



**Universidad Tecnológica Nacional**  
**Fac. Reg. Concepción del Uruguay**

**Ingeniería en Sistemas de Información**

**Desarrollo Cliente - Servidor**

**Proyecto Final: Click Order**

**Docentes:**

Ing. Adrian, Tournour.  
Ing. Lucas Schneider.

**Alumnos:** Machena, Maria Belen

**Fecha de presentación:** 25/04/2025

## Introducción

Para el proyecto final de la materia **Cliente-Servidor**, se decidió desarrollar una aplicación web de pedidos de comidas denominada "Click Order". El objetivo principal fue crear una plataforma accesible, moderna y funcional que permita a los usuarios realizar pedidos de forma sencilla, visualizando la información en tiempo real y aprovechando las tecnologías actuales tanto del lado del cliente como del servidor.

La aplicación fue construida bajo una arquitectura cliente-servidor moderna, priorizando el desarrollo multiplataforma. Para el **frontend** se utilizó una combinación de Ionic, Angular, Capacitor, TypeScript y JavaScript, tecnologías que permiten generar una experiencia de usuario fluida, adaptada tanto a dispositivos móviles como a navegadores web. Se implementó una interfaz limpia e intuitiva con navegación lateral, vistas dinámicas, y compatibilidad con distintos tamaños de pantalla.

Del lado del **backend**, se utilizó Firebase, aprovechando su ecosistema completo de servicios. Se integró **Firebase Firestore** como base de datos en tiempo real para almacenar información de usuarios, pedidos y productos disponibles. Además, se empleó Firebase Authentication para gestionar el registro e inicio de sesión seguro de los usuarios. Para garantizar la seguridad de los datos, se definió una estructura de reglas de seguridad personalizadas en Firebase que limitan tanto la lectura como la escritura según el rol del usuario y su autenticación.

Como funcionalidad destacada, se integró la **API de Google Maps** para permitir visualizar direcciones de entrega, ubicar al repartidor y generar rutas entre puntos. Esto mejora notablemente la experiencia del usuario, brindando transparencia y eficiencia en el seguimiento del pedido.

El sistema fue diseñado considerando aspectos de escalabilidad, seguridad y facilidad de uso. También se contempló la posibilidad de extenderlo en el futuro, agregando nuevas funcionalidades como notificaciones push, historial de pedidos y panel de administración.

## Arquitectura



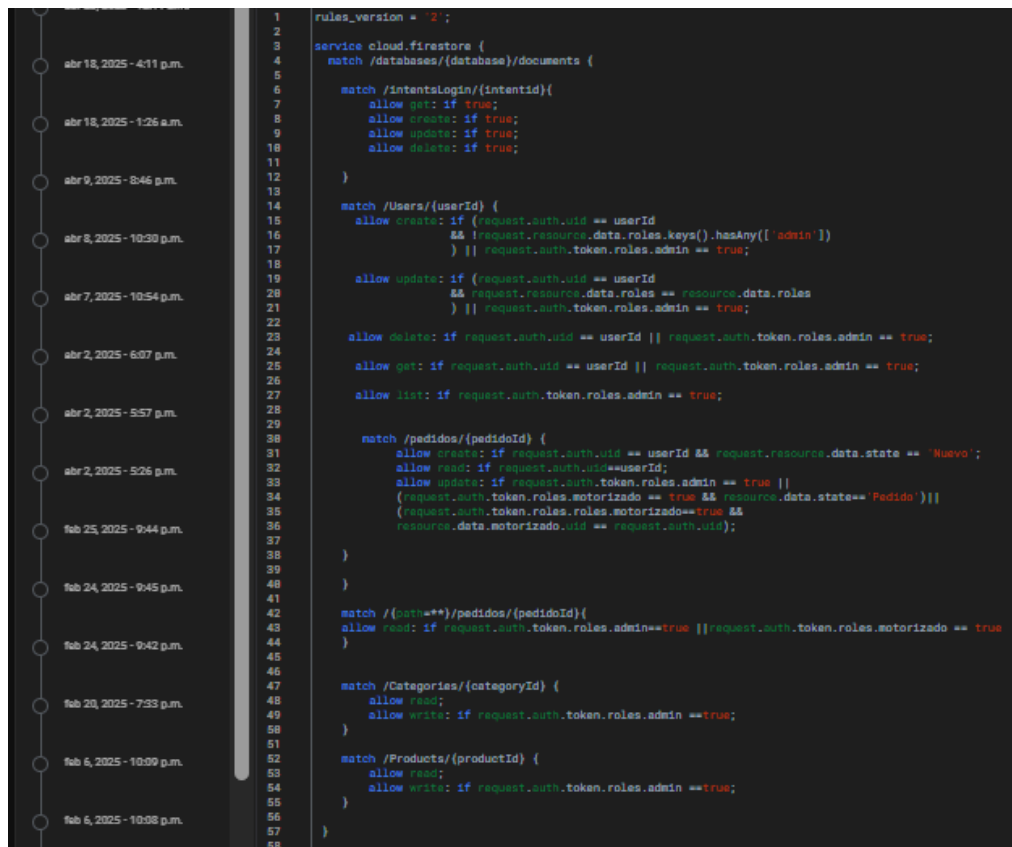
## Seguridad de la base de datos Firestore

La aplicación implementa un sistema de **reglas de seguridad en Firebase Firestore** para controlar el acceso a los datos almacenados, asegurando que solo los usuarios autorizados puedan leer o modificar la información según su rol dentro del sistema.

Estas reglas permiten proteger tanto los datos personales de los usuarios como las operaciones asociadas a pedidos, productos y categorías. Por ejemplo, los usuarios solo pueden acceder y modificar su propia información, mientras que los administradores tienen permisos extendidos para visualizar, listar y gestionar usuarios, productos y pedidos.

En el caso de los **pedidos**, los usuarios autenticados pueden crear pedidos nuevos y ver únicamente los que les pertenecen. Por su parte, los **motorizados** (repartidores) y **administradores** tienen permisos especiales para actualizar pedidos según su estado o asignación. Además, se restringe que los usuarios puedan asignarse roles como "admin", garantizando así una jerarquía de permisos segura y controlada.

Las reglas también definen que los productos y categorías pueden ser leídos por cualquier usuario, pero sólo los administradores están autorizados para crear o modificar esta información.



```
1 rules_version = '2';
2
3 service cloud.firestore {
4   match /databases/{database}/documents {
5
6     match /intentsLogin/{intentId}{
7       allow get: if true;
8       allow create: if true;
9       allow update: if true;
10      allow delete: if true;
11    }
12
13
14    match /Users/{userId} {
15      allow create: if (request.auth.uid == userId
16        && !request.resource.data.roles.keys().hasAny(['admin']))
17        ) || request.auth.token.roles.admin == true;
18
19      allow update: if (request.auth.uid == userId
20        && request.resource.data.roles == resource.data.roles
21        ) || request.auth.token.roles.admin == true;
22
23      allow delete: if request.auth.uid == userId || request.auth.token.roles.admin == true;
24
25      allow get: if request.auth.uid == userId || request.auth.token.roles.admin == true;
26
27      allow list: if request.auth.token.roles.admin == true;
28
29
30      match /pedidos/{pedidoId} {
31        allow create: if request.auth.uid == userId && request.resource.data.state == 'Nuevo';
32        allow read: if request.auth.uid==userId;
33        allow update: if request.auth.token.roles.admin == true ||
34          (request.auth.token.roles.motorizado == true && resource.data.state=='Pedido') ||
35          (request.auth.token.roles.roles.motorizado==true &&
36            resource.data.motorizado.uid == request.auth.uid);
37      }
38    }
39
40
41    match /{path=**}/pedidos/{pedidoId}{
42      allow read: if request.auth.token.roles.admin==true || request.auth.token.roles.motorizado == true
43    }
44
45
46    match /Categories/{categoryId} {
47      allow read;
48      allow write: if request.auth.token.roles.admin ==true;
49    }
50
51    match /Products/{productId} {
52      allow read;
53      allow write: if request.auth.token.roles.admin ==true;
54    }
55
56  }
57
58 }
```

## Funcionamiento de la Aplicacion

La aplicación fue pensada y desarrollada como un sistema web de gestión de pedidos de comida, con un enfoque práctico, organizado y adaptable a distintos dispositivos. Está diseñada para operar con **tres tipos de roles de usuario: Administrador, Cliente y Motorizado**, cada uno con funcionalidades específicas que garantizan una experiencia fluida y controlada en cada etapa del pedido.

Una de las principales ventajas del sistema es que **es accesible tanto desde una computadora como desde un dispositivo móvil** a través de cualquier navegador web moderno. Gracias a que fue desarrollada con **ionic y Angular**, la aplicación cuenta con una interfaz **responsive**, lo que significa que se adapta automáticamente al tamaño de pantalla, brindando una experiencia de uso fluida tanto en celulares como en tablets o PCs sin necesidad de instalar ninguna app.

### Inicio (Home)

Al ingresar a la plataforma, los usuarios son recibidos con una **página de inicio (home)** donde pueden:

- **Iniciar sesión** si ya cuentan con una cuenta registrada.

- **Registrarse** en caso de ser nuevos usuarios.

El registro puede realizarse de dos maneras:

- Mediante **correo electrónico y contraseña**, completando un formulario con los datos personales necesarios.
- A través de **inicio de sesión con Google**, aprovechando la autenticación segura que ofrece Firebase, lo que facilita el proceso de registro e ingreso sin necesidad de recordar contraseñas.

Una vez autenticados, los usuarios serán redirigidos a la sección correspondiente según su rol (Administrador, Cliente o Motorizado), desde donde podrán acceder a las funcionalidades diseñadas para cada tipo de usuario.

#### ♦ **Rol Administrador**

El **Administrador** representa al dueño de la tienda o negocio. Tiene acceso a un panel completo desde donde puede:

- **Cargar productos** a la tienda, asignándolos a una categoría, agregando una descripción, imagen y precio.
- **Visualizar todos los pedidos** realizados por los clientes a través de la plataforma.
- **Preparar y gestionar pedidos**, asignándoles un estado y viendo como va cambiando este.
- **Asignar estados a los pedidos**, como por ejemplo "Nuevo", "Pedido", "En Camino" y "Entregado".
- **Administrar usuarios**, visualizando todos los registrados en el sistema y pudiendo modificar sus roles según necesidad (por ejemplo, convertir un cliente en motorizado o viceversa).

Esto permite que el administrador tenga un control total sobre el negocio de forma remota, pudiendo gestionar pedidos en tiempo real desde cualquier lugar.

#### ♦ **Rol Cliente**

El **Cliente** es el usuario que realiza un pedido. El flujo pensado para este rol es el siguiente:

1. **Navega por la tienda** visualizando los productos disponibles, organizados por categorías.
2. **Agrega productos al carrito**, sumando los que desea comprar.
3. Una vez que termina de seleccionar, **completa un formulario con datos clave**:
  - Número de teléfono de contacto.
  - Fecha deseada para recibir el pedido.
  - Dirección exacta de entrega.
  - Nota adicional (opcional).
4. Al confirmar el pedido, este se guarda en la base de datos con el **estado "Nuevo"**.
5. Posteriormente, el cliente podrá **ver el estado de su pedido en tiempo real**, con actualizaciones automáticas, y seguir su progreso una vez que esté en camino.

Además, desde su cuenta el cliente podrá revisar sus pedidos anteriores y recibir notificaciones visuales del estado actualizado de sus órdenes.

#### ♦ **Rol Motorizado**

El **Motorizado** es la persona encargada de **entregar los pedidos**. Tiene acceso a una vista especializada donde:

- Puede **ver todos los pedidos disponibles** en estado "Pedido", es decir, los que fueron preparados por el administrador y aún no han sido tomados.
- Puede **asignarse un pedido**, haciéndose responsable del mismo.
- Una vez asignado, cambia el estado del pedido a **"En Camino"**, lo que automáticamente actualiza la vista del cliente, permitiéndole hacer un seguimiento en vivo del trayecto del motorizado gracias a la integración con **Google Maps**.
- Tiene acceso a la **dirección de entrega y nota adicional** del cliente para facilitar la entrega.
- Al finalizar la entrega, debe cambiar el estado del pedido a **"Entregado"**, cerrando así el ciclo del pedido.

Gracias a esta estructura, el motorizado puede gestionar sus entregas desde el mismo celular, visualizando los pedidos, rutas y direcciones sin necesidad de usar ninguna otra aplicación externa.

#### ♦ **Accesibilidad y Experiencia de Usuario**

La aplicación está diseñada para ofrecer una experiencia clara, rápida y sin complicaciones. No requiere instalación previa, ya que se puede utilizar desde cualquier navegador web moderno como Chrome, Safari o Firefox, tanto en PC como en dispositivos móviles. Esto facilita el acceso para todos los usuarios, sin importar el sistema operativo o el tipo de dispositivo que utilicen.

También se pensó en la posibilidad de **extender la plataforma a una Progressive Web App (PWA)**, lo que permitiría instalarla como si fuera una app nativa en el celular, sin pasar por tiendas de aplicaciones.

### **Requisitos Funcionales y No Funcionales**

Para garantizar el correcto diseño, desarrollo e implementación del sistema, se definieron un conjunto de requisitos funcionales y no funcionales

#### **Requisitos Funcionales**

##### ♦ **Generales**

- **RF-01:** El sistema debe permitir el registro e inicio de sesión de usuarios mediante correo electrónico y mediante cuenta de Google.
- **RF-02:** El sistema debe identificar el rol del usuario autenticado (Administrador, Cliente o Motorizado) y mostrarle la vista correspondiente.
- **RF-03:** El sistema debe registrar la fecha de cada pedido y mantener un historial de su estado.

##### ♦ **Rol Administrador**

- **RF-04:** El administrador debe poder cargar productos con nombre, descripción, categoría, imagen y precio.
- **RF-05:** El administrador debe poder editar o eliminar productos existentes.
- **RF-06:** El administrador debe poder crear, ver y modificar categorías.
- **RF-07:** El administrador debe poder ver todos los pedidos realizados por los clientes.

- **RF-08:** El administrador debe poder cambiar el estado de los pedidos: Nuevo, Pedido, En Camino, Entregado.
- **RF-09:** El administrador debe poder ver la lista de usuarios registrados.
- **RF-10:** El administrador debe poder cambiar el rol de un usuario (por ejemplo, de Cliente a Motorizado o viceversa).

#### ♦ **Rol Cliente**

- **RF-12:** El cliente debe poder visualizar productos por categoría.
- **RF-13:** El cliente debe poder agregar productos al carrito.
- **RF-14:** El cliente debe poder completar su pedido con dirección, número de teléfono, fecha deseada y nota adicional.
- **RF-15:** El cliente debe poder confirmar el pedido, el cual queda registrado en estado "Nuevo".
- **RF-16:** El cliente debe poder ver el estado del pedido en tiempo real.
- **RF-17:** El cliente debe poder seleccionar su dirección exacta de entrega a través de un mapa interactivo utilizando la API de Google Maps.
- **RF-18:** El cliente debe poder visualizar el recorrido del pedido en tiempo real mediante el mapa integrado que utiliza la API de Google Maps.

#### ♦ **Rol Motorizado**

- **RF-20:** El motorizado debe poder ver los pedidos disponibles en estado "Pedido".
- **RF-21:** El motorizado debe poder aceptar un pedido y asumir la entrega.
- **RF-22:** El motorizado debe poder visualizar la dirección de entrega y nota del cliente.
- **RF-23:** El motorizado debe poder cambiar el estado del pedido a "En Camino" y luego a "Entregado".
- **RF-24:** El motorizado debe poder ver su lista de pedidos asignados.



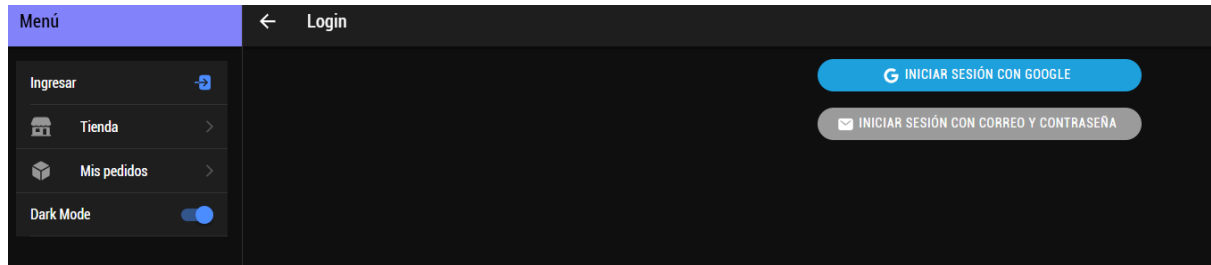
- **RF-25:** El motorizado debe poder visualizar en el mapa la dirección del cliente mediante la API de Google Maps para optimizar la entrega.
- **RF-26:** El motorizado debe poder seguir la mejor ruta hacia el cliente desde su ubicación actual usando el mapa integrado.

## **Requisitos No Funcionales**

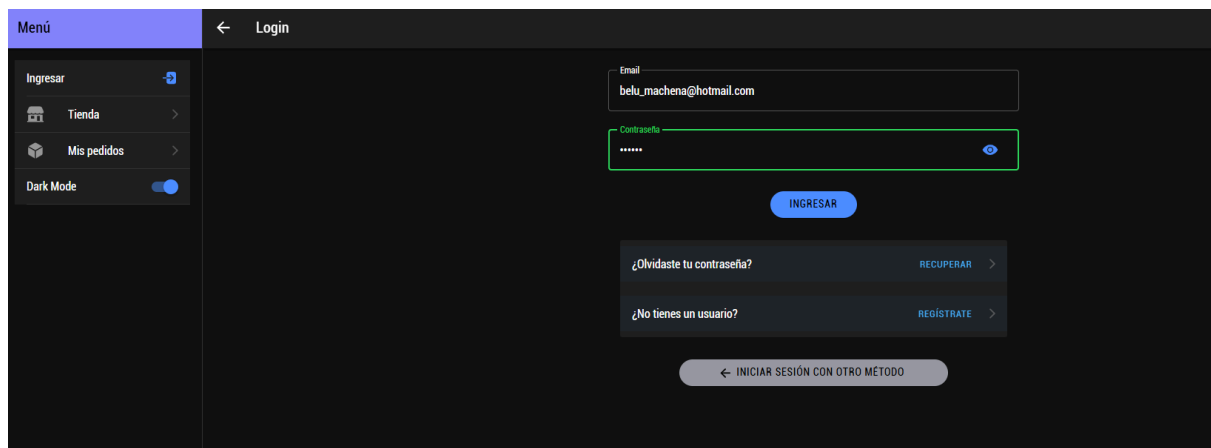
- **RNF-01:** El sistema debe utilizar Firebase Authentication para asegurar que solo usuarios registrados accedan a sus funcionalidades.
- **RNF-02:** Las reglas de seguridad de Firestore deben restringir el acceso a documentos según el rol del usuario.
- **RNF-03:** Los pedidos, usuarios y productos deben estar protegidos contra escritura o lectura no autorizada.
- **RNF-04:** El sistema debe contar con una interfaz responsive, adaptada a diferentes tamaños de pantalla (PC, tablet y celular).
- **RNF-05:** La navegación debe ser intuitiva y accesible, mostrando menús y funciones según el rol del usuario.
- **RNF-06:** El sistema debe responder rápidamente a las acciones del usuario (crear pedido, cambiar estado, etc.).
- **RNF-07:** La interacción con la base de datos debe ser en tiempo real, mostrando actualizaciones sin necesidad de recargar la página.
- **RNF-08:** La aplicación debe ser accesible desde cualquier navegador moderno sin necesidad de instalación.
- **RNF-09:** El sistema debe poder soportar múltiples usuarios conectados al mismo tiempo sin afectar su rendimiento.
- **RNF-10:** La base de datos debe poder crecer para soportar mayor cantidad de productos, usuarios y pedidos sin pérdida de rendimiento.
- **RNF-11:** El sistema debe integrarse con la API de Google Maps para permitir la selección de direcciones, visualización de rutas y seguimiento de pedidos.
- **RNF-12:** El sistema debe actualizar automáticamente la ubicación del motorizado en el mapa sin necesidad de recargar la página.
- **RNF-13:** El sistema debe manejar errores de conexión con la API de Google Maps y notificar al usuario en caso de fallos sin interrumpir su uso.

## Imágenes de la Aplicación Web

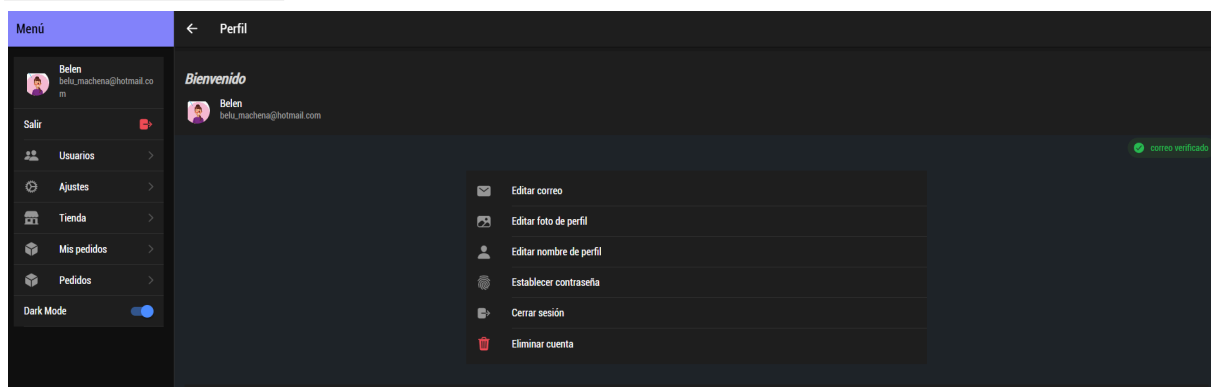
Inicio para loguearse



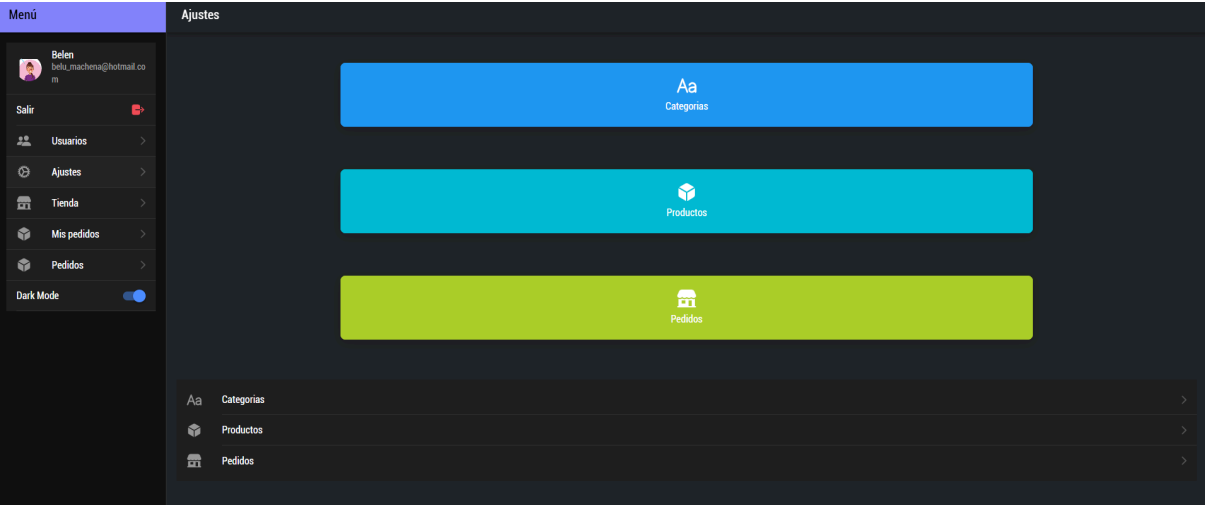
## Iniciar Sesión



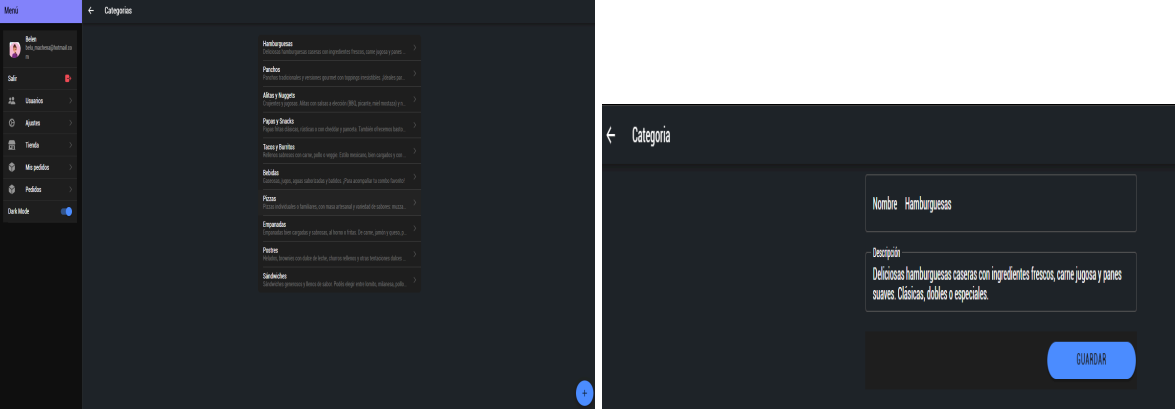
## Rol Administrador



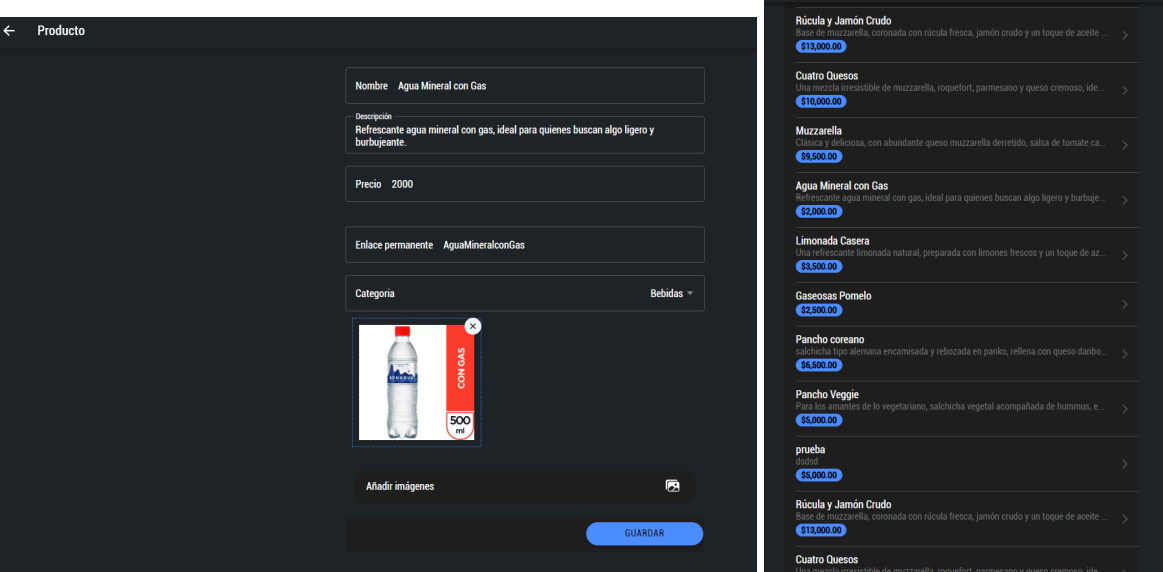
## Ajustes

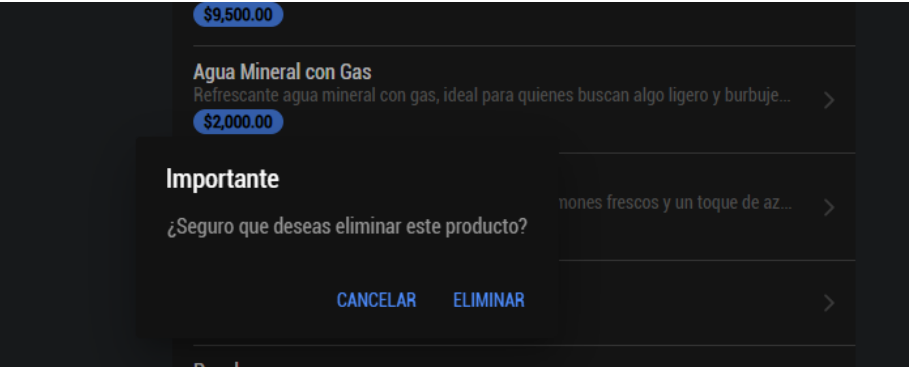


**Categorías:** Modificar, agregar o eliminar

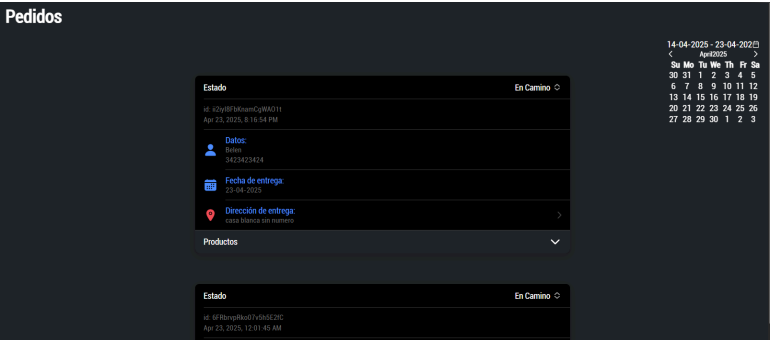
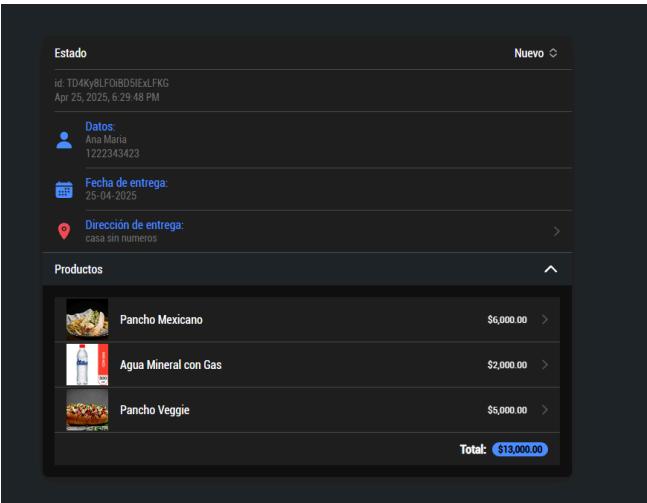


**Productos:** Modificar y agregar productos a la tienda

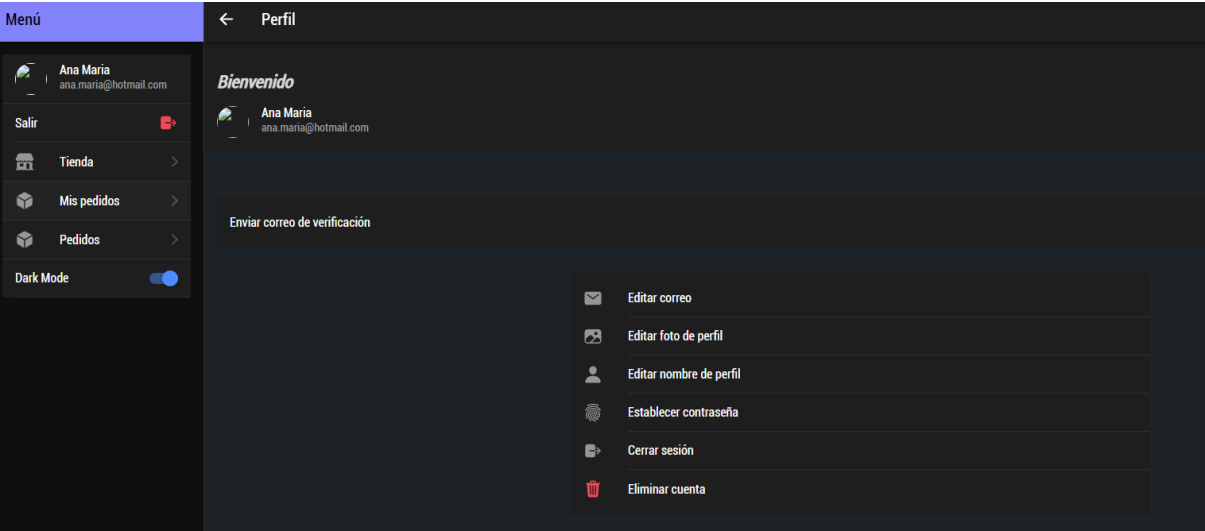




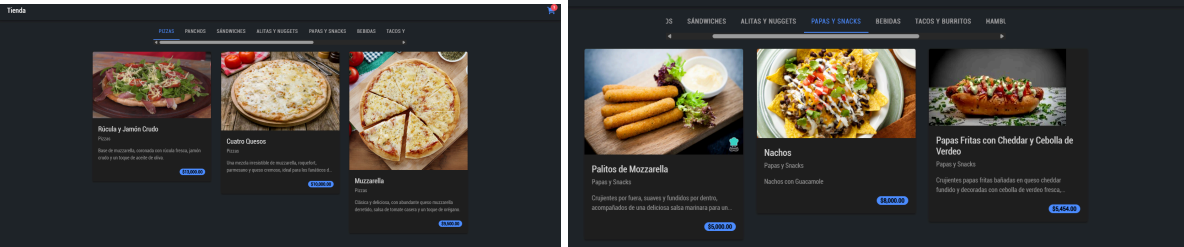
**Pedidos:** Visualizar los pedidos nuevos y filtrar los pedidos por fecha.



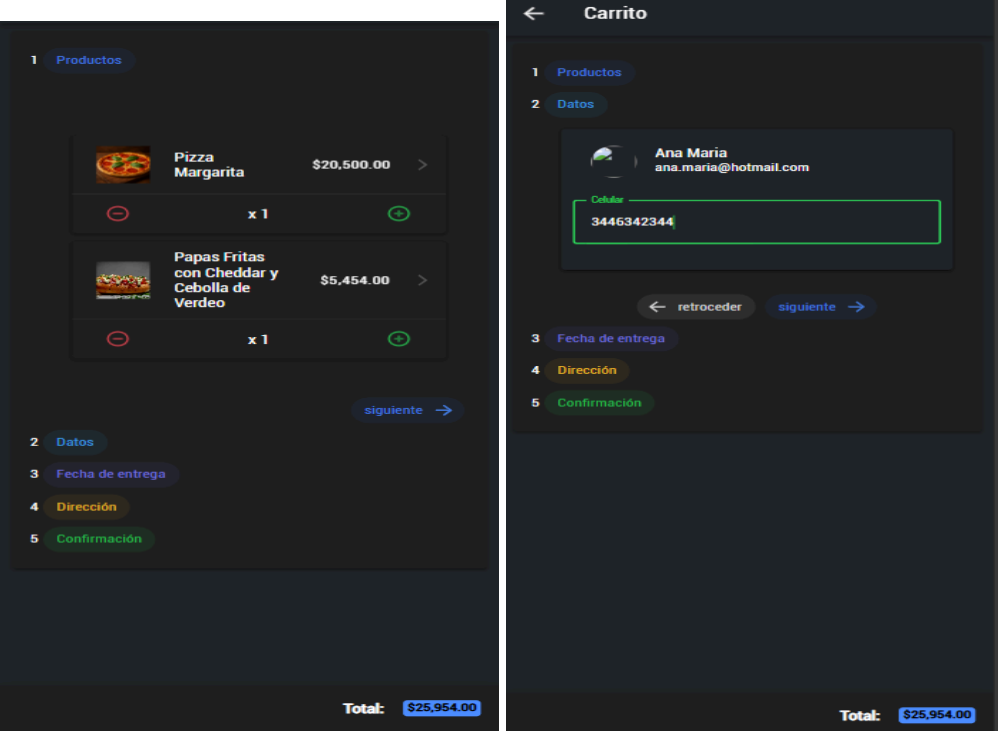
**Rol Cliente:**

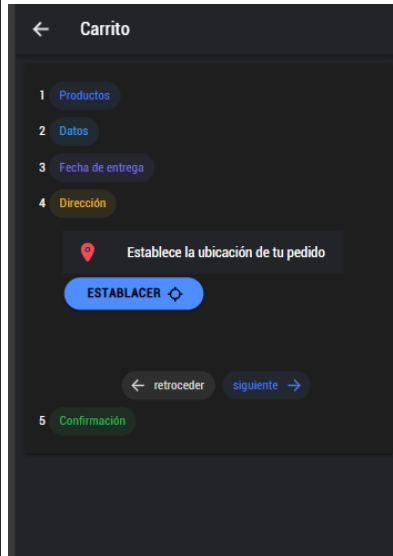
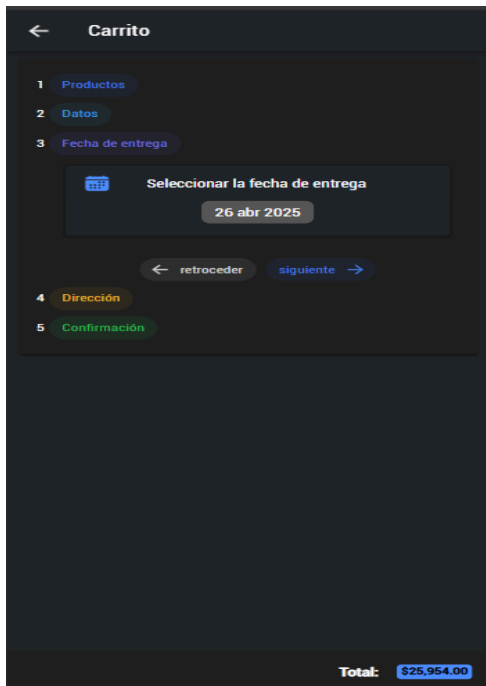


Visualizar la tienda

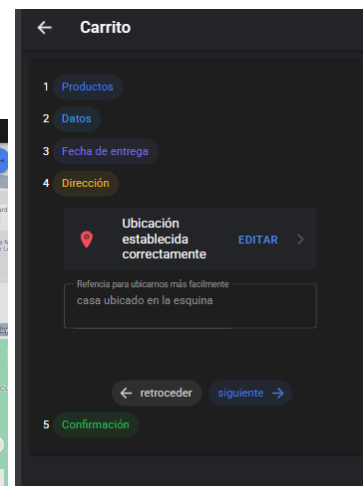
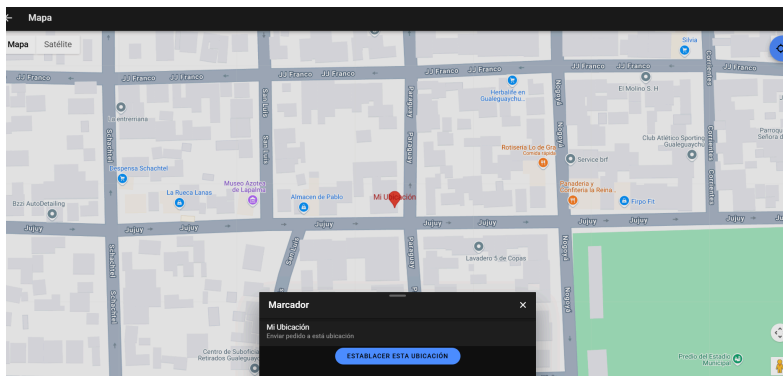


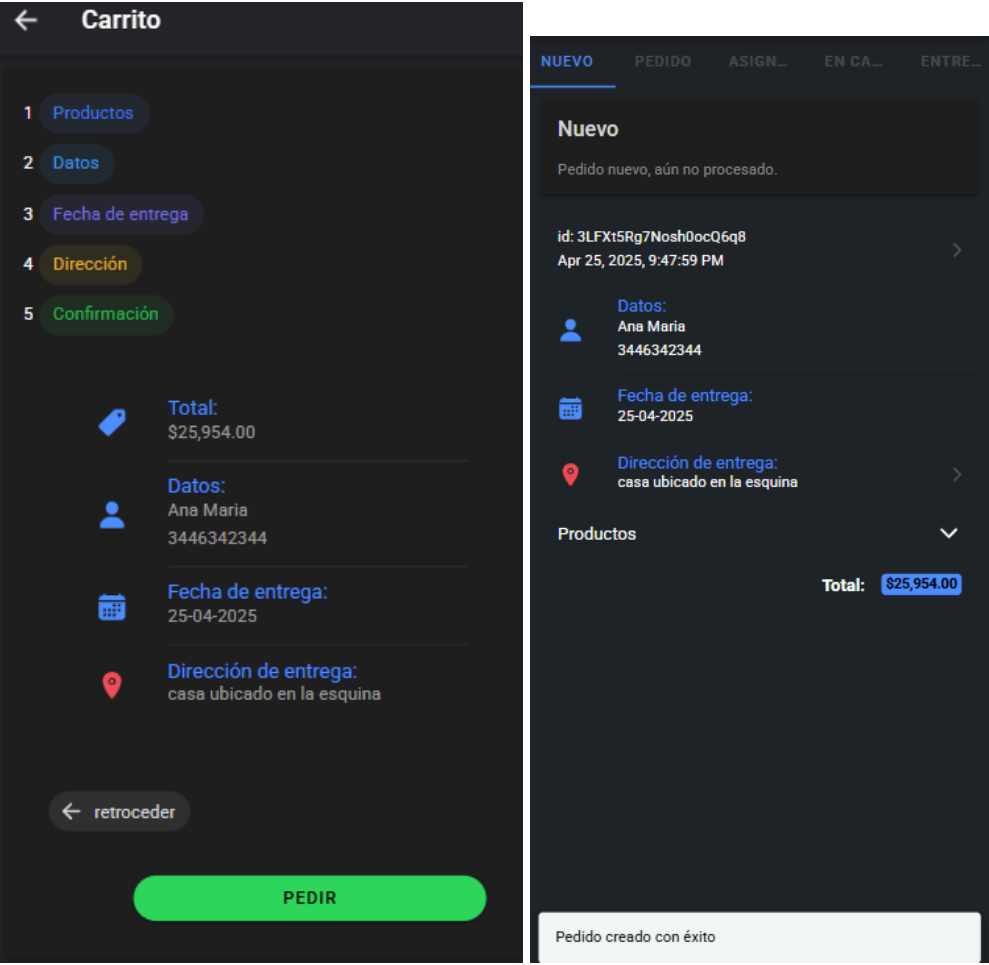
Hacer Pedido:



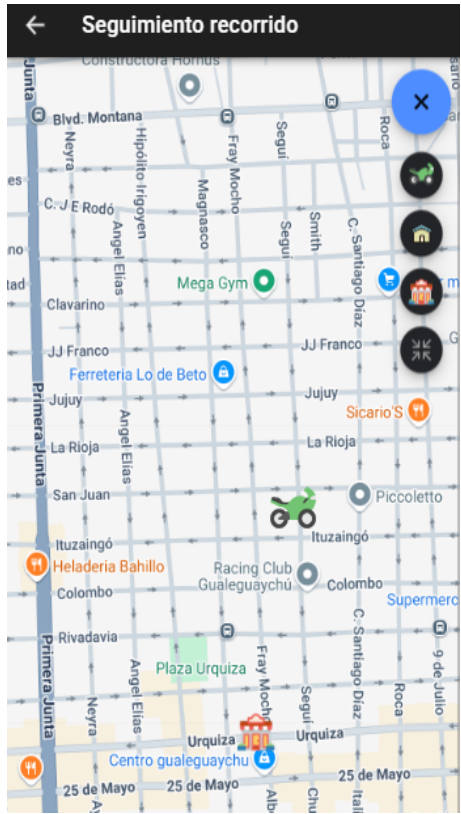
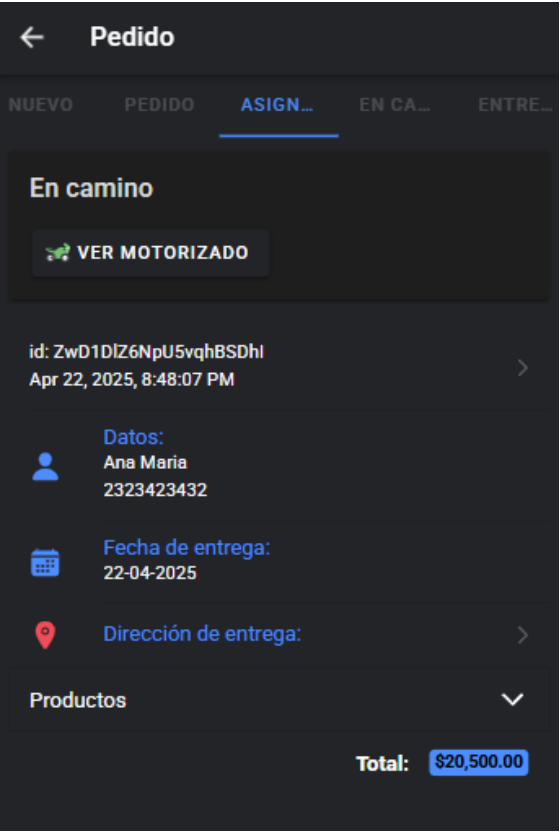


## Consumo de Api google Maps

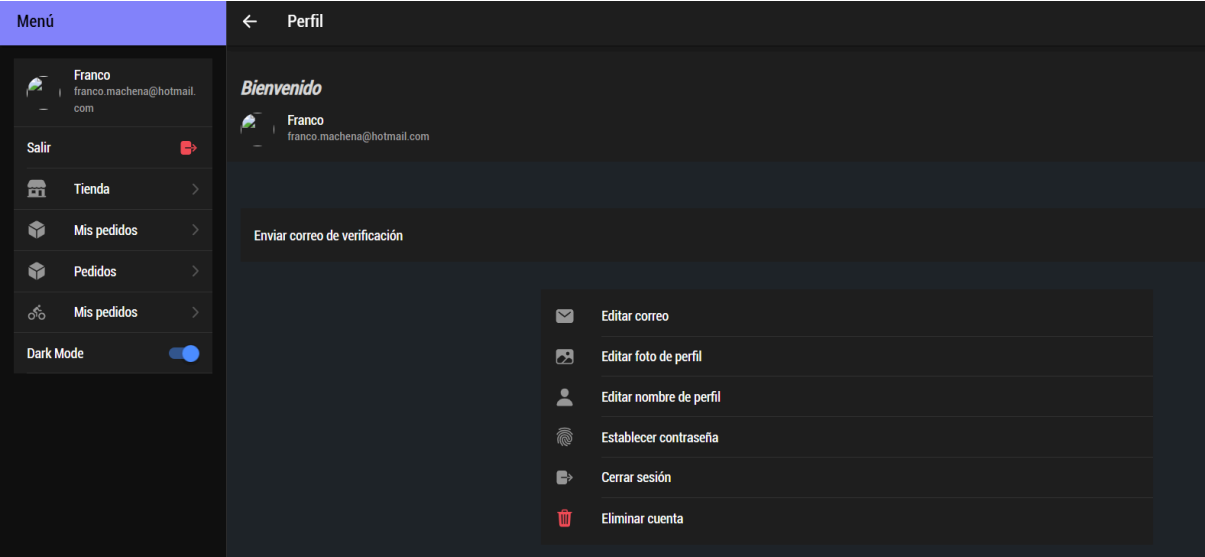




Ver estados de los Pedidos y donde viene el mismo:



Rol Motorizado:



Tomar pedido.



Pedidos

Estado

Libre

Hacerme cargo del pedido

>

Id: 3LFXt5Rg7Nosh0ocQ6q8

Apr 25, 2025, 9:47:59 PM

Datos:

Ana Maria

3446342344

Fecha de entrega:

25-04-2025

Dirección de entrega:

>

casa ubicado en la esquina

Productos

>

Estado

Libre

Hacerme cargo del pedido

>

Id: T8Mv6B18B058EALFY0

Apr 25, 2025, 6:29:48 PM

Datos:

Ana Maria

1222343423

Fecha de entrega:

25-04-2025

Dirección de entrega:

>

casa sin numeros

Productos

>

Estado

Ocupado

Iniciar recorrido

>

Id: 3LFXt5Rg7Nosh0ocQ6q8

Apr 25, 2025, 9:47:59 PM

Datos:

Ana Maria

3446342344

Fecha de entrega:

25-04-2025

Dirección de entrega:

>

casa ubicado en la esquina

Productos

>

← Recorrido

Estado

☐ Asignado

☒ En Camino

☐ Entregado

CANCEL

OK

