clientes este cambio en su negocio?

Pedro Sánchez: Dependerá de si la aplicación sea intuitiva y rápida. A la mayoría de la gente le gusta la tecnología pero hay otra gente que le gusta más la atención de toda la vida.

Juan Pedro: ¿Le gustaría que el sistema permita pagos en línea o prefiere mantener el pago con los camareros?

Pedro Sánchez: Por ahora prefiero que los clientes paguen directamente al camarero, pero en un futuro podríamos considerar los pagos digitales a través de la aplicación.

Juan Pedro: Solamente me faltan dos preguntas por hacerle y no le quito más tiempo don Pedro. ¿Le preocupa la seguridad de los datos de los pedidos y del negocio?

Pedro Sánchez: Por supuesto, no quiero que personas terceras puedan acceder a la información de las ventas o de los pedidos.

Juan Pedro: Por último, ¿Quién debería tener acceso para modificar precios y menús?

Pedro Sánchez: Solo el administrador y quizás el encargado del restaurante.

Juan Pedro: Muchas gracias don Pedro. En resumen busca una aplicación que agilice los pedidos, reduzca errores y libere tiempo a los camareros. Además, valora los reportes de ventas y la seguridad de los datos. Con esta información, podemos diseñar una solución eficiente para su restaurante.

Pedro Sánchez: Exactamente. Me gustaría ver una prueba cuando este lista.

Juan Pedro: Por supuesto. Le mantendré informado sobre los avances de la aplicación.

Gracias por su tiempo.

## 3.1 .2 SEGUNDA ENTREVISTA. FUNCIONALIDAD DE OLVIDO DE CONTRASEÑA Y QUE LOS CLIENTES PUEDAN REGISTRARSE

Juan Pedro: Buenos días don Pedro, es un placer poder reunirme con usted nuevamente. En nuestra última conversación hablamos de mejorar la gestión de pedidos con código QR. Hoy quiero hablar sobre la posibilidad de que los clientes puedan poner observaciones si por ejemplo un usuario pide un tipo de carne y la carne la quiere más o menos hecha.

Pedro Sánchez: Buenos días. Sí, yo creo que sería buena idea que los clientes puedan poner sus observaciones de cómo quieren la carne o como quieren el pescado.

Juan Pedro: ¿Le gustaría que el cliente tuviera la opción de pagar a través de la página o directamente pagan a través de la barra?

Pedro Sánchez: Me gustaría que se pudiera de las dos formas para dar facilidades de pago al cliente.

Juan Pedro: Genial, eso es todo por hoy. Con esta información seguiré trabajando en la mejor forma de implementarlo. Le avisaré cuando tengamos avances en la aplicación, muchas gracias por su tiempo.

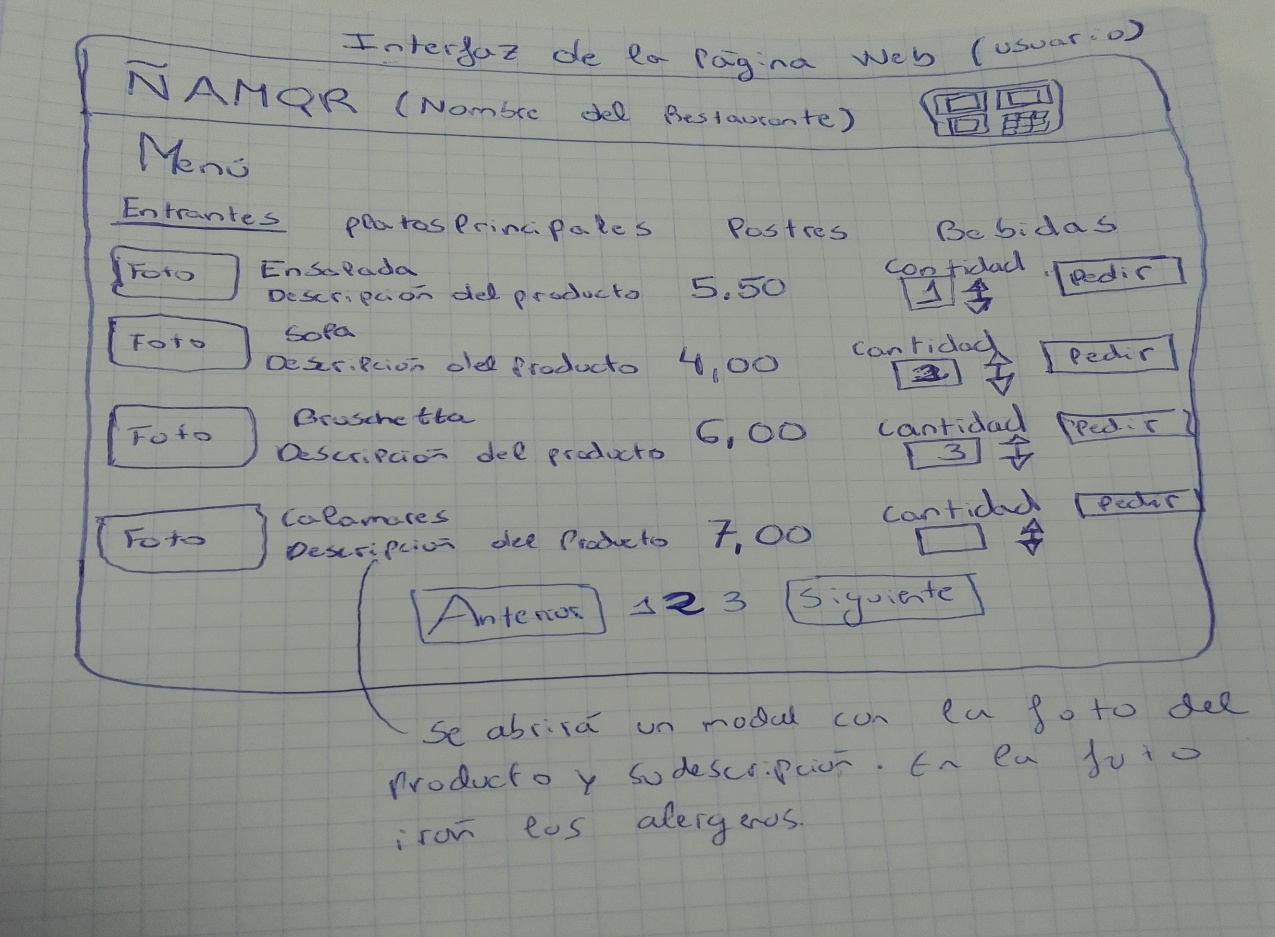
Pedro Sánchez: De acuerdo, gracias a ti.

## 3.1 .3 TERCERA ENTREVISTA (PRESENTACIÓN DE PROTOTIPOS)

Juan Pedro: Buenas tardes don Pedro, estoy encantado de que se haya podido reunir conmigo nuevamente. En esta reunión lo que pretendo es enseñarle cómo se vería la aplicación que estoy diseñando, tanto desde la perspectiva del cliente como la del personal del restaurante (cocina, barra, camarero y administración). Me interesa mucho conocer su opinión para seguir ajustando el desarrollo a sus necesidades.

Pedro Sánchez: Buenos días Juan Pedro, muchas gracias. Me gustaría ver el prototipo de la interfaz que verá el usuario para ver si se asemeja a lo que yo quiero que vea el cliente.

Juan Pedro: Se lo enseño ahora mismo para ver lo que opina.



Juan Pedro: Cuando el cliente escanea el código QR desde su mesa accederá a la interfaz, que será de este estilo. El cliente podrá ver un menú con las diferentes categorías. En cualquiera de las opciones del menú aparecerá una foto del producto, el nombre del producto y una descripción cuando se pinche en descripción del producto, que será un modal donde aparecerá una foto más grande con la descripción del producto y la información sobre alérgenos. De cada producto nos aparecerá el precio y el número de unidades que vaya a pedir el cliente. El cliente solo tiene que seleccionar que producto quiere, las unidades que desea y pinchar en el botón pedir, que estará al lado de la cantidad que quiere el cliente.

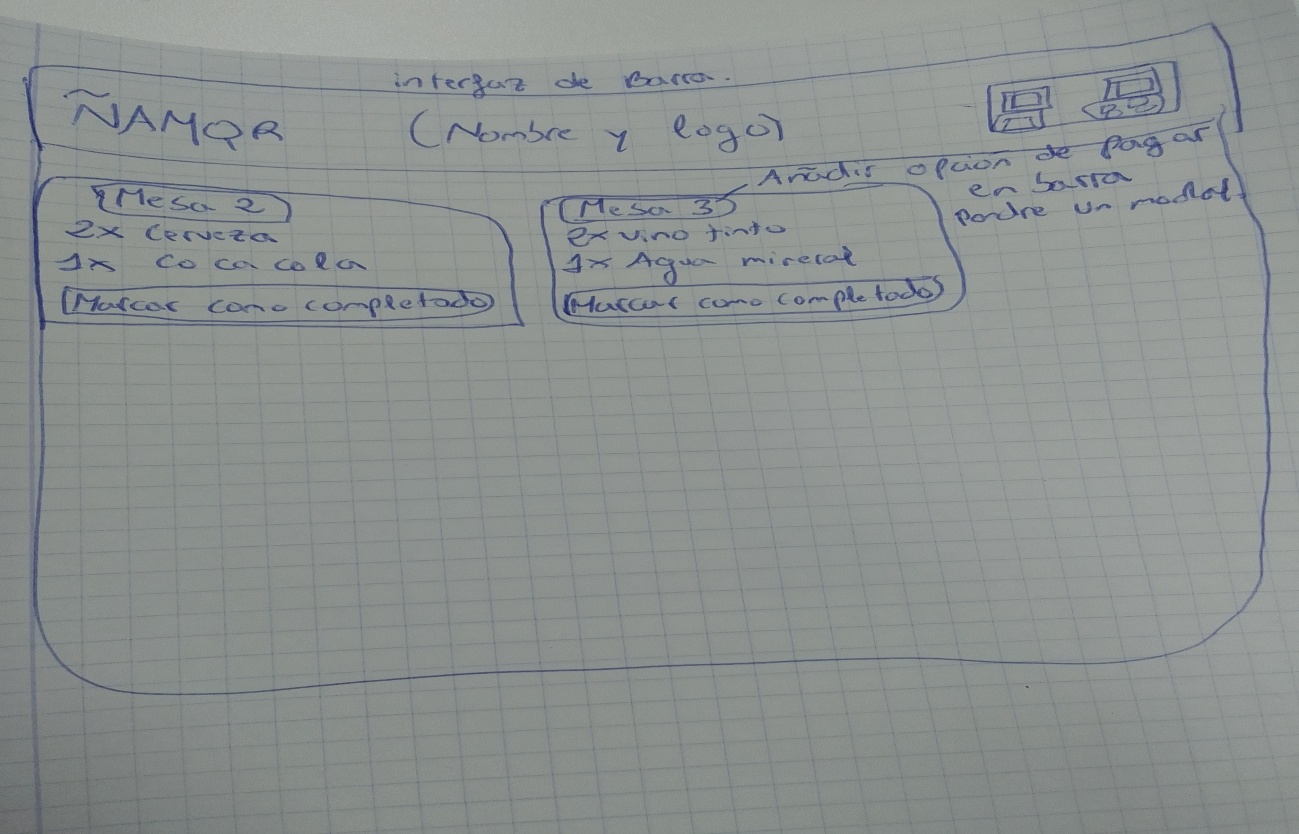
¿Le resulta clara y sencilla esta pantalla para sus clientes?

Pedro Sánchez: Me resulta muy sencilla y clara aunque también se podría mejorar añadiendo una opción de pago para que el cliente pueda pagar a través de la aplicación web.

Juan Pedro: De acuerdo don Pedro, le añadiremos esa opción para que los clientes tengan comodidad a la hora de pagar. Podríamos poner como método de pago por ejemplo pagar por Bizum, por tarjeta y en efectivo.

Pedro Sánchez: Me parece una genial idea, eso le dará comodidad al cliente a la hora de pagar.

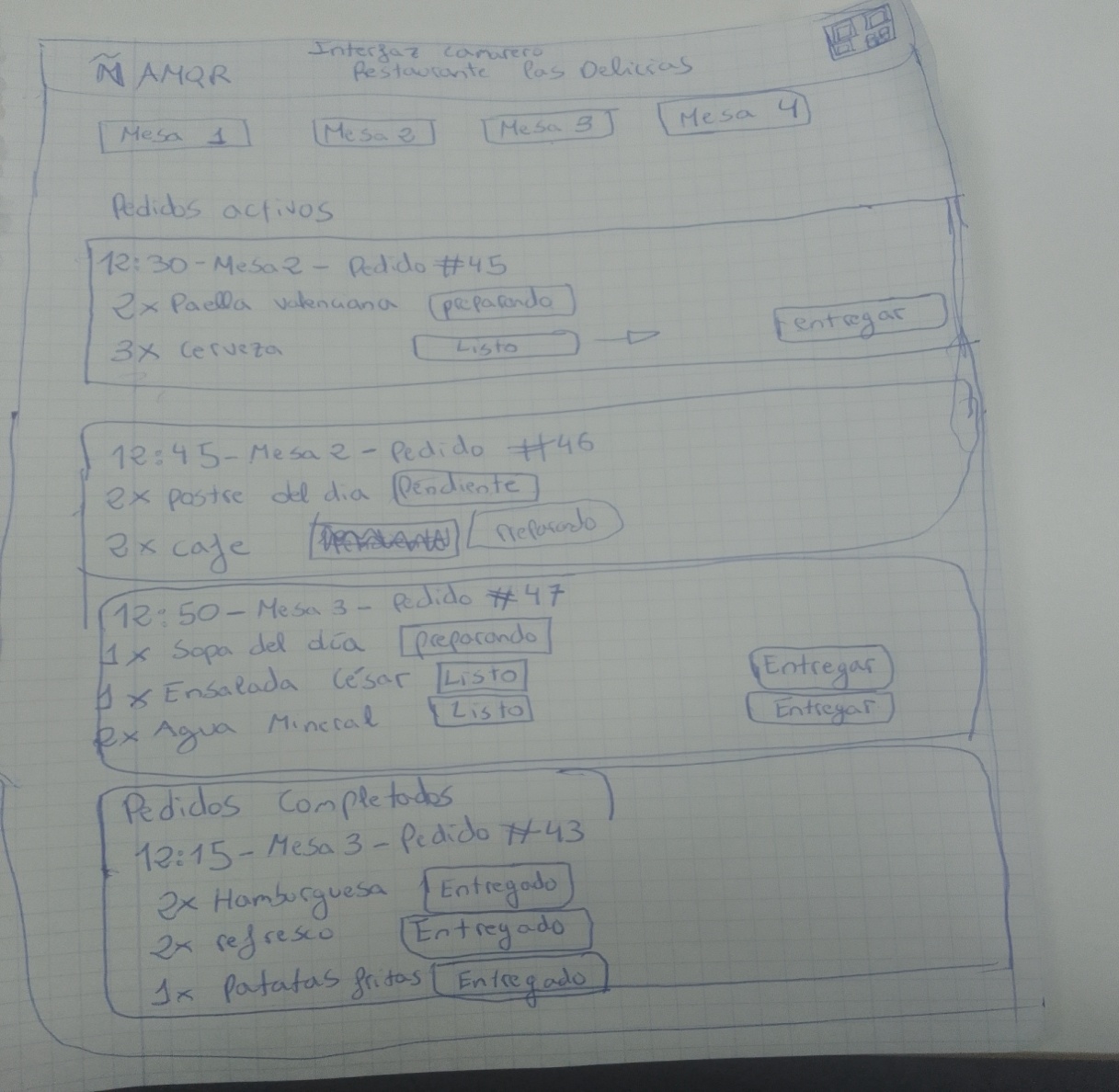
Juan Pedro: De acuerdo, ahora pasaremos al prototipo de la barra.



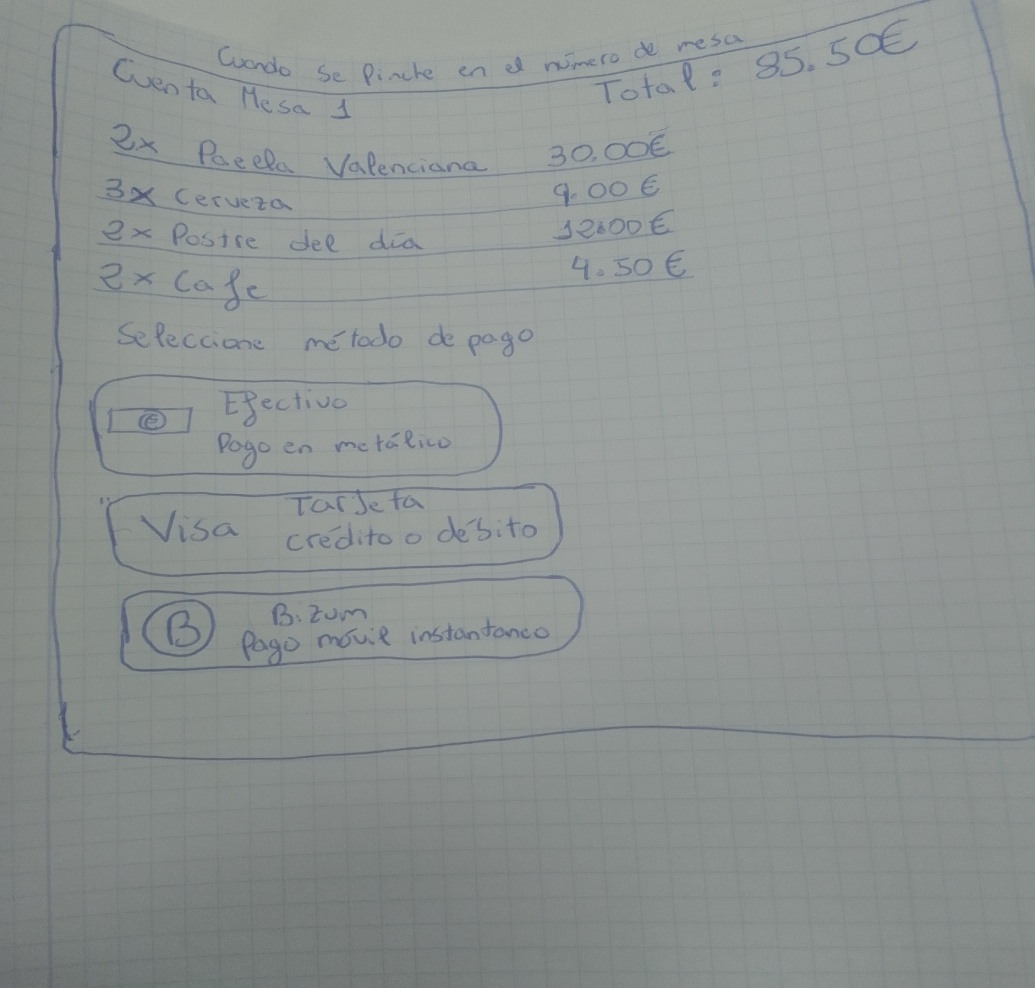
Este es el prototipo que tiene la interfaz del que se encarga de la barra. En la interfaz le aparecerán el número de la mesa y la bebida que falta por salir. Me gustaría saber si quiere que cada vez que se pinche en el número de mesa nos muestre un modal con la cuenta de esa mesa para que el cliente pueda pagar en barra si así lo desea.

Pedro Sánchez: El prototipo me parece bien y la opción de que el cliente pueda pagar en barra también ya que el trabajador de la barra solo se encargará de preparar la bebida a las mesas y atender en barra por lo que me parece una idea genial.

Juan Pedro: De acuerdo pues pasemos al prototipo del camarero.



La interfaz del camarero tendrá este esqueleto. Estará formado por los números de mesa que hay en su negocio y abajo tendrá los pedidos activos y los pedidos que han sido completados. Cuando un pedido esté listo los camareros sabrán que tienen que llevarlo a la mesa y una vez que lo lleven a la mesa pincharán en la opción entregar y el pedido se quedará entregado. Una vez que todos los pedidos de la mesa estén entregados el pedido pasará a los pedidos completados. También le podemos poner la opción de que cuando el pedido este completado borrarlo completamente y quitar la opción de pedidos completados. Cuando el cliente quiera pagar el camarero solo pinchará en la mesa que ha pedido la cuenta y le aparecerá un modal con la cuenta total de la mesa. Lo pondré como un modal y daré facilidades para que el cliente pueda pagar.

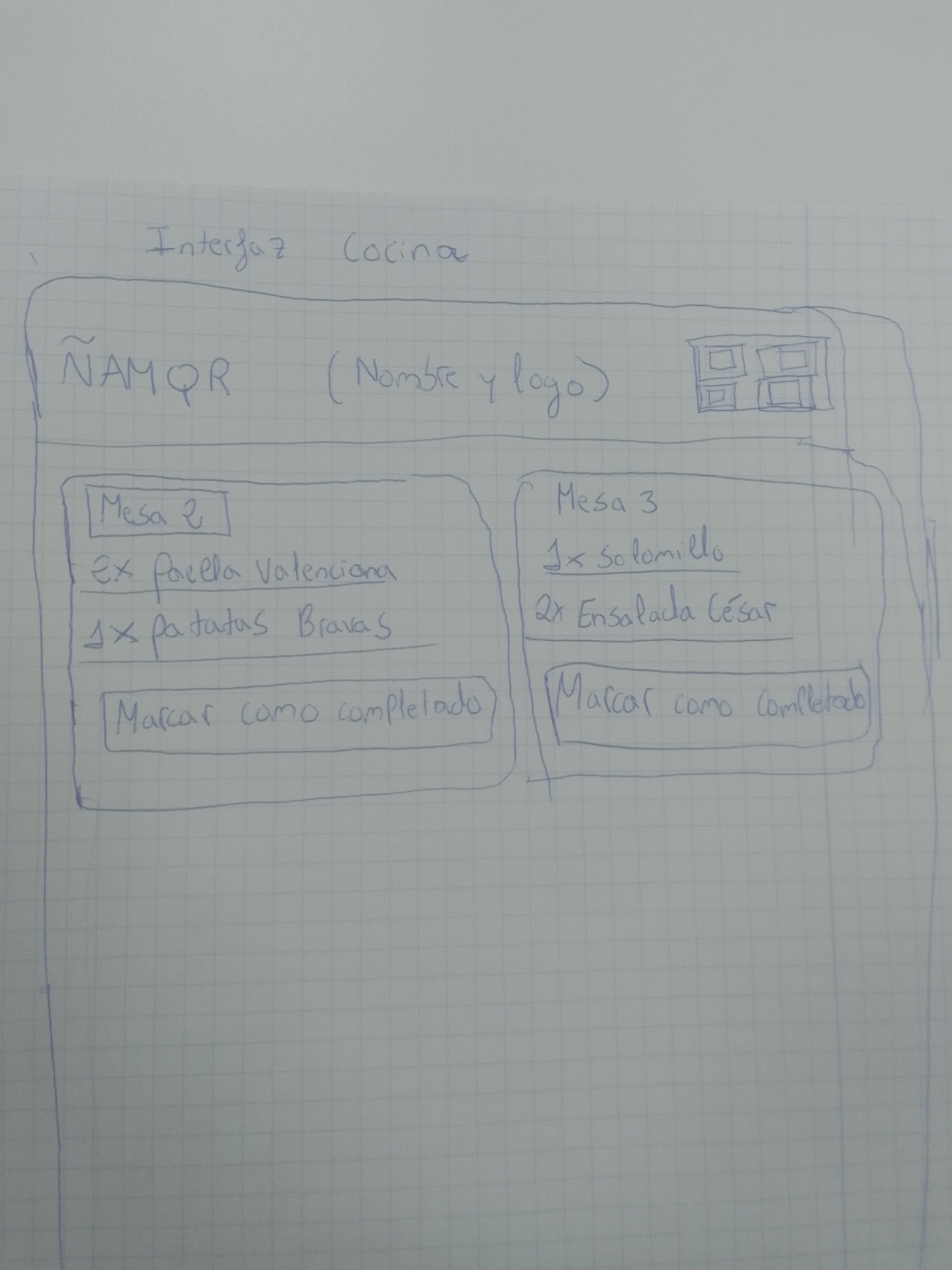


Pedro Sánchez: La idea del prototipo me parece correcta pero me gustaría que los pedidos que se han completado quitarlos de la interfaz ya que ahorrará espacio en la aplicación. Lo que si me gustaría es que el total de todas las cuentas se me guardará para yo poder hacer mis cuentas. Ni que decir que cada vez que una mesa pague esa mesa volverá a estar libre cuando se vayan, por lo que los pedidos que estén pagados también se borrarán del modal.

Juan Pedro: De acuerdo don Pedro, haré las mejoras que me propone.

El siguiente prototipo que le enseñaré es el de cocina para ver qué le parece o que le podríamos mejorar.

Pedro Sánchez: De acuerdo, puede enseñármelo.

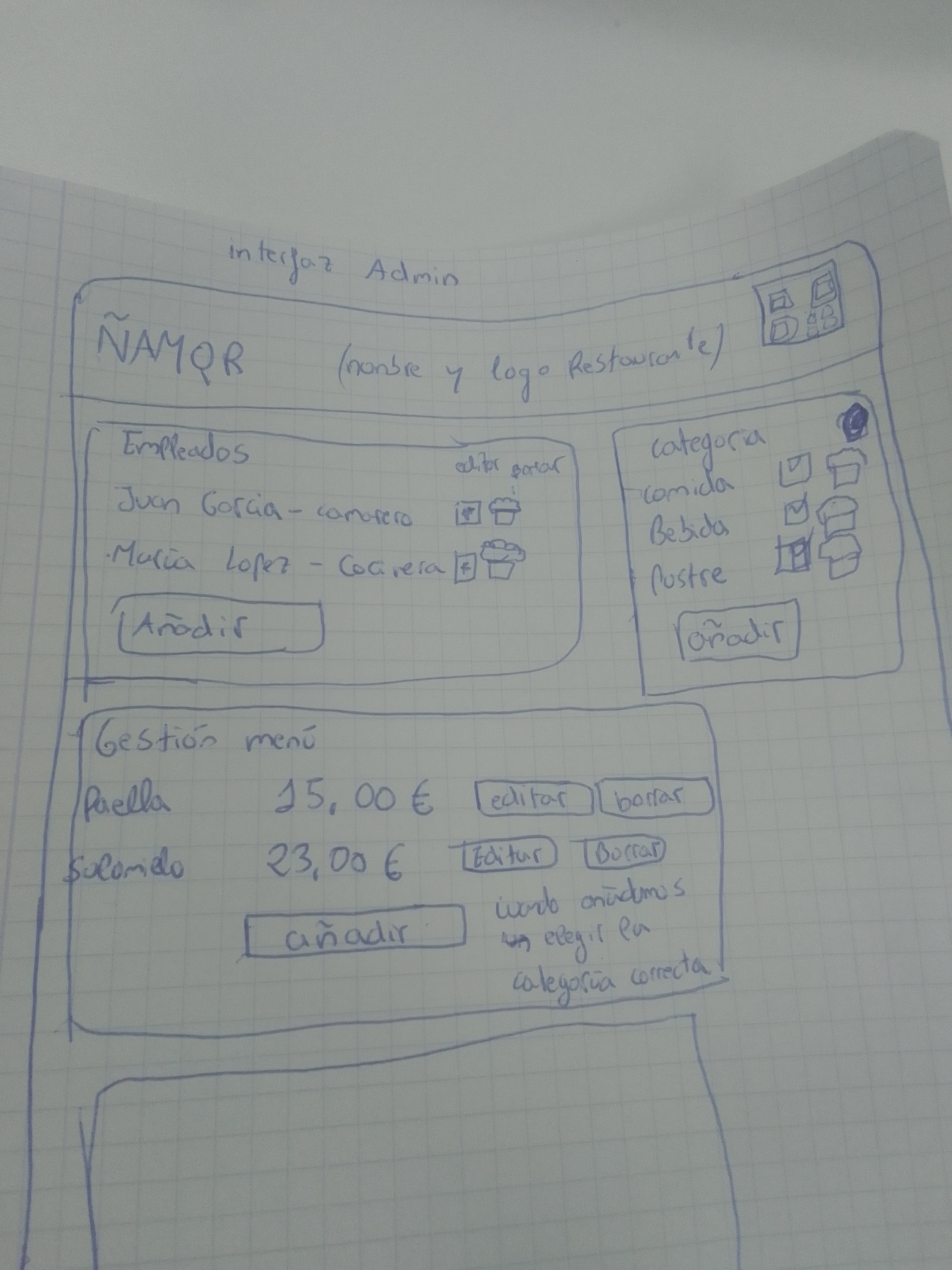


Juan Pedro: En este prototipo veremos el número de la mesa y lo que falta por salir de comida. Una vez que el cocinero ha preparado un plato pinchará en completado para que al camarero le llegué la notificación para poder recogerlo.

Pedro Sánchez: En vez de poner el botón en todo el pedido podrías poner un botón en cada plato para saber la comida que ha salido, así se verá más claro. Todo lo demás lo veo bien.

Juan Pedro: De acuerdo don Pedro haremos esa mejora.

Le voy a enseñar el último prototipo que es el del administrador.



Juan Pedro: Este es el prototipo del menú de administración. En este prototipo nos dará la opción de añadir, editar y borrar algún trabajador, categoría o producto. Más adelante en el manual de administrador se explicará más detalladamente.



Pedro Sánchez: De acuerdo, veremos cómo se queda el panel de administración y ya te comentaré si me gusta o no.

Juan Pedro: Perfecto, Por mi parte eso es todo don Pedro, le tendré informado de cualquier duda o sugerencia que tenga con respecto a la aplicación.

Que pase un buen día.

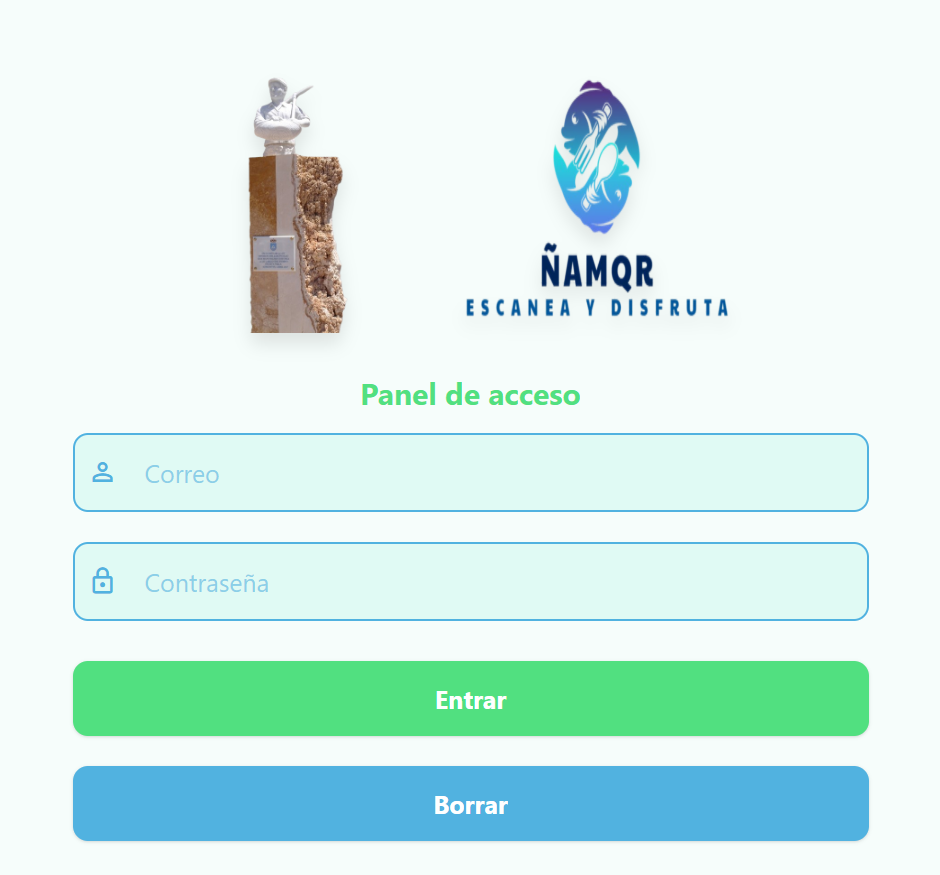
Pedro Sánchez: Perfecto, estoy emocionadísimo de este proyecto. Nos vemos.

## 3.1 .4 CUARTA ENTREVISTA (DISEÑO Y FUNCIONALIDADES)

Juan Pedro: Buenos días don Pedro estoy encantado de volver a verte. Le enseñare como llevamos la aplicación hasta ahora, yo creo que le va a gustar mucho.

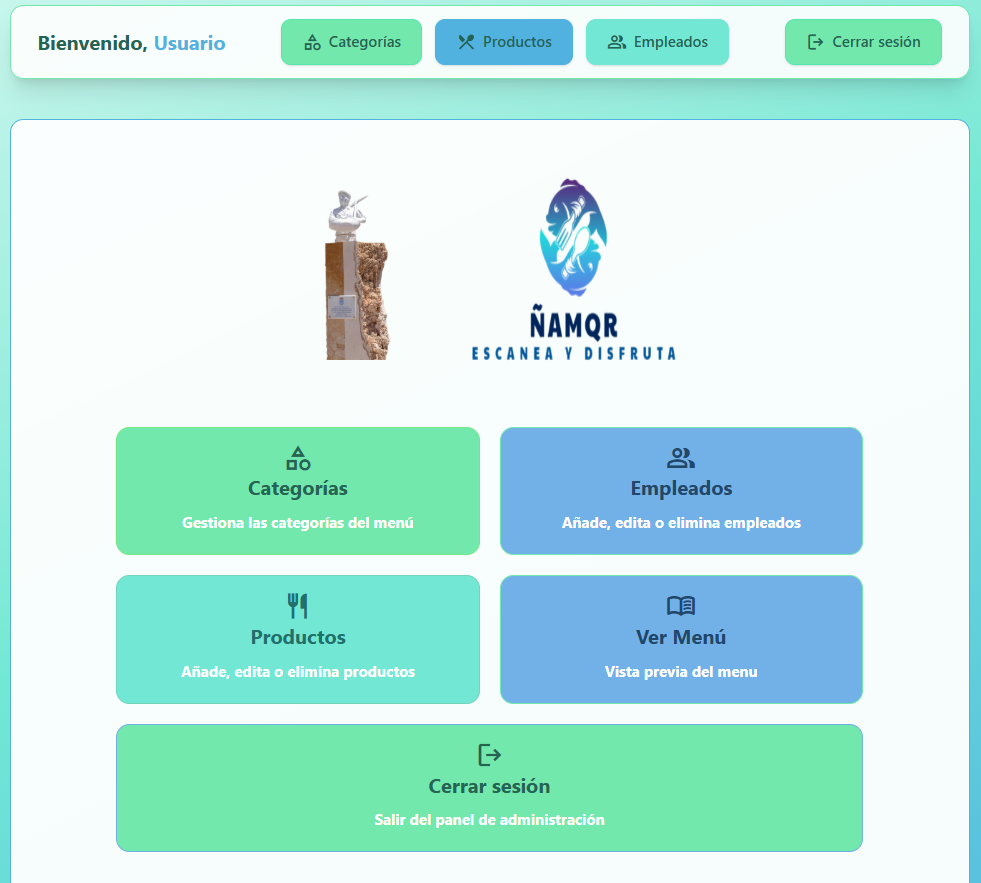
Pedro Sánchez: Buenos días Juan Pedro, podemos comenzar cuando usted quiera.

Juan Pedro: Primero empezaremos por la pantalla de login. Le he hecho un diseño moderno y vistoso.



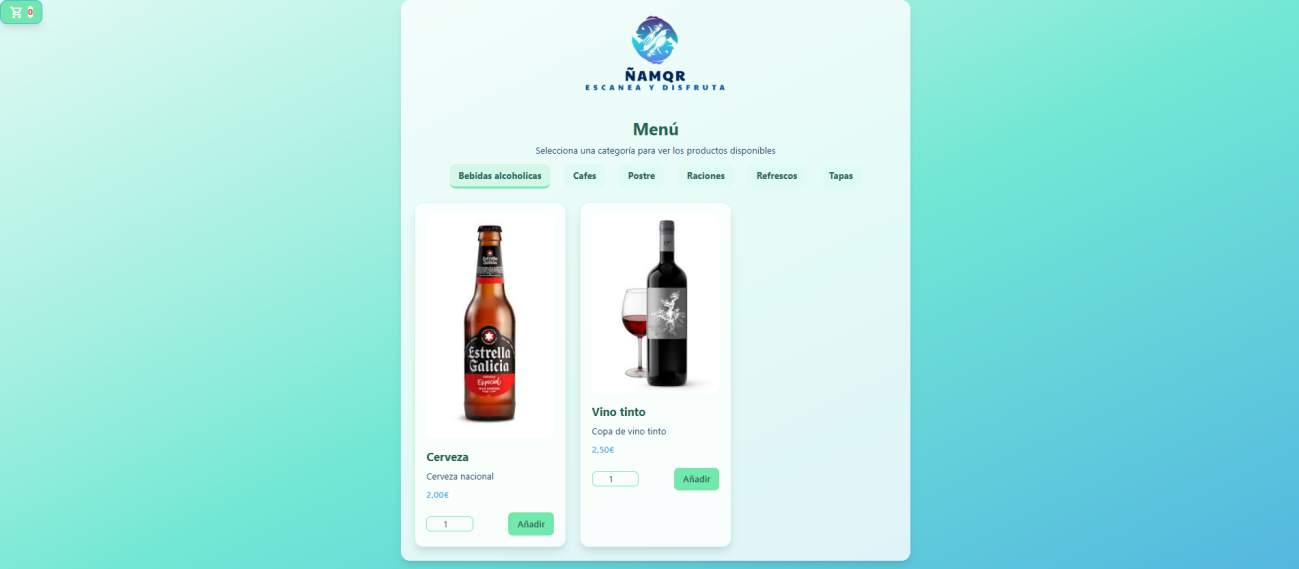
Pedro Sánchez: Me gusta mucho el estilo que le has dado a la página del login, se puede quedar así.

Juan Pedro: Perfecto pues ahora te enseñaré el panel de administración.



Pedro Sánchez: Me gusta muchísimo y los colores que has implementado pegan mucho con lo que yo quería. Por ponerte un pero en la cabecera me gustaría que haya un botón para poder volver al menú.

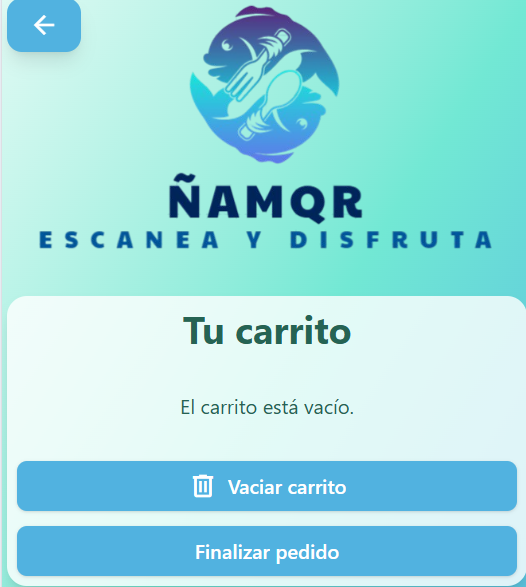
Juan Pedro: vale perfecto, le añadiremos el botón. Ahora te enseñaré como se verá el menú que verán los clientes para hacer su pedido.



Pedro Sánchez: Me parece muy bonito pero me gustaría que en esta página existiera un botón para que el cliente pueda llamar al camarero sin necesidad de levantar la mano o levantar la voz para que le oiga. También quiero una opción para que el cliente pueda pagar su cuenta a través de la aplicación.

Juan Pedro: Sin problema, en los próximos días lo implementaremos.

También me gustaría enseñarle como quedará la página cuando el cliente pinche en el carrito para procesar su pedido.



Pedro Sánchez: Me gusta mucho el diseño y la forma.

Juan Pedro: Me alegro mucho de que te guste. Por mi eso es todo por hoy, ¿tiene alguna sugerencia?

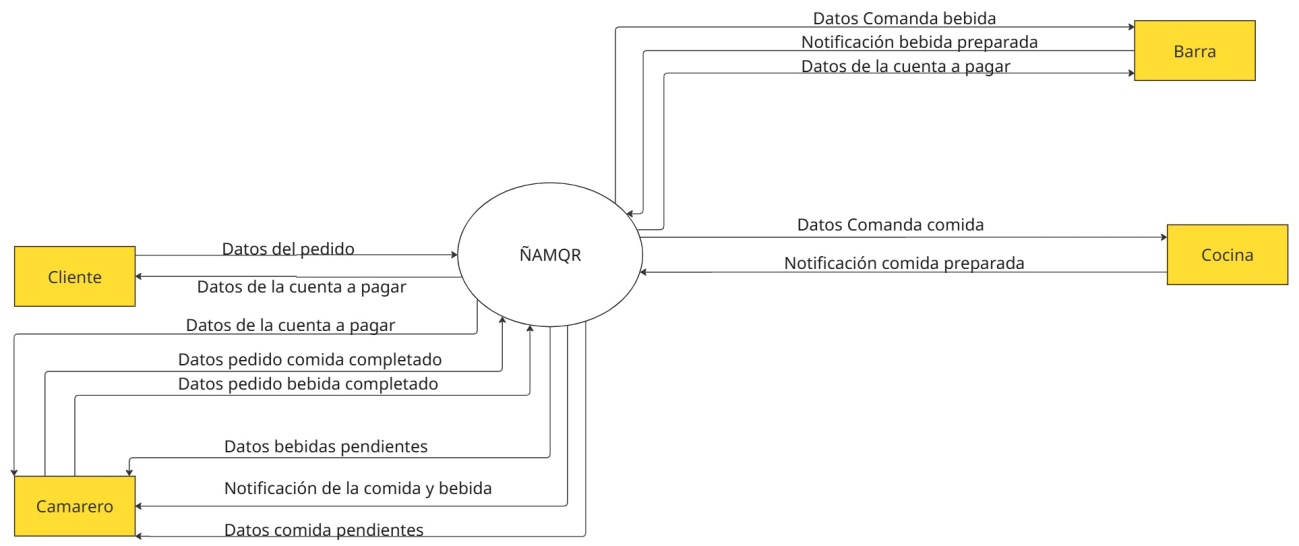
Pedro Sánchez: No, está todo bien. En la próxima reunión vemos las funciones que faltan y damos por terminada la aplicación.

Juan Pedro: Perfecto don Pedro, ha sido un placer como siempre.

Pedro Sánchez: Hasta la próxima.

# 3.2 DIAGRAMA DE FLUJO DE DATOS

El diagrama de flujo de datos de la aplicación es el siguiente:



# 4. DISEÑO

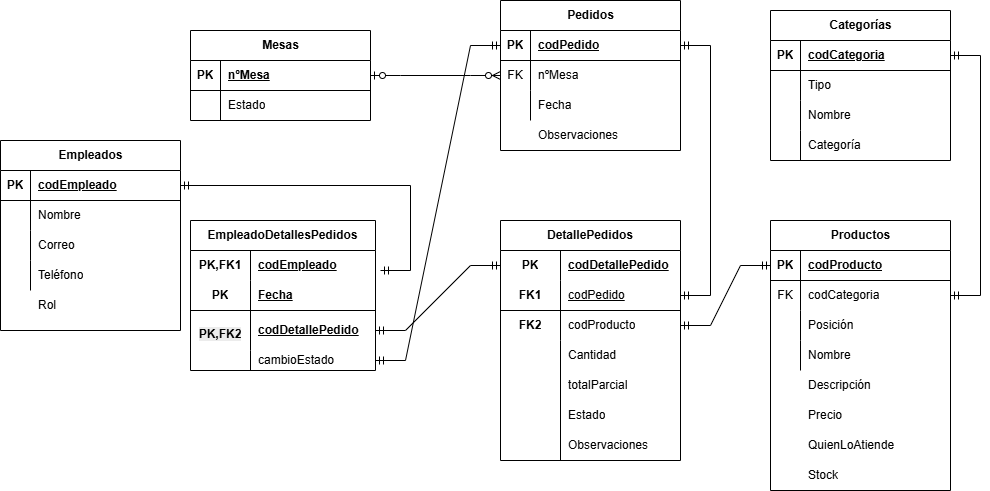
# 4.1 DIAGRAMA ENTIDAD RELACIÓN

A continuación mostraré el diagrama entidad relación de mi proyecto, que es el que usaré para poder realizar toda la aplicación.



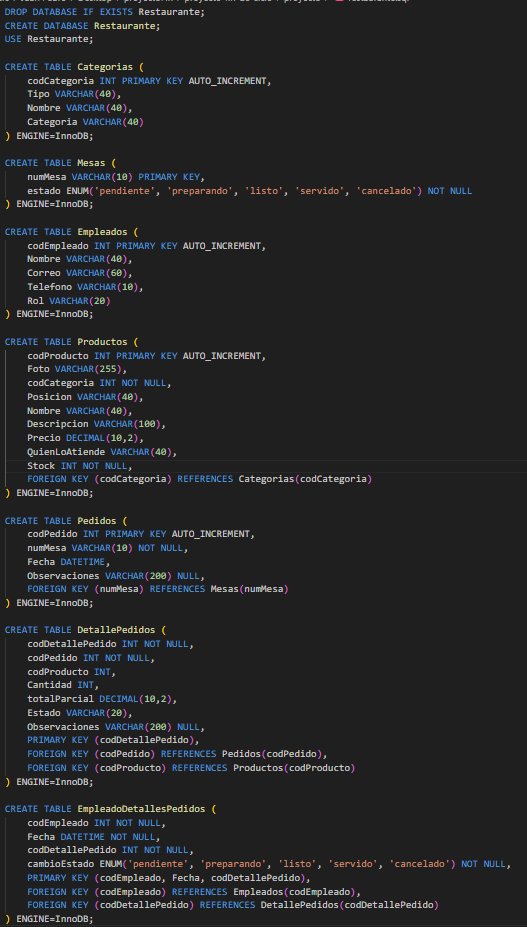
# 4.2 DISEÑO RELACIONAL

Una vez que hemos diseñado el modelo E/R procederemos a realizar el diseño relacional. He utilizado draw.io, que es una página web en línea que nos da la facilidad de poder hacer diagramas de todo tipo.



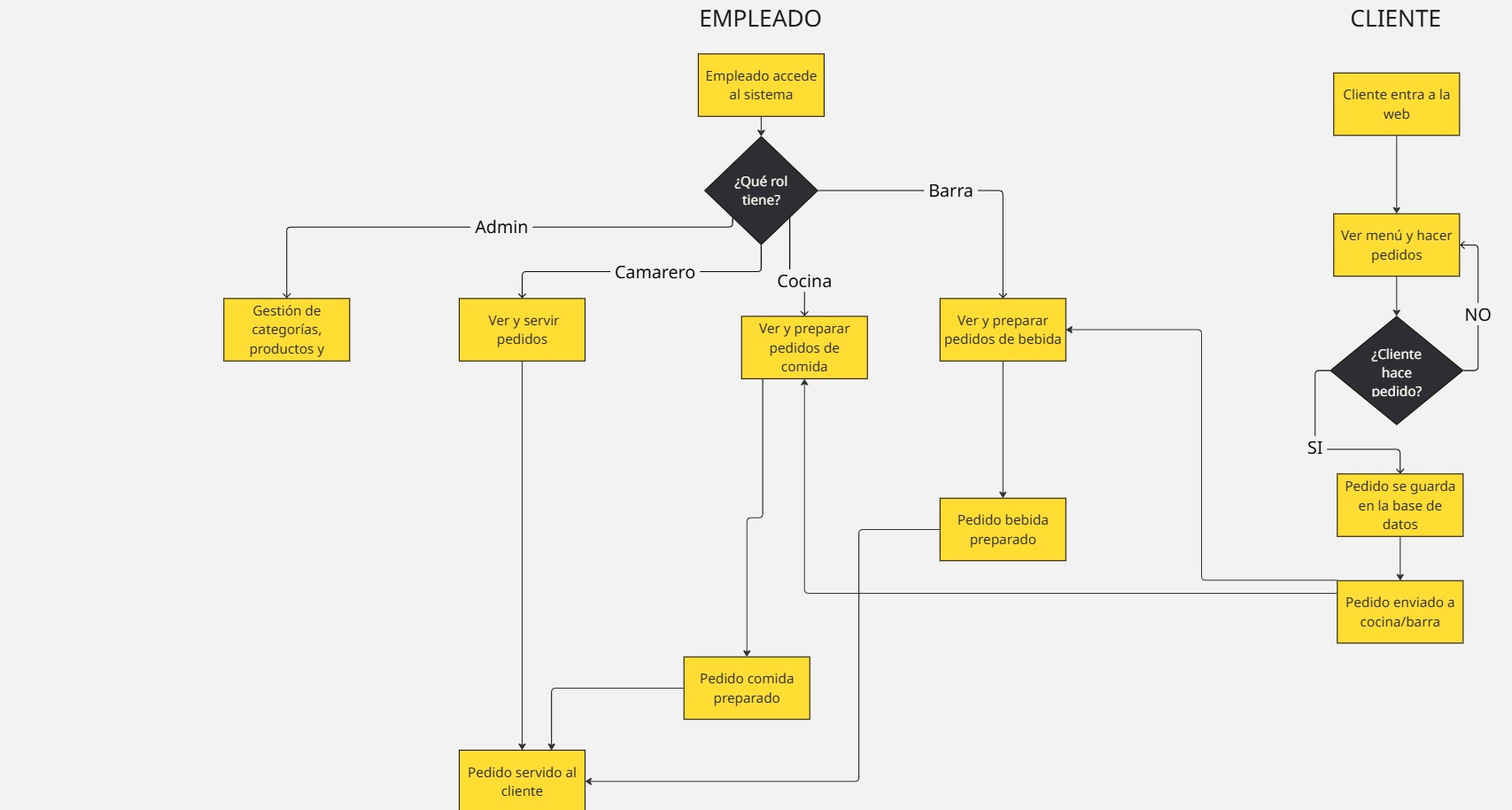
# 4.3 PASO A TABLAS

Este será el script que usaré para realizar todas las consultas en mi base de datos.



# 4.4 PROCESOS

Este será mi diagrama de procesos. En el veremos todas las acciones y entradas y salidas que hará mi programa.



A continuación detallaré lo que hace cada pantalla de mi aplicación.

* **menu** - >
* Una vez que el cliente haya escaneado el código QR le llevará a esta pantalla.
* El cliente podrá seleccionar los productos que quiera pedir. Cuando el cliente seleccione una cantidad de un producto y pinche en pedir, el producto y la cantidad irá directamente al carrito. Dentro del carrito tendremos los productos que ha seleccionado el cliente junto con las unidades de cada producto. Para hacer el pedido pinchamos en hacer pedido y directamente le llegará la comida a la cocina y la bebida a barra. En este menú tendremos la opción de poder pagar online mediante un botón y la posibilidad de llamar al camarero.
* Tendremos el carrito en el lado derecho para que el usuario cuando quiera hacer el pedido que pinche y que le lleve a la confirmación de su pedido.
* **login** - >
* Esta pantalla es la encargada de que los empleados puedan iniciar sesión poniendo su correo electrónico y su contraseña. Dependiendo del rol que desempeñe el empleado esta pantalla nos llevará a un lugar o a otro. Si las credenciales no son correctas nos mostrará el mensaje de error.
* **admin ->** 
  + Si el empleado pone las credenciales como administrador, aplicación le redirigirá a esta pantalla. Dentro de esta pantalla podremos hacer diferentes acciones:
* Añadir, modificar o eliminar categorías.
* Añadir, modificar o eliminar productos.
* Añadir, modificar o eliminar empleados.
* Añadir, modificar o eliminar mesas.
* Un botón que nos redirigirá a la pantalla de menú para ver si los productos se han creado correctamente.
* Los cambios se actualizarán automáticamente.
* **Carrito ->** 
  + Si el usuario pincha en el carrito entrará en esta interfaz. En esta interfaz podremos encontrar:
* Resumen del pedido con los productos que el cliente ha añadido al producto.
* Botones individuales para poder eliminar un producto del carrito.
* Un botón para vaciar el carrito y otro para confirmar el pedido.
* **cocina ->**
* Si un usuario pone las credenciales como cocinero, la aplicación le redirigirá a esta pantalla. En esta pantalla veremos las siguientes opciones:
* Ver los pedidos de la comida que han pedido los clientes. Estos pedidos irán agrupados por nombre y los pedidos que estén pendientes estarán en la punta arriba para que los cocineros puedan verlo rápido y no pierdan tiempo.
* Notificar al camarero que un pedido está listo para servir.
* **barra ->**
* Si un empleado pone sus credenciales como encargado de barra la aplicación le redirigirá directamente a esta pantalla. En esta pantalla encontraremos:
* Los pedidos de la bebida agrupados por nombre.
* Los pedidos que estén pendientes estarán arriba por eficiencia.
* Opción para añadir producto a una mesa por si un cliente pide en barra.
* Marcar producto como listo para servir.
* Podremos ver todos los pedidos agrupados por mesa.
* Opción para que el cliente pueda pagar en la barra.
* Todos los números de las mesas que están ocupadas. Al pinchar en una mesa nos saldrá todo lo que ha pedido la mesa junto con su total. Habrá un botón para imprimir la cuenta para que la vea el cliente.
* **Camarero ->**
* Está interfaz es la que verá el empleado que ponga las credenciales como camarero. En esta pantalla encontraremos:
* Los pedidos pendientes de servir agrupados por nombre.
* Marcar un pedido como completado una vez que haya sido servido al cliente.
* Todos los números de las mesas que están ocupadas. Al pinchar en una mesa nos saldrá todo lo que ha pedido la mesa junto con su total. Habrá un botón para imprimir la cuenta para que la vea el cliente.

# 5. CODIFICACIÓN

En esta fase explicaré:

* Para qué sirven los archivos más importantes de la aplicación.
* Explicaré en detalle websockets.
* Pondré las funciones más importantes.

## 5.1 ARCHIVOS MÁS IMPORTANTES

**funciones.php:** En este archivo se encuentran casi todas las consultas que hago en mi aplicación. La he utilizado para modularizar y no repetir código.

**camarero.php:** En este archivo podremos ver los pedidos que están listos para servir y marcarlos como servidos. Cuando un producto esta servido desaparece de la interfaz y si un pedido no tiene productos ya para servir desaparece también.

**cocina.php:** En este archivo veremos los pedidos que están en preparación o preparando. Cuando un producto lo marcamos como listo desaparece de la interfaz. Si todos los productos de un pedido están listo s desaparece el pedido entero de la interfaz.

**menu.php:** Esta es la interfaz que verá el cliente. En ella verá todos los productos agrupados por su categoría. El usuario podrá añadir la cantidad que quiere de cada producto.

**carrito.php:** En esta interfaz el usuario podrá ver lo que ha añadido al carrito en la página de menú.php. El cliente podrá poner observaciones de producto y observaciones generales del pedido. Si no quiere el pedido podrá eliminar los productos del carrito, ya sea de forma manual o eliminar el carrito entero.

**confirmarPedido.php:** En esta interfaz el cliente verá el resumen del pedido que ha ejecutado.

**mesa.php:** Este archivo lo que hace es asignarle la mesa al cliente y marcar la mesa como reservada cuando el cliente escanea el QR en la mesa.

**servidorWebsocket.php:** Es el archivo principal de los mensajes en tiempo real.

**websockets.js:** Será el encargado de transmitir los mensajes de los distintos clientes en nuestro programa.

**enviarNotificaciones.php:** Su función principal es permitir que otros archivos como cocina.php o confirmarPedido.php puedan enviar una notificación a mi servidor websocket.

**gestionPedidos.js:** Este archivo sirve para no repetir el mismo código en diferentes archivos y centraliza la lógica para mostrar, actualizar o eliminar pedidos de la interfaz.

**cocina.js:** Se encarga de cargar y mostrar los pedidos pendientes que deben prepararse, haciendo peticiones a cocina.php Cuando un cocinero marca un producto como "listo", cocina.js actualiza la interfaz y avisa al servidor, que a su vez notifica a los camareros en tiempo real.

**carrito.js:** Se encarga de darle funcionalidad al carrito para añadir productos, cambiar cantidades, eliminarlos y guardar el estado del carrito en localStorage. También calcula y muestra el total del pedido antes de que el cliente decida confirmar el pedido.

**camarero.js:** Este archivo es el encargado de cargar y mostrar los productos que hay listos para servir. Cuando un camarero marca el producto como servido actualiza la interfaz y avisa a camarero.php para que el cambio se registre correctamente en la base de datos.

## 5.2 WEBSOCKET

En nuestra aplicación, las pantallas de la cocina y los camareros se actualizan solas, sin necesidad de recargar la página. Esto se logra gracias a una tecnología llamada WebSockets.

A continuación explicaré el flujo que hace esta maravillosa herramienta:

Este es el corazón del sistema: **servidorWebsocket.php.**

Imagina que este es un operador de radio que siempre esta encendido y escuchando. Es un script PHP especial que se ejecuta constantemente en el servidor. Su único trabajo es mantener una lista de quién está "conectado en el sistema y pasarles los mensajes que llegan al servidor.

La Conexión de los Usuarios: **websocket.js.**

Cuando un cocinero o un camarero abre su interfaz, este archivo JavaScript es el encargado de establecer una conexión directa y permanente (como una llamada telefónica que no se cuelga) con nuestro servidorWebsocket.php

A partir de ese momento, el navegador se queda a la escucha de cualquier mensaje que el servidor le envíe.

3. El Evento: Alguien Hace Algo Importante ejemplo: un cliente confirma un nuevo pedido. Esta acción la procesa un archivo normal en confirmarPedido.php que guarda el pedido en la base de datos.

4. El Mensajero: **enviarNotificacion.php**

Una vez guardado el pedido, confirmarPedido.php necesita avisar a todos. Para ello, llama a una función del archivo enviarNotificacion.php. Este archivo actúa como un mensajero: se conecta rápidamente a nuestro operador de radio (servidorWebsocket.php) y le entrega el mensaje, por ejemplo: {"tipo": "NUEVO\_PEDIDO", "id": 123}. Luego, el mensajero se va.

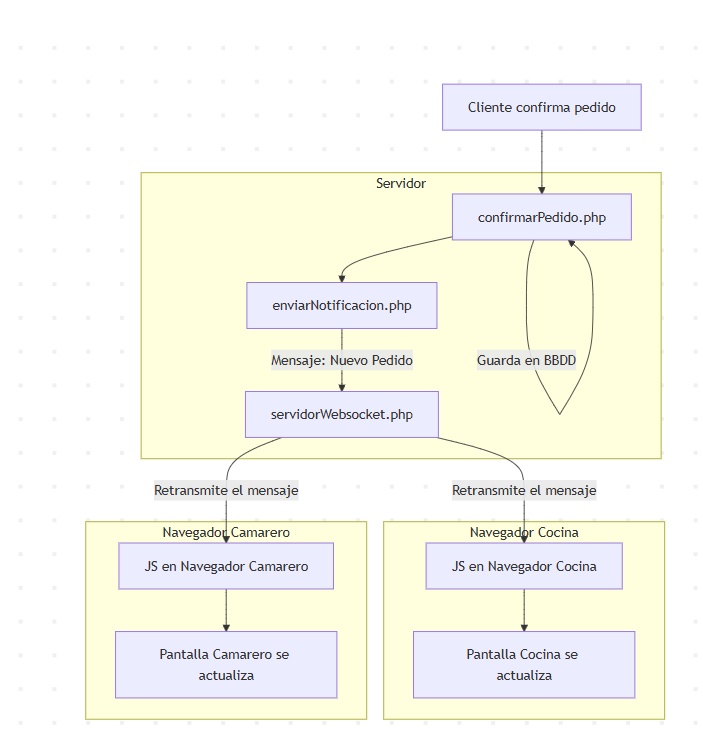
5. La Difusión: El Operador Entra en Acción

El servidorWebsocket.php recibe el mensaje del mensajero. Inmediatamente, busca en su lista de conexiones activas y retransmite el mensaje al cliente necesario.

En los navegadores de la cocina, el archivo websocket.js recibe el mensaje.

Este se lo pasa a cocina.js, que sabe qué hacer con él: lee la información del nuevo pedido y actualiza la pantalla dinámicamente para mostrarlo, sin que el cocinero tenga que hacer nada.

Este es el diagrama para entender mejor como funciona websocket.



# 6. PRUEBAS

Esta fase la he acompañado con la fase de codificación.

Probé que la función comprobar admin() funcionará en todos los archivos. Si un cliente no ha iniciado sesión se le redirija automáticamente al login ya que la variable de sesión esta vacía. También he probado que los de cocina por ejemplo no puedan entrar a ningún archivo relacionado con la administración.

He probado que tanto las categorías, empleados y productos se puedan añadir, borrar o editar correctamente y que las consultas se guarden bien en la base de datos.

También he probado que los productos y las categorías salgan bien en el menú cuando añado, elimino o edito una categoría o un producto.

También he hecho pruebas relacionadas con el carrito para asegurarme que cuando un cliente añade productos que los productos salgan bien en el carrito. También he hecho pruebas para ver si el contador del carrito se actualizaba correctamente cuando el cliente añada productos al carrito y cuando el cliente borraba un producto del carrito o todo el carrito.

En informacionPedido he mirado que el pedido se inserte correctamente en la base de datos y que al usuario le salgan todos los datos de su pedido.

Relacionado con websocket he probado que todos los cambios de estado se actualicen correctamente en tiempo real.

En la interfaz tanto del camarero como la de cocina he probado que cuando cambia de estado un producto en la interfaz se vea bien y cuando algún pedido cambia de estado o un producto que se elimine directamente de la interfaz dependiendo del estado final que se tiene que ver en cada interfaz.

También he hecho pruebas para ver qué pasa si websocket no funciona. El único problema que habría es que tanto el camarero como el cocinero tendrían que estar refrescando constantemente la interfaz para ver el cambio de estado de los pedidos. Igualmente el sistema se intentará reconectar cada 5 segundos para ver si ha sido una pérdida de conexión breve.

# 7. MANUALES

## 7.1 MANUAL DEL ADMINISTRADOR

Iremos al login y nos loguearemos como administrador:



Si todo ha salido bien y no nos hemos equivocado al poner el correo o la contraseña entraremos al panel del administrador.

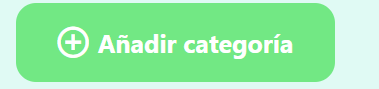


Dentro del panel de administración tendremos varias opciones:

**Categorías:** En esta opción podremos añadir, eliminar o editar una categoría existente:



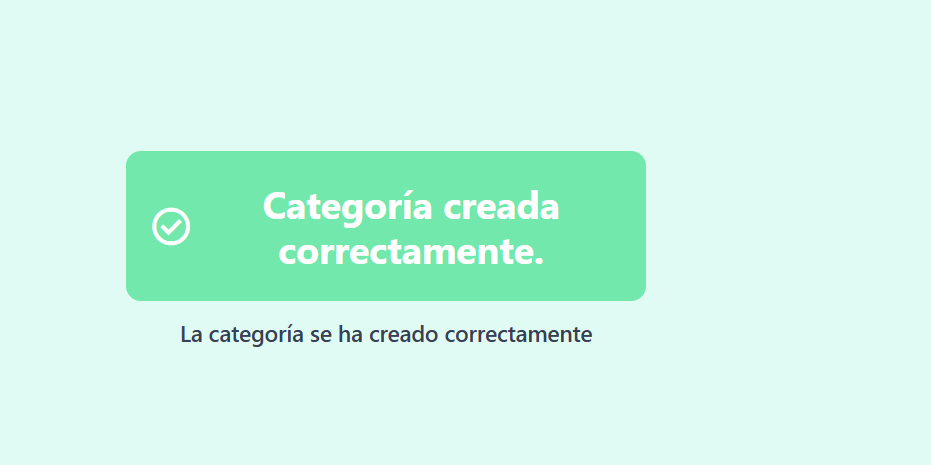
Para añadir una categoría pincharemos en añadir categoría.



Cuando pinchemos nos saldrá un formulario para poner el nombre de la categoría y su descripción. Para crearla pinchamos en añadir categoría.

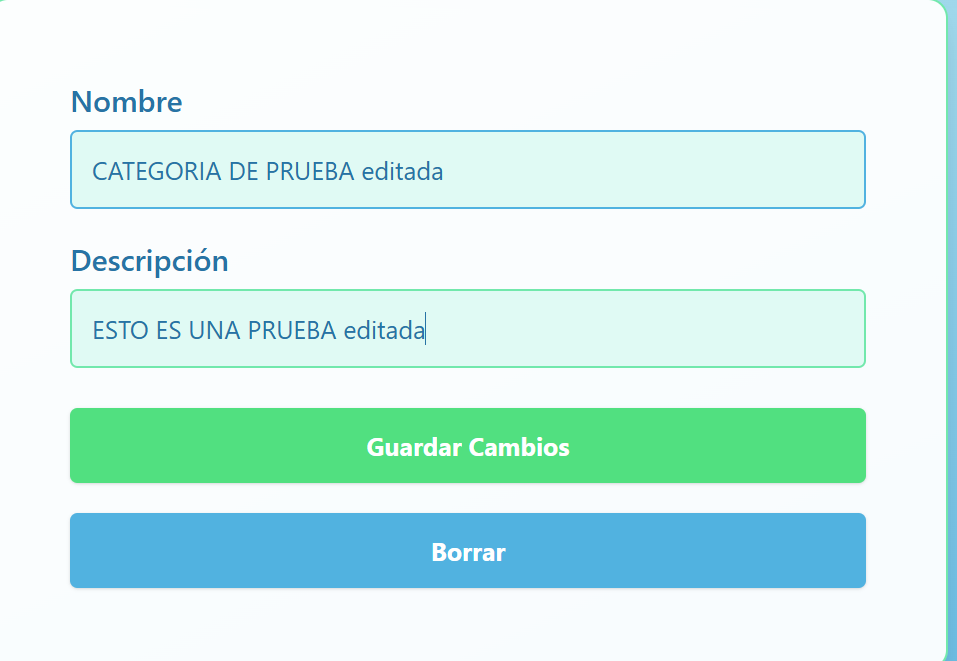


Una vez que pinchemos en añadir categoría nos saldrá un mensaje de éxito. Si un caso fallará la creación de la categoría el programa tambien nos avisaría con un mensaje de error.



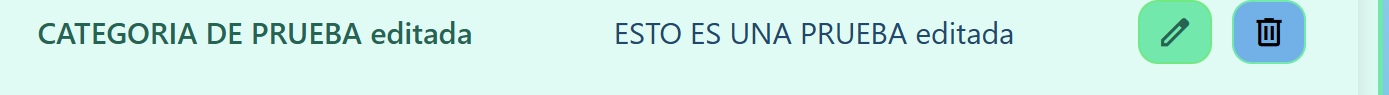
Para editar una categoría pincharemos en el icono del lápiz. Una vez hemos pinchado en el lápiz nos saldrá un formulario con los datos actuales de la categoría. Para editar un campo basta con borrarlo y escribir el nuevo valor o añadir información . También nos saldrá mensaje de éxito o error.

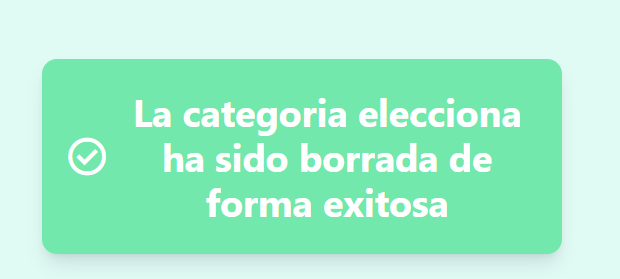






Para borrar una categoría solo habrá que pinchar en el icono del cubo de la basura y la categoría se borrará si no ha habido ningún problema. Si no ha sido posible borrarla también nos avisará el programa con un mensaje de error.

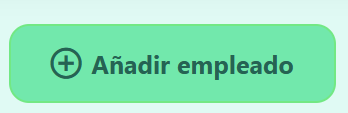




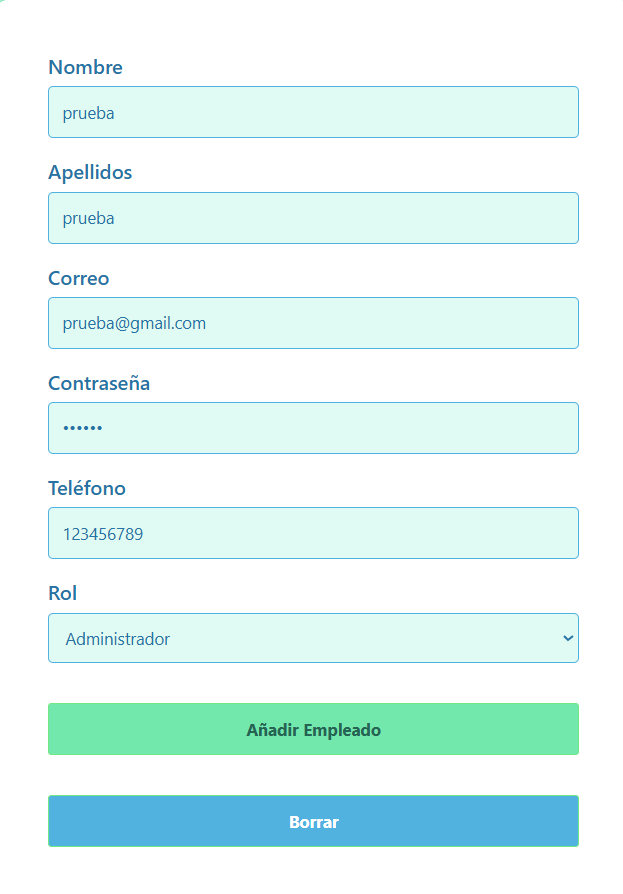
**Empleados:** En esta opción podremos añadir, eliminar o editar un empleado existente.



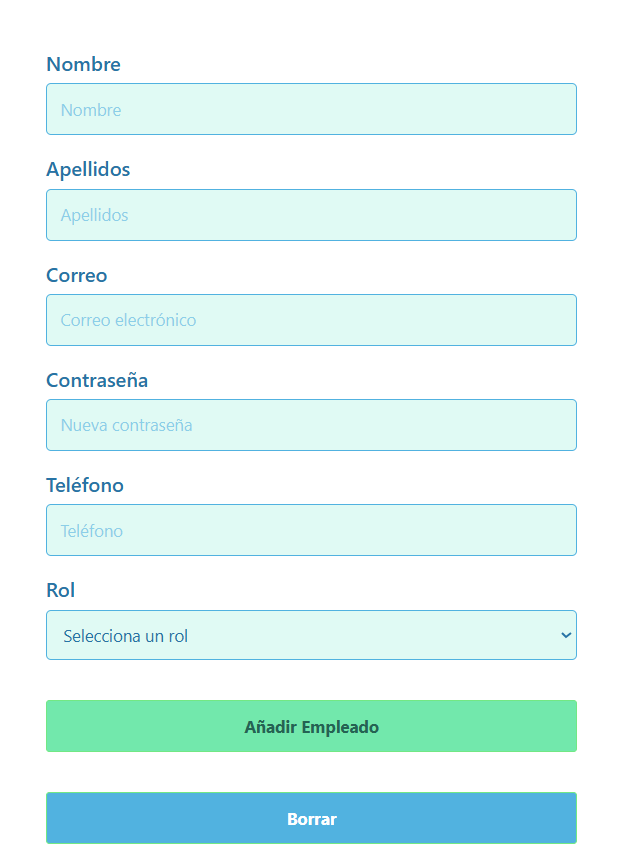
Para añadir un empleado pincharemos en la opción añadir empleado.

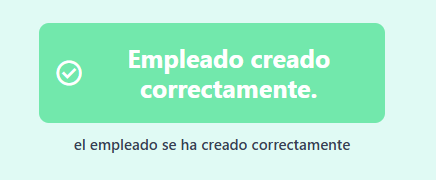


Una vez que hemos pinchado en añadir empleado nos saldrá un formulario donde podremos poner los datos del nuevo empleado. Para mayor privacidad y seguridad la contraseña que pongamos al nuevo empleado estará hasheada.

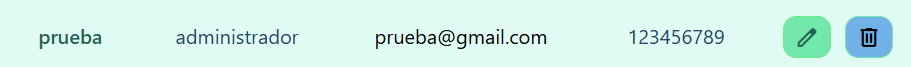


Una vez que ponemos todos los datos y pinchamos en añadir empleado el empleado se creará y nos saldrá un mensaje de éxito o de error dependiendo de si el empleado se ha creado correctamente o no

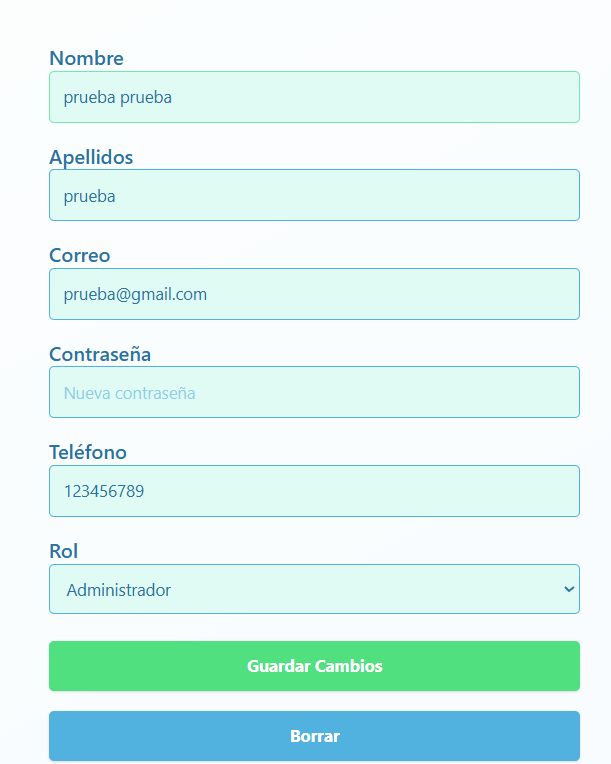


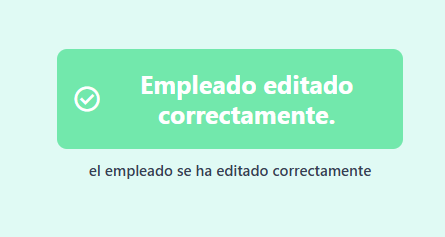


Para editar a un empleado pincharemos en el lápiz que hay en cada usuario.

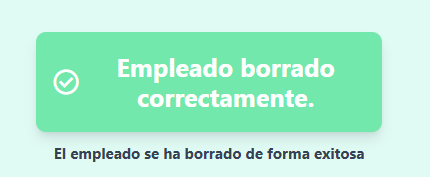


Cuando pinchemos en el lápiz aparecerá un formulario con los datos actuales (La contraseña no se verá por privacidad pero si podremos cambiarla). Para cambiar los datos pincharemos en el input que nos interese y editaremos los datos. Una vez que ya lo hemos editado pinchamos en guardar cambios.





Para borrar un usuario pincharemos en el cubo de basura del empleado que queremos borrar. Una vez hayamos pinchado el empleado se borrará si no ha habido ningún error.

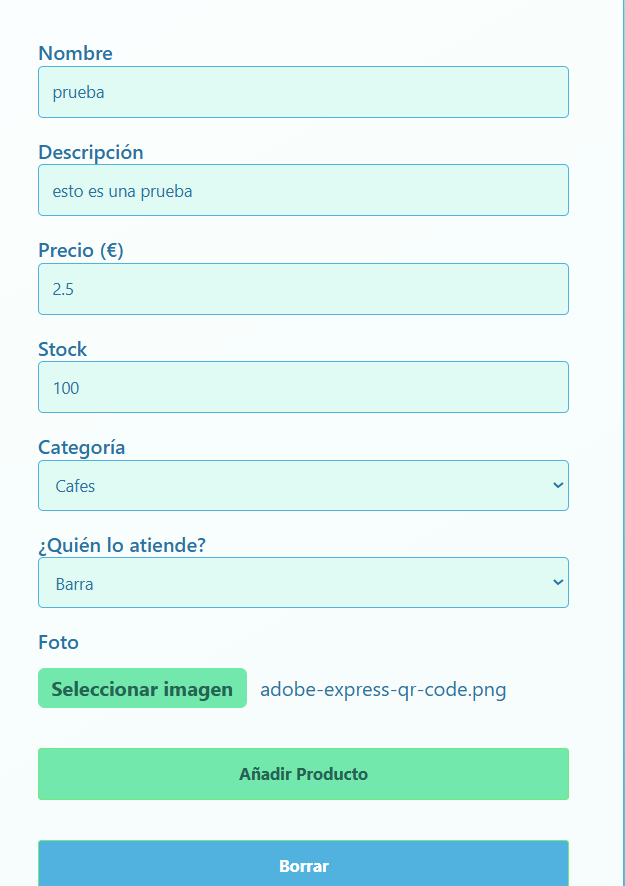


**Productos:** En esta opción podremos añadir, eliminar o editar un producto existente.

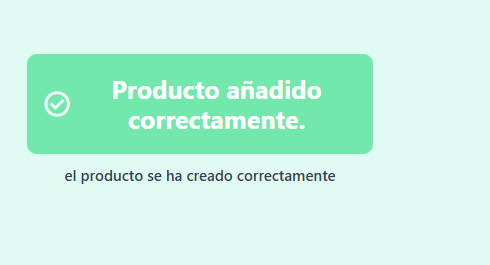


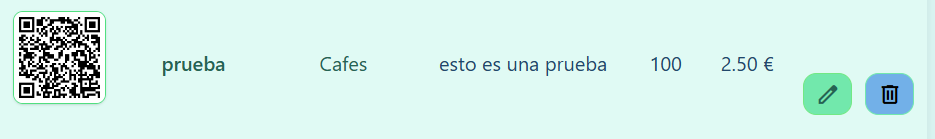
Para añadir un producto pincharemos en añadir producto.

Una vez que hemos pinchado en añadir producto nos saldrá un formulario para rellenar los datos del nuevo producto. También podremos adjuntar una foto, que es la que se verá en el menú.



Una vez que hemos rellenado los datos del nuevo producto pinchamos en añadir producto y se nos creará si no ha habido ningún error.

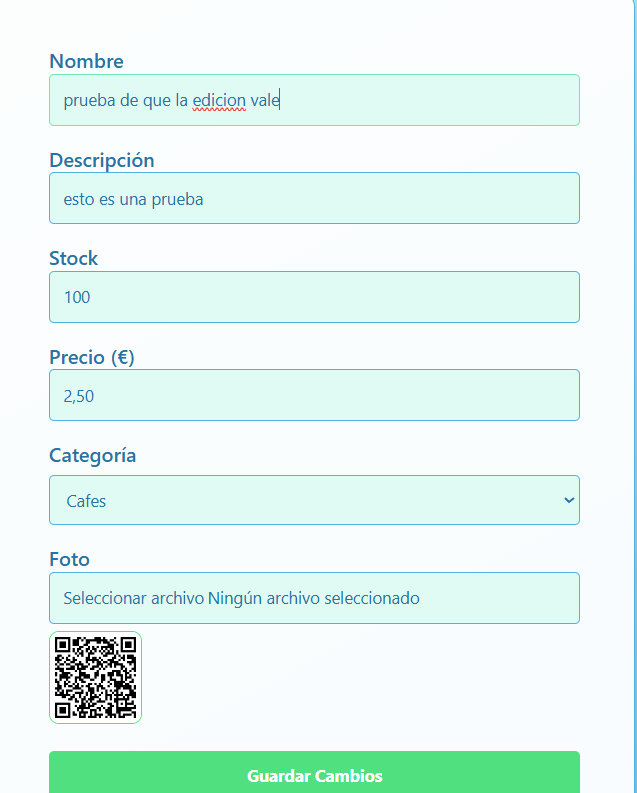


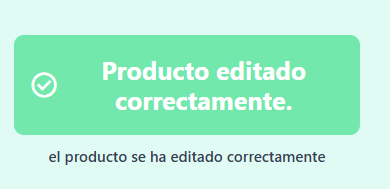


Para editar un producto pincharemos en el icono del lápiz.

C:\Users\Juan Pedro\AppData\Local\Packages\MicrosoftWindows.Client.Core_cw5n1h2txyewy\TempState\ScreenClip\{81AF11F8-CB79-4A83-8A81-E5DFAE59D79D}.png

Una vez que hayamos pinchado en el lápiz se nos abrirá un formulario con los datos actuales de ese producto. Para editarlo solo tenemos que borrar el campo entero o añadir información de la que ya ahí. Si queremos cambiar la foto habrá que adjuntar la nueva foto. Si pinchamos en guardar cambios y no ha habido ningún error el producto se habrá editado.



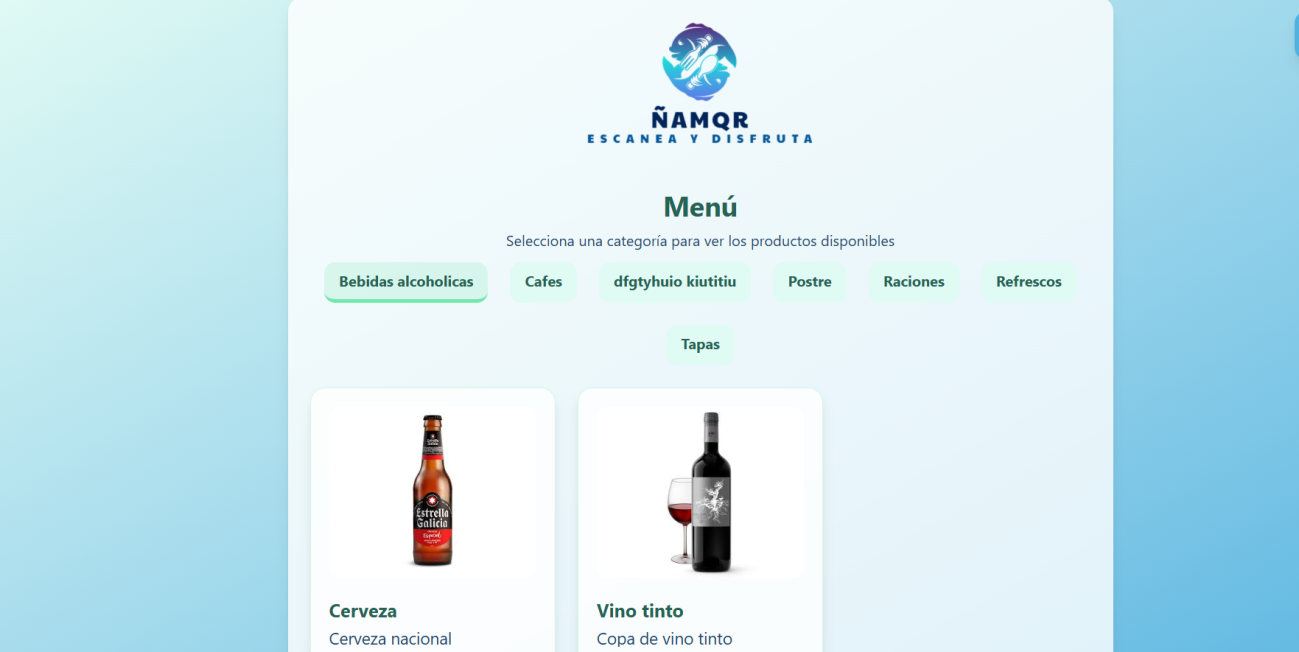


Para borrar un producto pincharemos en el cubo de basura y si no hay ningún error el producto se borrará.



En el menú de administración también podremos ver el menú como se quedaría si añadimos una categoría o un producto a una categoría.

C:\Users\Juan Pedro\AppData\Local\Packages\MicrosoftWindows.Client.Core_cw5n1h2txyewy\TempState\ScreenClip\{DC485383-F386-40CE-BF89-41BFD09DD150}.png



Para cerrar la sesión pinchamos en cerrar sesión que se encuentra en la cabecera como última acción. Si hacemos eso tendremos que volver a iniciar sesión si queremos ver el menú de administración.



## 7.2 MANUAL DEL COCINERO

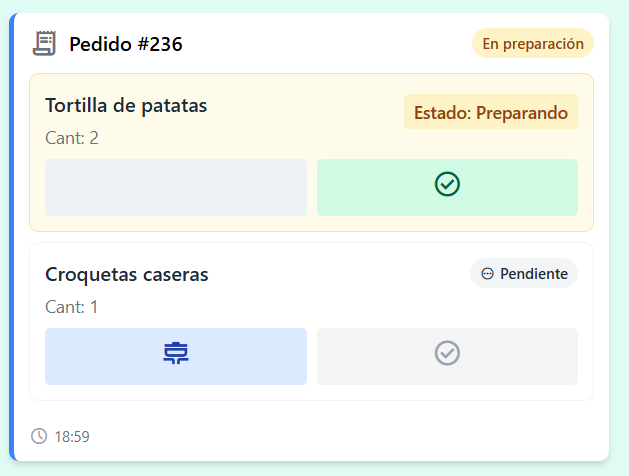
Primero tendremos que iniciar sesión como camarero o cocinero.

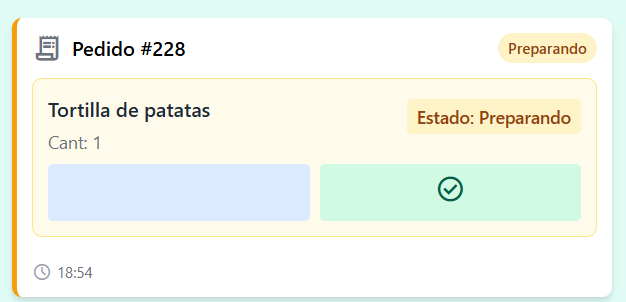


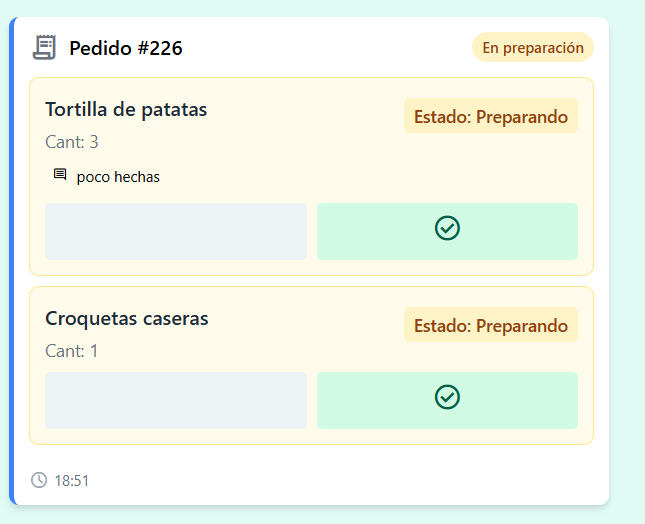
Una vez que hayamos iniciado sesión como cocinero veremos en la interfaz los pedidos que hay pendientes y en preparación.

Para cambiar el estado del producto para ponerlo en preparación pincharemos en el primer icono que aparece (si el producto ya está preparándose no cambiará el estado puesto que ya está en preparación. Para marcarlo como listo pincharemos en el segundo botón y el estado del producto cambiará a listo. Una vez que el producto esté listo automáticamente el producto desaparecerá de la interfaz. Cuando todos los productos de un pedido estén listos también se borrará el pedido entero de la interfaz. Ten en cuenta que si el servidor en tiempo real no funciona tendrás que estar actualizando la interfaz constantemente para que lleguen los pedidos.





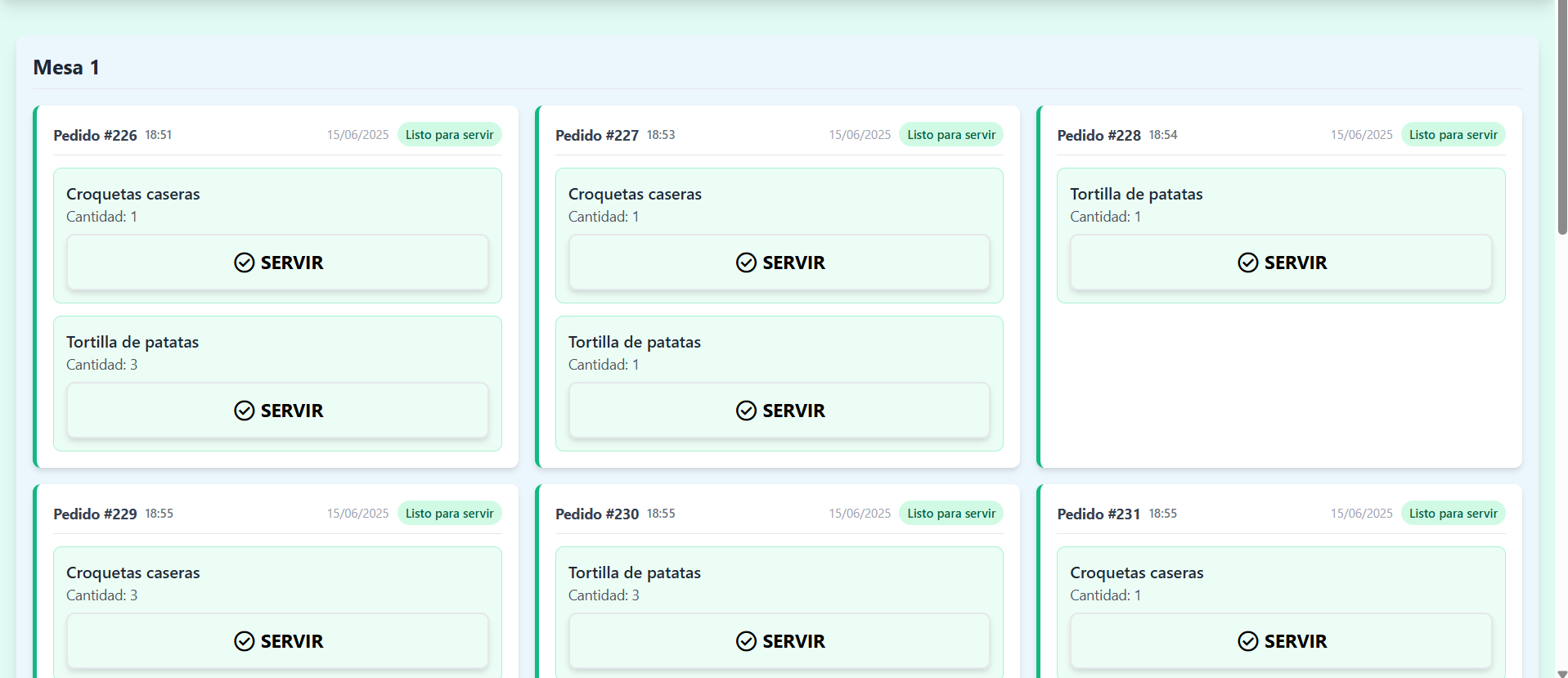




## 7.4 MANUAL DEL CAMARERO

Una vez que nos hayamos logueado como camarero nos saldrán todos los pedidos que hay que servir en las mesas. Una vez que un producto este servido lo único que tiene que hacer el camarero es pinchar en servir y automáticamente el producto desaparecerá de la interfaz. Ha y que tener en cuenta que si el servidor websocket falla por cualquier circunstancia habrá que recargar la página para ver los nuevos productos que hay que servir a las mesas.





## 7.4 MANUAL DEL USUARIO

El cliente deberá de escanear el código qr que se le proporciona en la mesa. Una vez que el cliente escanee el código la url le llevará directamente al menú del restaurante.



Dentro de la página del menú el cliente podrá navegar por las diferentes categorías mirando los productos. Para añadir un producto al carrito habrá que seleccionar la cantidad que quiera el cliente y pinchar en añadir. Una vez que añadas el producto el programa te pondrá un mensaje de confirmación.



Una vez hayamos terminado de pedir pinchamos en el logo del carrito y hay nos saldrá el resumen de todo el pedido que queremos realizar.



Si nos hemos equivocado y no queremos un producto en concreto podemos pinchar en el icono del cubo que hay en cada producto para eliminarlo individualmente. Si por el contrario queremos eliminar todos los productos del carrito pinchamos en vaciar carrito. También podemos poner observaciones individuales de cada producto y la observación general del pedido. Una vez que cercioremos que está todo bien pinchamos en finalizar pedido. Al pinchar en finalizar pedido nos saldrá una ventana con el resumen del pedido que hemos hecho. Para volver al menú por si queremos pedir algo más pinchamos en volver al menú.

