

**Informe Final Capstone:**

**ShopIA - Chatbot de IA integrado al e-commerce de Tienda Aurora**

Fecha: 02/12/2025

**Integrantes: Tomás Figueroa, Vicente Morales, Matías Hernández**

**Profesor(a) encargado(a): Rocío Contreras Águila**

**Asignatura: CAPSTONE\_007D**

**Carrera: Ingeniería en Informática**

**Índice**

[**Introducción: 3**](#_heading=h.i95nnmb4nfb3)

[**Contexto del proyecto: 5**](#_heading=h.xn116l76h0nd)

[**Justificación del proyecto: 6**](#_heading=h.u4qmhpeah0zy)

[**Áreas de desempeño 7**](#_heading=h.rvdnar625cud)

[**Competencias 7**](#_heading=h.1kxorxpe43mg)

[**Relevancia del proyecto APT 8**](#_heading=h.4dwx5yb3ao14)

[**Objetivos 9**](#_heading=h.dp98xj2dusxp)

[**Objetivo General Alcanzado: 9**](#_heading=h.4oa2onfaofky)

[**Objetivos específicos cumplidos: 10**](#_heading=h.44fqxj4kwd)

[**Visión General del Proyecto finalizado: 12**](#_heading=h.b3c3ae1ih29m)

[**2.- Requerimientos abordados: 13**](#_heading=h.vd6h9ewvrnb0)

[**3.- A quién beneficia la solución:  
  
El proyecto finalizado beneficia directamente a dos grupos involucrados: 14**](#_heading=h.qfe7o6jzasif)

[**Metodología 15**](#_heading=h.nuk7nfmzq647)

[**Desarrollo 16**](#_heading=h.7t7g4zmpzn9d)

[Etapas y actividades: 16](#_heading=h.qldhzorn0uwu)

[Dificultades y facilitadores: 18](#_heading=h.8ozx0glgkp16)

[**Evidencias: 19**](#_heading=h.iwhje1y2809m)

[1.- Modelo de datos 19](#_heading=h.ordnd9jq8u7n)

[2.- Arquitectura 20](#_heading=h.p9ghxyauw46a)

[**3.- Tecnologías 21**](#_heading=h.mvz9a32g961c)

[**4.- Resultado del producto final (imágenes) 22**](#_heading=h.gh1fzm34g5ot)

[Home 22](#_heading=h.shye0pwnet3r)

[Colecciones 23](#_heading=h.yn38tgazb4w7)

[Productos 24](#_heading=h.a0gxv2im265s)

[Ofertas 25](#_heading=h.trodp4dm8cac)

[Carrito de compra 26](#_heading=h.nfthh7mu9km)

[“Nuestras Sucursales” 27](#_heading=h.dj5o5coiffv7)

[Login y Registro 28](#_heading=h.7iqjy2yrp30q)

[Panel de Administrador 30](#_heading=h.x2i24uixf47k)

[Pedidos (Panel de Administrador) 31](#_heading=h.jr15envejvcu)

[Productos (Panel de Administrador) 33](#_heading=h.l4flrct08ot0)

[Reportes (Panel de Administrador) 35](#_heading=h.u27e40p7x24h)

[Códigos de descuento (Panel de Administrador) 36](#_heading=h.fxc27mbr1nhp)

[Configuración 37](#_heading=h.2j08tzen4hw6)

[Vista principal del panel de gestión de la información de la administración de Aurora 38](#_heading=h.p1tb9wkpw0ay)

[CRUD de Productos 39](#_heading=h.ny5y4b52kbfd)

[**Intereses y proyecciones profesionales 45**](#_heading=h.l9iobldg5x9u)

[**Conclusión 46**](#_heading=h.v6zy2mwce1p5)

# **Introducción:**

Este informe detalla el desarrollo y finalización del proyecto que tiene por nombre “ShopIA”, solución tecnológica e innovadora diseñada y desarrollada para la Tienda Aurora, un e-commerce de moda en crecimiento que enfrentaba dificultades (críticas) respecto a la atención al cliente. Ante un caso donde el 35% de las consultas de soportes eran repetitivas y saturaba a todo el equipo humano encargado de esa área en Aurora, el proyecto se tornó a un objetivo principal, que fue la implementación de un chatbot con inteligencia artificial integrada transaccionalmente al sitio web de Aurora.

A lo largo del presente documento se describen las etapas de planificación inicial, diseño y ejecución de los resultados realizados bajo una **metodología ágil**, abarcando desde la arquitectura de software hasta su correcta culminación, dando paso a las pruebas de funcionamiento efectivas esperadas por todo nuestro equipo. La solución construida no solo consiste en el chatbot capaz de responder las 24 horas del día, sino que también incluye el desarrollo de la plataforma de la Tienda Aurora, ya que al momento de empezar a desarrollar el chatbot, que es la “funcionalidad fuerte” del proyecto, Aurora no contaba con un sitio web para desplegar el asistente virtual; para ello, se construyó una página web con una arquitectura desacoplada (Frontend en Node.js y Backend en Flask) y una base de datos relación en PostgreSQL (inicialmente administrada por pgAdmin 4 y posteriormente migrado a Supabase).

En las siguientes secciones se exponen las decisiones técnicas tomadas, como la estrategia para la integración de las “pasarelas” de pago (APIs de prueba de pago online) de Transbank y Mercado Pago y la lógica RAG del chatbot, así como los desafíos enfrentados durante la integración de sistemas heterogéneos para el posterior desarrollo de un puente de enlace entre ellos. Finalmente, se presentan los resultados (correctamente) obtenidos, demostrando cómo la tecnología aplicada en nuestro proyecto logra optimizar los procesos operativos dentro de Aurora y mejorar significativamente la experiencia de compra del usuario final.

# **Contexto del proyecto:**

**Tienda Aurora** es un e-commerce de moda con 4 sucursales físicas (Región Metropolitana y V Región). El tráfico móvil es 72% y el 35% de las consultas a soporte son repetitivas (envíos, cambios, medios de pago, disponibilidad por talla). La empresa busca reducir tiempos de respuesta, aumentar conversión y ofrecer soporte 24/7, integrando un **chatbot de IA** embebido en el sitio, conectado a **stock en tiempo real**, **ubicación/horarios de sucursales** y **catálogo de ofertas/cupones**.

 Es en este escenario que surge **Aurora**, un e-commerce dedicado a moda y belleza que busca modernizar su operación y ofrecer una experiencia más completa a sus clientes. La plataforma está diseñada para que las compras sean rápidas, seguras y confiables, apoyadas por un **chatbot con inteligencia artificial** que ayuda a resolver dudas, encontrar productos, gestionar el carrito y consultar pedidos de manera sencilla.

Por otro lado, Aurora incluye un **panel administrativo** que permite a los encargados del negocio controlar productos, stock y pedidos desde un único lugar, de forma intuitiva y organizada. Para lograr esto, se utilizan tecnologías modernas como **Node.js**, que permiten construir interfaces rápidas, dinámicas y escalables.

Este proyecto no solo apunta a mejorar la experiencia de los usuarios, sino que también permite aplicar conocimientos de ingeniería informática en un entorno real, siguiendo buenas prácticas de desarrollo y seguridad. Al final, se espera obtener un prototipo funcional que refleje tanto el aprendizaje técnico del equipo como su capacidad de trabajo colaborativo.

# **Justificación del proyecto:**

El desarrollo del proyecto se fundamentó en la acelerada digitalización de los negocios y la imperante necesidad del rubro de ofrecer experiencias de compra eficientes, transparentes y confiables a los usuarios. La implementación de un e-commerce completo, que incluye un carrito de compra, integración de medios de pago (sandbox) y el uso de inteligencia artificial, ha permitido entregarles a los usuarios una experiencia completa y satisfactoria, fomentando la fidelización hacia nuestra plataforma. De esta manera, se incorporaron funcionalidades críticas para la administración interna, tales como la visualización y trazabilidad de los pedidos, gestión de productos y control de stock de estos mismo por sucursal, entre otras herramientas de gestión de inventario y de información fundamental para Aurora.

Desde el punto académico, este proyecto constituyó una “oportunidad de oro” para aplicar los conocimientos adquiridos durante la carrera, aplicando e integrando áreas relevantes como lo es el Desarrollo Web, Arquitectura de Software, Seguridad Informática y la Integración de Plataformas heterogéneas. La solución final entregada por parte de nuestro equipo demuestra los aprendizajes consolidados y la capacidad técnica de cada integrante para desarrollar y trabajar en un sistema de software completo, diverso y dinámico cumpliendo con los estándares de escalabilidad, calidad, rendimiento y seguridad, y optimizando las interacciones tanto de los clientes y usuarios finales como para los administradores encargados de Aurora.

El haber tenido que enfrentar un caso real aplicando nuestra formación académica entregada por DUOC UC confirma que el proyecto presentado y trabajado durante todo el semestre fue el idóneo para culminar un producto de software robusto y funcional para las necesidades del cliente.

# **Áreas de desempeño**

Nuestro proyecto abordó varias áreas clave de la carrera. Principalmente, como equipo, nos enfocamos en el **Desarrollo de Software Full-Stack**, ya que tuvimos que levantar tanto el **Frontend** (con Node.js) como el **Backend** (Flask/Python). También conectamos y profundizamos en **Base de Datos**, diseñando y gestionando un modelo relacional en **PostgreSQL** y, por supuesto, nos ocupamos también de la **Inteligencia Artificia**l, implementando la **lógica RAG** y orquestando un chatbot idóneo y capaz de interactuar con el cliente o el usuario final. Finalmente, también tocamos el área de lo que fue la **Arquitectura de Software**, diseñando un sistema desacoplado totalmente, en donde las conexiones entre funcionalidades fueron vía APIs.

# **Competencias**

Como equipo de trabajo, pusimos a prueba toda nuestra capacidad de Resolución de Problemas en el camino, frente a las dificultades impuestas por las distintas tecnologías que usábamos, especialmente al integrar técnicas que no siempre “conversan” bien a la primera (como lo fue el mecanismo de seguridad **“CORS” y sesiones**). También fue fundamental el Trabajo en Equipo bajo metodologías ágiles, donde cada uno de los integrantes del equipo (Matías Hernández como Gestor, Tomás Figueroa como desarrollador y Vicente Morales en QA) cumplió un rol específico para avanzar semana a semana (sprints). Por último, desarrollamos la competencia de Autoaprendizaje, ya que las integraciones de las APIs de pago de prueba **(Transbank y Mercado Pago)** y la configuración de la IA en el chatbot requirieron investigar más allá de lo básico y cambiar métodos para presentar el producto final sin ningún tipo de problema al presentarlo.

# **Relevancia del proyecto APT**

El problema central que buscamos resolver es la saturación que sufre el equipo de soporte de Tienda Aurora. Actualmente, pierden mucho tiempo respondiendo las mismas preguntas una y otra vez (stock por sucursales, tallas de los productos, estado de los pedidos que van realizando), lo que genera algo parecido a un “cuello de botella” que afecta las ventas y la atención del cliente en general. Lo mismo pasa con la trazabilidad de los pedidos que van haciendo los clientes, ya que, como no contaban inicialmente con un sitio web a como lo tienen hoy, los problemas en cuanto a este tema abundaban, y la integración del nuevo chatbot, fue una muy acertada solución a un problema de esta envergadura.

Esta situación es muy relevante para nuestro campo porque demuestra cómo la tecnología no es solo el código que desarrollan los equipos de trabajo, sino una solución directa y concreta a problemas reales de negocio que afectan a los dueños de la tiena, que pierden ventas y a la experiencia del cliente al momento de operar en la página y al no obtener respuestas rápidas y concretas. El proyecto se sitúa en el contexto del **“retail digital chileno”**, donde la inmediatez de las cosas lo es todo.

El valor real de “ShopIA” es que democratiza el acceso a la atención 24/7, o sea, cambiar ya directamente a cómo se estaba trabajando internamente antes del desarrollo de este proyecto. No solo, como equipo de trabajo, implementamos un e-commerce funcional y completo, sino que le otorgamos un “cerebro” a Aurora que puede revisar el stock en tiempo real y resolver dudas sin la necesidad de intervención humana inmediata, liberando al personal para tareas más complejas.

# **Objetivos**

**Objetivo General**: Diseñar, desarrollar e implementar un chatbot inteligente integrado al e-commerce de la **Tienda Aurora**, capaz de automatizar la atención al cliente mediante inteligencia artificial. Buscamos proporcionar respuestas rápidas emitidas por el bot sobre los envíos, stock y pedidos (respuestas personalizadas dependiendo las peticiones hechas por el usuario hacia el chatbot) con el fin de reducir la carga operativa y mejorar la experiencia del usuario final.

# **Objetivo General Alcanzado:**

Se desarrolló un sistema de comercialización electrónico (e-commerce) escalable, seguro y personalizado, que le da total libertad a los clientes y usuarios finales de navegar por el sitio web de Tienda Aurora, buscar productos en específico, gestionar su carrito de compra permitiéndole al cliente elegir el método de cómo recibir su producto (Retiro en una sucursal de Aurora o Despacho a domicilio utilizando la API de Coberturas de Regiones de Chilexpress) y realizar pagos mediante Transbank o Mercado Pago en un ambiente de pruebas seguro. El sistema logra e integra eficazmente la consulta de pedidos por parte del usuario final y cuenta con un chatbot con inteligencia artificial integrada que resuelve dudas personalizadas y mejora la atención del cliente en todo aspecto. Además, se entregó con creces un panel administrativo funcional al 100% (funcionando mediante una aplicación CRUD) para la administración de pedidos y pagos,

# **Objetivos específicos cumplidos:**

Dentro de los objetivos específicos para alcanzar la meta de nuestro proyecto están:

* **Desarrollar una base de datos** robusta con **PostgreSQL** para gestionar todo el inventario disponible por producto en sucursales en específico, tallas disponibles por producto creado, usuarios y roles específicos de cada cuenta creada, detalles de las sucursales, códigos de descuento y registro de los pagos y pedidos realizados para su posterior gestión.
* **Implementación de catálogo:** Se creó un catálogo de productos con funcionalidades completas de búsqueda, filtrado, creación, modificación, visualización y eliminación
* **Sistema de usuarios:** Se desarrolló un módulo de autenticación seguro con registro vía Gmail, inicio de sesión y recuperación de contraseña vía Gmail
* **Checkout y pagos:** Se construyó el flujo de carrito de compras con cálculo automático de totales, impuestos (IVA) y envíos dependiendo la dirección del despacho, integrando pagos electrónicos con Transbank y Mercado Pago, incluyendo manejo de webhooks y comprobantes de compra.
* **Chatbot (con IA integrada):** Se implementó un chatbot inteligente capaz de responder preguntas frecuentes, dudas personalizadas dependiendo el caso, consultar el estado de pedidos de usuarios autenticados e informar sobre stock y ubicación de sucursales en tiempo real con información precisa e, incluso, conectándose con Google para mayor detalle de la ubicación de los locales disponibles en Aurora.
* **Panel de administración**: Se diseño una interfaz de gestión para los administrador para gestionar productos, sucursales, pedidos, cupones de descuentos y ofertas, con capacidad de generar reportes relevantes para el negocio.

# 

# **Visión General del Proyecto finalizado:**

A continuación, se describen los cuatro aspectos principales que definieron la solución entregada por nuestro equipo de trabajo:

**1.- Alcance:**

El proyecto culminó con el desarrollo de una plataforma centralizada que facilita la interacción entre la empresa (Tienda Aurora) y sus clientes. El sistema logrado y entregado actúa como un puente entre estos dos, permitiendo una administración y flujo fluído de recursos clave para el correcto funcionamiento de Aurora (inventario, sucursales, promociones) y optimizando al máximo la experiencia del cliente final mediante el acceso a información actualizada en tiempo real. Además, se implementó exitosamente un espacio de soporte y resolución de problemas automatizado, logrando así, una relación dinámica, cercana y ágil entre el usuario y el sitio web. En resumen, el sistema integral entregado centraliza lo que es la gestión de los recursos disponibles y potencia la relación comercial de Aurora.

# **2.- Requerimientos abordados:**

**Requerimientos del negocio:**

* **Gestión de productos:** La plataforma permite registrar, editar y actualizar productos nuevos y ya creados con todos sus atributos (SKU, nombre, descripción, precio, stock por tallas, categoría, imágenes).
* **Administración de sucursales**: Se administra la información detallada de sucursales disponibles en el momento, incluyendo ubicación exacta, horarios de operación y contacto (vía email y número telefónico)
* **Control de inventario:** El sistema refleja con exactitud el stock disponible a tiempo real en cada sucursal y bodega, variando en función en cuál sucursal el cliente esté navegando y a tiempo real.
* **Manejo de pedidos y cupones**: Se registran los pedidos de los clientes y se aplican descuentos a través de cupones de descuentos (también gestionados por el administrador y todo aquel que tenga acceso a los paneles de administrador) según las reglas de negocio definidas al momento de la creación de los códigos de descuento.
* **Promociones:** Existe un módulo para administrar las ofertas asociadas a productos en específicos.
* **Soportes y Chatbot:** El asistente virtual implementado en Aurora resuelve dudas frecuentes y personalizadas por parte del usuario y guía a los clientes en sus procesos de compra, operando como un primer nivel de soporte.
* **Roles:** Se garantiza la seguridad mediante niveles de accesos restringidos y diferenciados (Admin, Soporte, Cliente).

# **3.- A quién beneficia la solución:**

# **El proyecto finalizado beneficia directamente a dos grupos involucrados:**

* **Clientes finales:** Después de finalizado y entregado el proyecto, disponen de una experiencia de navegación y de compra eficiente, con visibilidad de stock en tiempo real, ubicación de tiendas e información detallada y completa de estas, trazabilidad de sus compras y pedidos y resolución inmediata de dudas gracias al chatbot con IA integrada, lo que les otorga a los clientes finales total autonomía y confianza.
* **Tienda Aurora:** Después de meses de desarrollo y trabajo, Aurora cuenta a día de hoy con una herramienta potente y robusta que centraliza la gestión de inventario y ventas. La automatización de la atención al cliente y el manejo digital de pedidos han reducido la carga operativa del equipo humano detrás de la tienda, mejorando la competitividad y modernizando la imagen de la marca.

# **Metodología**

Para sacar este proyecto adelante, decidimos usar una metodología ágil basada en **SCRUM**. Decidimos no trabajar a ciegas hasta el final de cada entregable así que, como equipo de trabajo, nos organizamos para que el trabajo de cada uno de los integrantes avance mediante sprints semanales, poniendo como **meta cada miércoles con las funcionalidades que hablamos cada jueves.**

Esta forma de trabajo fue la que el equipo creyó pertinente, ya que permitió dividir lo que es un proyecto grande en tareas pequeñas y manejables. Si algún integrante del equipo se atrasaba, el resto lo ayudaba a resolver o le aconsejaba seguir adelante con otras tareas para evitar una congestión en el cronograma: primero se diseñó, luego se construyó el sitio base, se siguió con la integración de las **APIs** y, finalmente, sumamos el chatbot, que es la funcionalidad principal del proyecto. Como equipo, esta manera en la que decidimos trabajar nos sirvió de mucho para validar los avances individuales y grupales de cada semana y, así, corregir el rumbo rápido cuando algo técnico no lograba funcionar, asegurando llegar con un **MVP** funcional a la entrega final.

# **Desarrollo**

## **Etapas y actividades:**

**Dividimos el proyecto en tres grandes etapas:**

La **Fase 1** fue solo planificación y diseño de la arquitectura de lo que sería nuestro producto final y la base de datos (que luego sería modificada, al menos en cómo administramos los datos). En esta fase, luego de la planificar y diseñar, se hizo una pequeña base de lo que sería el bosquejo o plantilla de nuestro sitio web, para que tome forma y pensáramos de una manera más clara. En la parte visual de la página trabajamos, en conjunto de **HTML** por supuesto, **CSS** y **Bootstrap**, lo que nos permite crear formularios y tablas para mostrar productos y sucursales. Incluimos también detalles de diseño, como marcos y fondos diferenciados para que cada sección sea más clara y fácil de identificar.

En la **Fase 2**, que fue la más intensa sin dudas, nos decidimos pasar directamente a la programación y fuimos de lleno: el Backend en Flask, el Frontend en Node.js y ya conectamos la base de datos en PostgreSQL, ya que en la fase 1 solo trabajamos de manera local y fue aquí, en la conexión de una base de datos robusta, en que nos afirmamos para avanzar progresivamente en nuestro proyecto. Terminando esta fase nos decidimos a trabajar con un gestor de base de datos diferente (estuvimos la gran parte del proyecto trabajando en pgAdmin 4): **Supabase**. Esta plataforma de desarrollo de Postgres fue un alivio en cuanto a cómo íbamos haciendo las pruebas, ya que, al estar trabajando todos en distintas máquinas físicas al principio, mis datos en las tablas eran diferentes a los de mis compañeros y Supabase actualizaba todo en tiempo real, lo que flexibilizó muchos aspectos del desarrollo.

Finalmente, en la **Fase 3**, nos enfocamos en agregar detalles e ideas que se nos iban ocurriendo en el camino a medida que íbamos diseñando funcionalidades base de nuestra página y no quedarnos solo con un producto base, común y corriente, e ir más allá de lo que decía el contexto inicial del caso, aportándole más robustez a nuestro producto final. En esta fase, por supuesto, integramos la funcionalidad del chatbot con inteligencia artificial integrada (que también probamos varios métodos hasta que al final de esta fase nos decantamos por una opción mejor) y terminando con las pruebas de optimización y marcha blanca para asegurar que todo funcionara fluido al momento de tener que presentar el producto final.

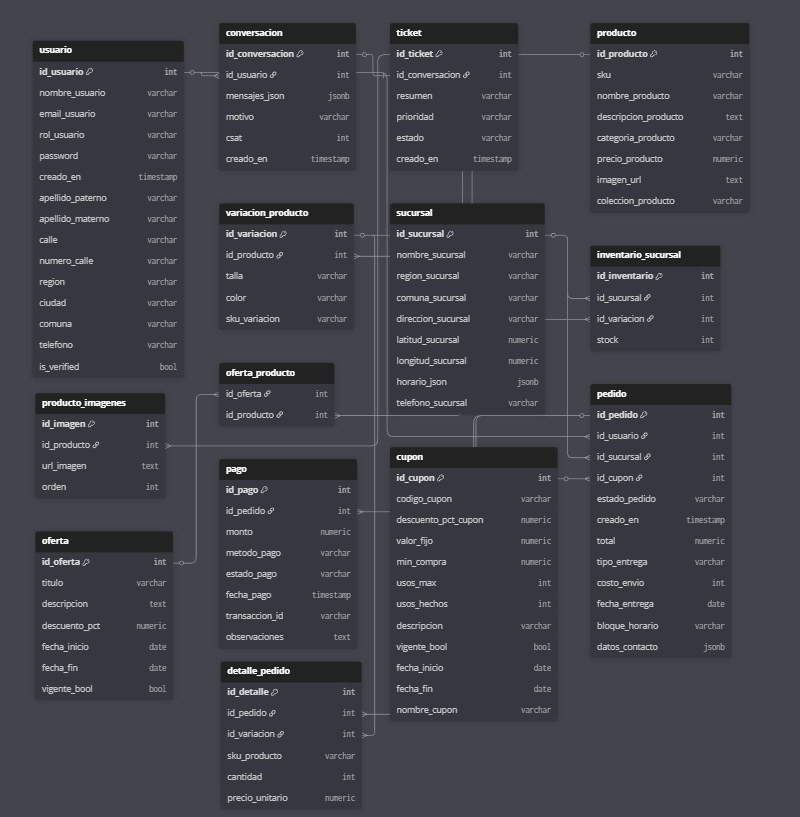
## **Dificultades y facilitadores:**

Lo que más nos ayudó fue tener los roles claros dentro del equipo y el uso de tecnologías dentro del proyecto que, a pesar de su complejidad, tienen una documentación sólida e información nunca faltó, como **Express y PostgreSQL**. Sin embargo, enfrentamos dificultades técnicas serias, que nos atrasó en cuanto a los sprints semanales. Lo más complicado fue el manejo de **CORS** y **sesiones**: al trabajar con una arquitectura desacoplada y tener el Frontend con el Backend en puertos distintos, las cookies de sesión se perdían al apretar un botón o al cambiar de página en el sitio web, lo que nos costó mucho lograr que el usuario se mantuviera logeado como correspondía y los datos se guardaran dentro de la sesión actual. También sufrimos con los puertos correspondientes a las **APIs de pago de Transbank y Mercado Pago**, sobre todo en esta última; Mercado Pago, por ejemplo, devolvía muchos *null* al momento de pagar o tenía falsos positivos al momento de concretar la compra, ya que, a pesar de entregar un mensaje de éxito en la compra de prueba, en la base de datos no se guardaba como correspondía lo que nos demoró mucho más tiempo del esperado.

Ajustes realizados: Para superar estos baches en el camino, tuvimos que ajustar nuestra planificación y Roadmap. Decidimos priorizar la estabilidad de lo principal en nuestra aplicación (el e-commerce no podía presentar fallas de ningún tipo) antes de integrar toda la complejidad de lo que significa un chatbot con respuestas personalizadas y en tiempo real. Como equipo, tuvimos que re-estudiar cómo flask maneja las sesiones “cross-origin” para solucionar el problema de autenticación. Básicamente, sprint a sprint, fuimos haciendo la aplicación más robusta para no encontrarnos con sorpresas al final.

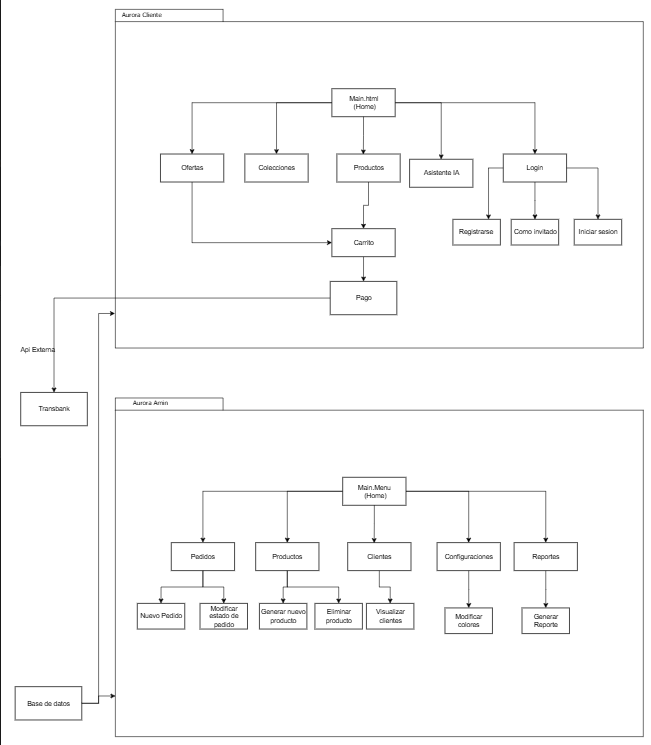
# **Evidencias:**

## **1.- Modelo de datos**



## **2.- Arquitectura**

Arquitectura cliente - servidor desacoplado con comunicación mediante APIs REST y soporte CORS.



# **3.- Tecnologías**

**Node js:**

* Framework principal del servidor backend.
* Maneja rutas de pago (WebPay, MercadoPago), carrito, pedidos, y conexión con el frontend.

**Python flask:**

* Framework principal del servidor backend.
* Maneja rutas de pago (**WebPay, MercadoPago**), carrito, pedidos, y conexión con el frontend.

**PostgreSQL + Supabase:**

* Base de datos PostgreSQL
* creación de tablas como:  
   usuario, producto, inventario\_sucursal,  
   pedido, etc.
* Gestionado desde SupaBase.

**HTML, CSS, JS**:

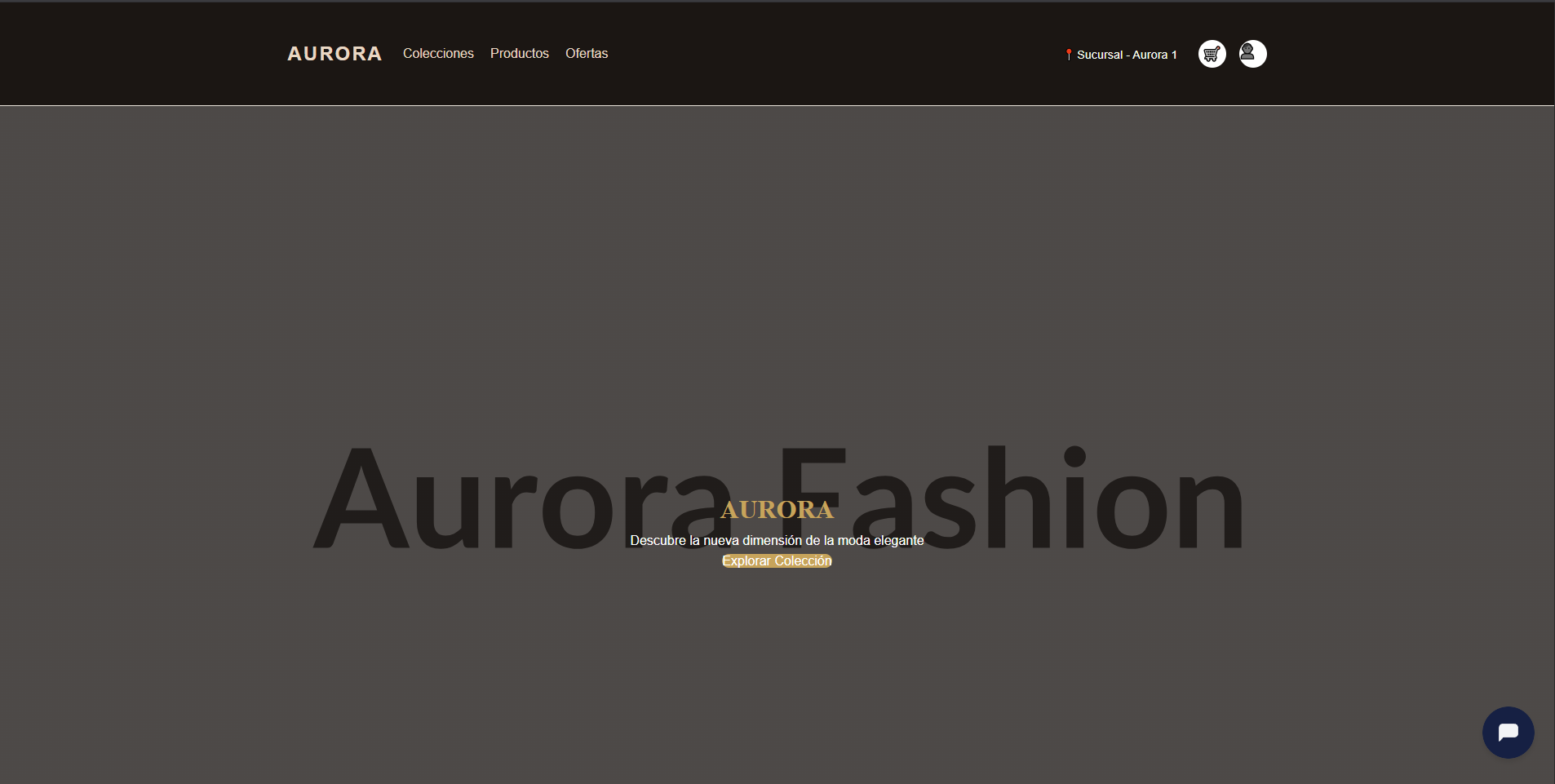
* Interfaz completa del sitio Aurora (main, colecciones, productos, carrito, boleta, etc.).
* Interacción dinámica con el carrito, gestión de pagos, chat, etc.

**Botpress:**

* Se usó para la creación del chabot
* Se entrenó para ayudar al equipo de soporte
* Automatizar respuestas

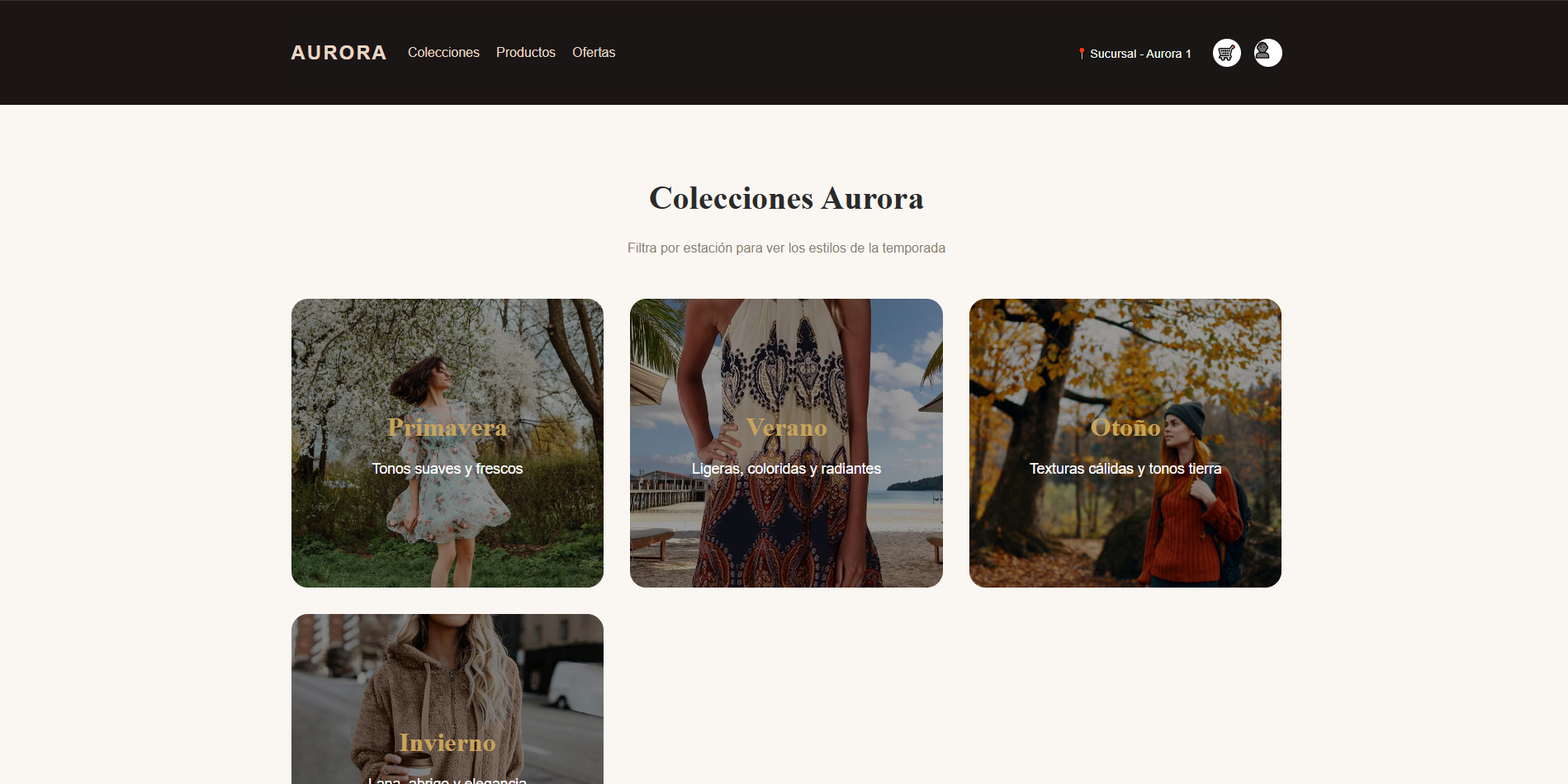
# **4.- Resultado del producto final (imágenes)**

## **Home**

Esta página corresponde a lo primero que ve el cliente al ingresar a la Tienda Aurora. Esta página muestra la sucursal a la que se le asignó por geolocalización al cliente, símbolo para loguearse/registrarse, botón del carrito de compras, logo de la página y un Navbar con los botones “Colecciones”, “Productos” y “Ofertas”.

## **Colecciones**

Catálogo de colecciones de los productos. Cada producto, al momento de crearse, se le tiene que asignar una categoría de “Colecciones” para que pueda aparecer en esta sección. Las colecciones son: **Primavera, Verano, Otoño e Invierno.**



## **Productos**

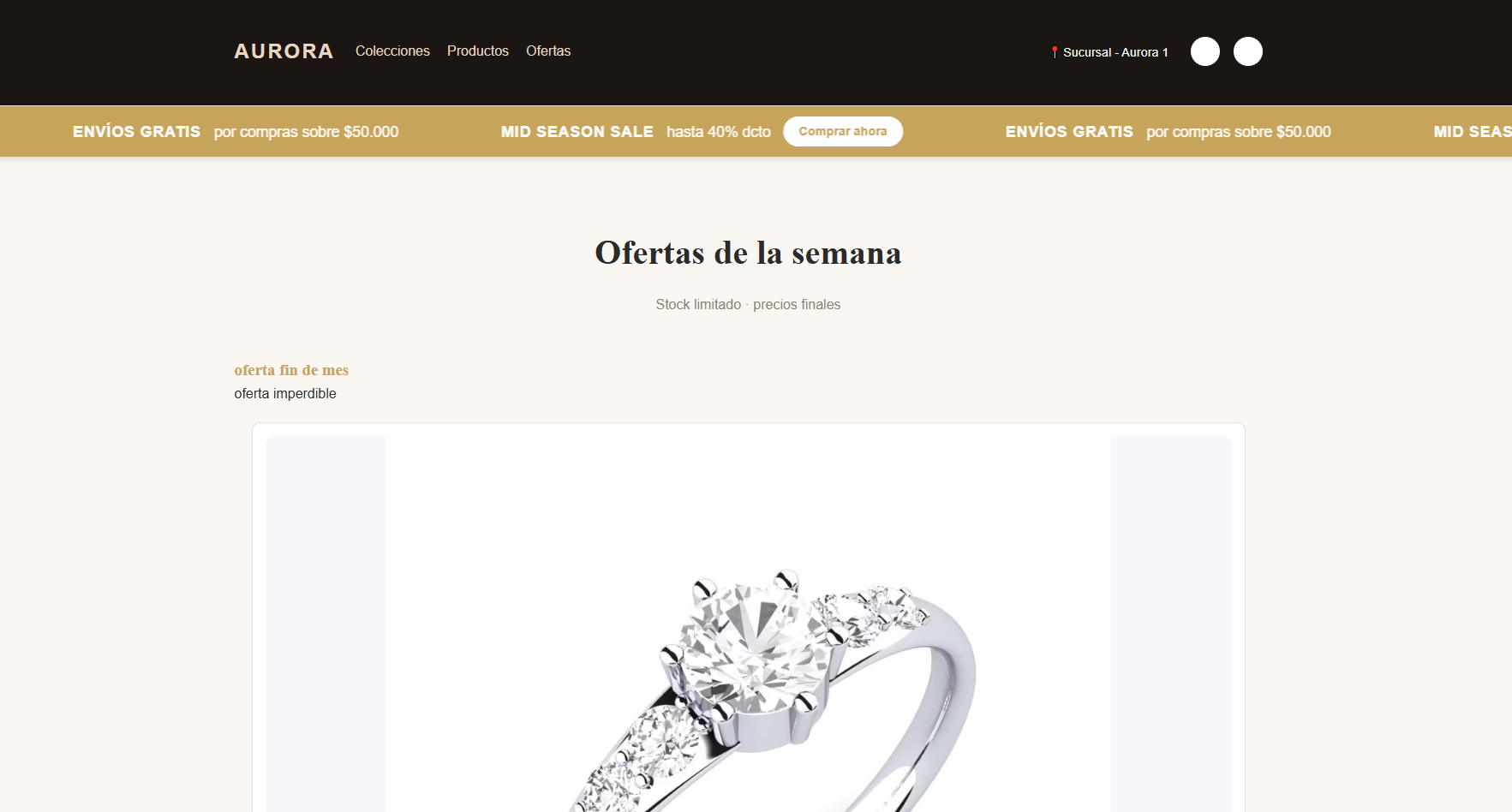
Catálogo de productos que lista todos los productos disponibles en Aurora, con los nombres de los productos, precio, categoría a la que pertenece y precio en oferta, en caso de que el producto posea activamente una oferta. Al hacer click en cualquier imagen, la página redirige a una página con información específica del producto escogido.



## 

## **Ofertas**

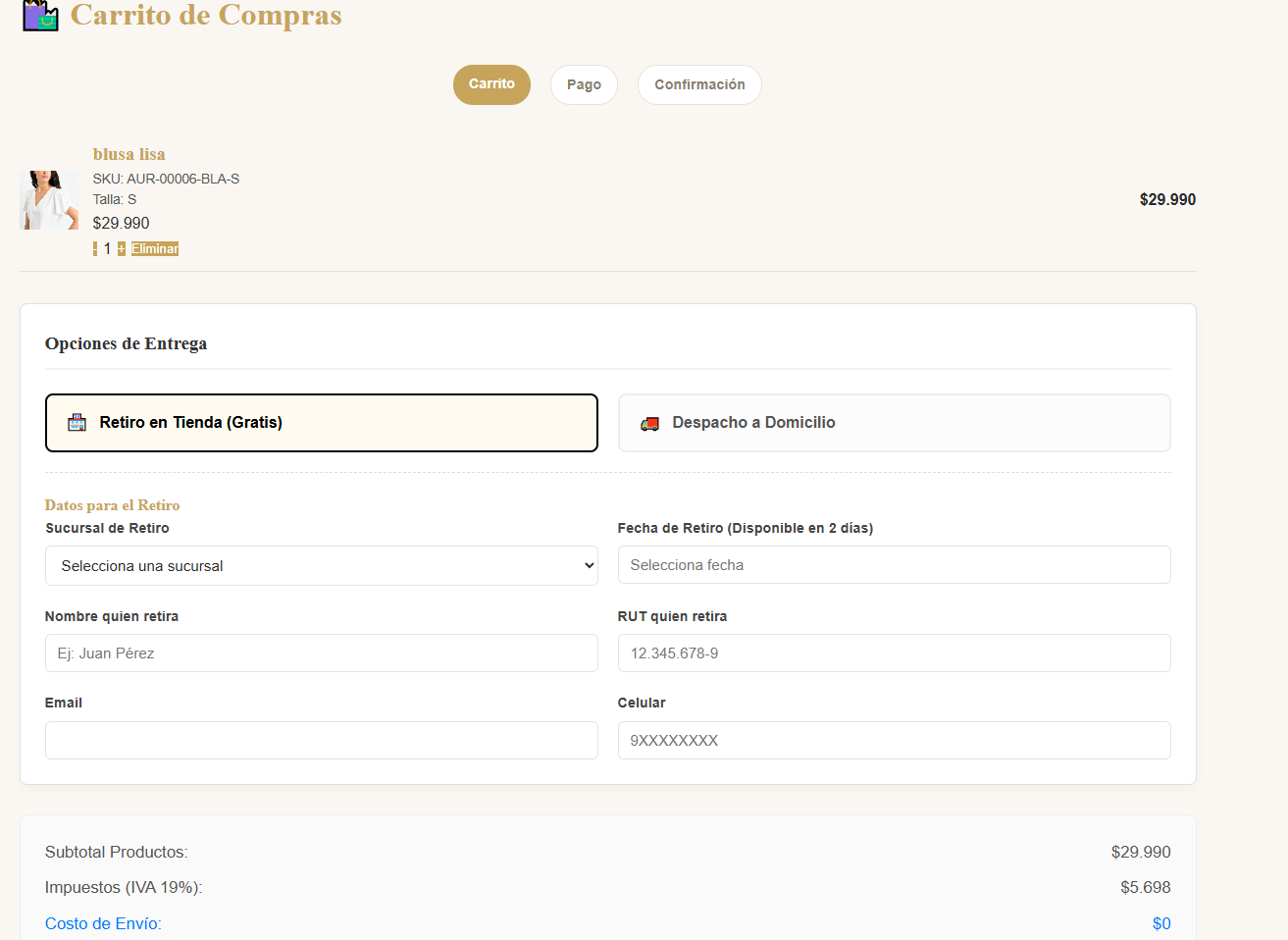
Ofertas de los productos con su respectivo descuento. Al hacer click en la imagen, redirige a una página con todos los detalles del producto en oferta. El precio en descuento se respeta en todo momento, hasta que la fecha haya caducado.



## **Carrito de compra**

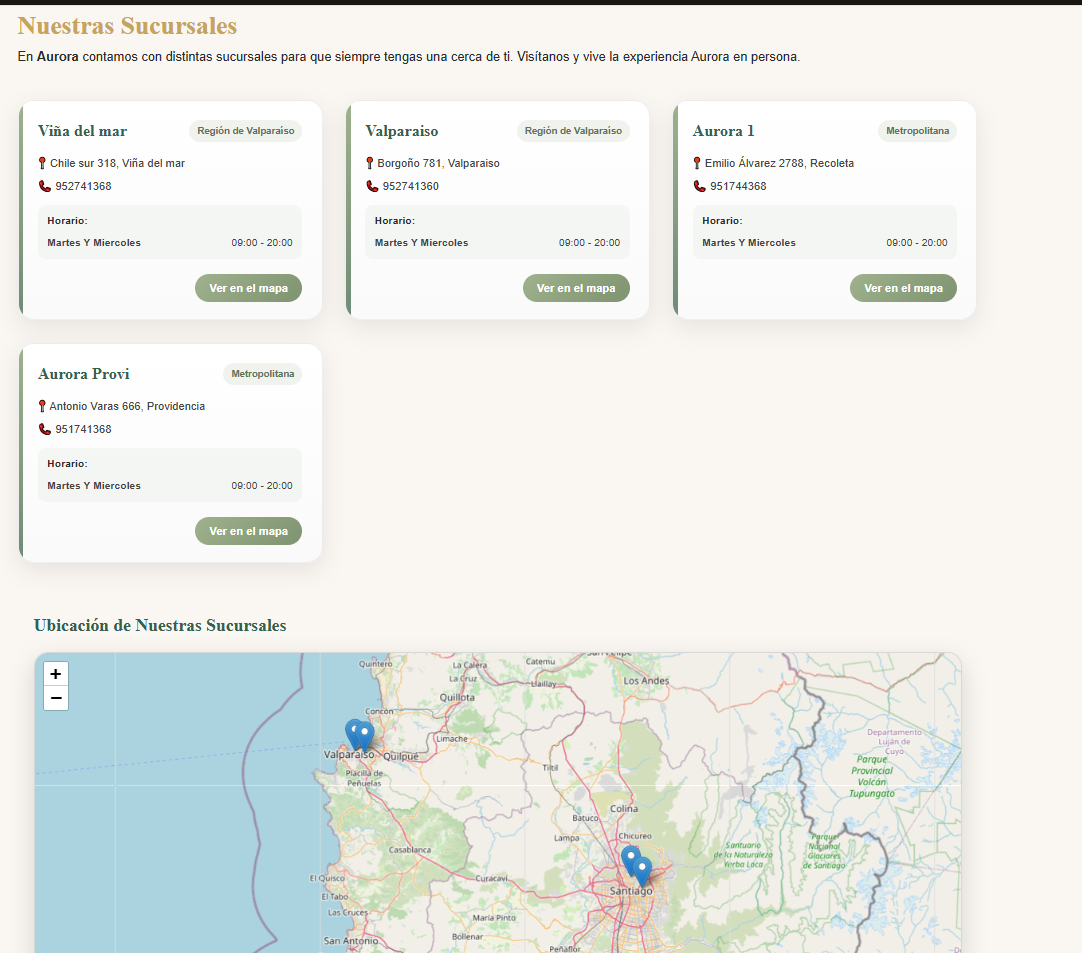
Con la información del producto que se comprará, más las opciones de entrega que pueden ser retiro en tienda o despacho a domicilio (**API Chilexpress**), además de la suma del iva (19%) al precio total. Respecto al **Despacho a Domicilio**, el costo de envío varía según la región (para la Región Metropolitana hace un recargo de $5.000, fuera de esta, es de $10.000). Se rellena un formulario eligiendo la hora aproximada, a gusto del cliente, para recibir su pedido en casa.

Respecto a **Retiro en Tienda (gratis)** se tiene que seleccionar sucursal y completar un formulario para su posterior retiro.



## **“Nuestras Sucursales”**

En el apartado del Footer se puede ver información de la tienda, donde incluye las sucursales en las que se muestran su ubicación agregado la **API** de Maps. Esta sección muestra información relevante sobre cada sucursal existente de Aurora a lo largo del país.



## **Login y Registro**

Panel de Login para ingresar a la cuenta. Cuenta con un botón para Registrarse en caso de no estar registrado en la plataforma. El login cuenta con validaciones del usuario, en caso de escribir mal el correo o la contraseña no coincida con la dirección de correo electrónico. Cuenta, también, con un botón de recuperación de la contraseña, que redirige a una sección para escribir el correo electrónico de la cuenta a la cual quiere efectuar la recuperación de la clave, con enlace directamente a la cuenta de gmail real del usuario, en el cual se le enviará un correo con una redirección a un panel de recuperación para que efectúe el cambio. Luego de eso, será redirigido al mismo apartado de “Iniciar Sesión” para que ingrese con sus credenciales actualizadas.

El Formulario de Registro contiene información relevante que se necesita recopilar del usuario. Contiene validaciones por parte del número telefónico, correo electrónico y contraseña. Antes de terminar el registro de la cuenta nueva, el formulario hace una búsqueda en la base de datos para verificar que los datos ingresados (Correo electrónico y celular) no existan previamente y, así, permitirle al usuario crear su cuenta sin problemas. La contraseña contiene requisitos específicos para su creación que, si no se cumplen, será imposible avanzar.



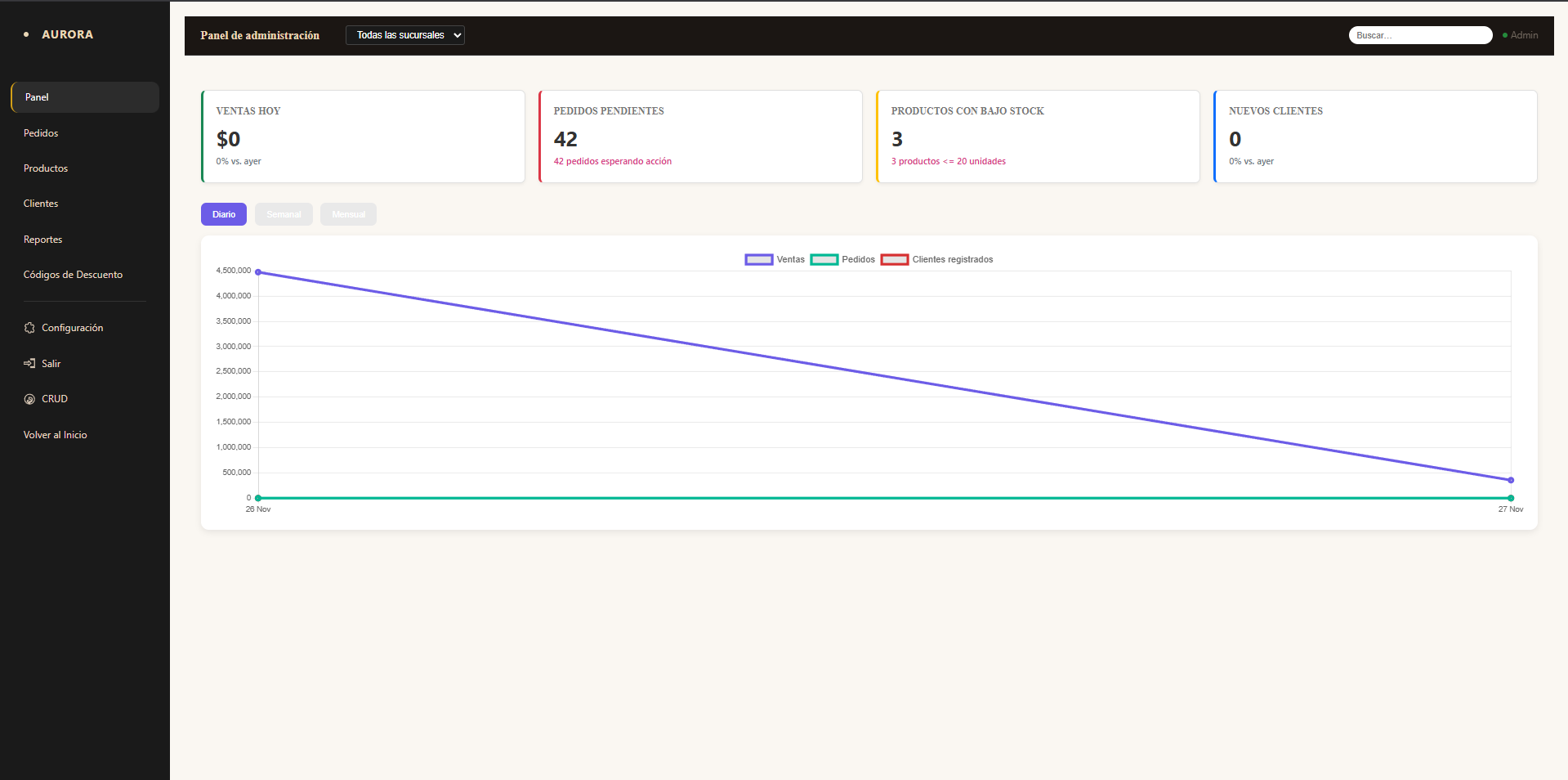
## 

## **Panel de Administrador**

Sección principal. Es donde el administrador tiene las opciones de ver los pedidos hechos por los clientes, ver los pedidos, productos, los nuevos clientes y los códigos de descuento.

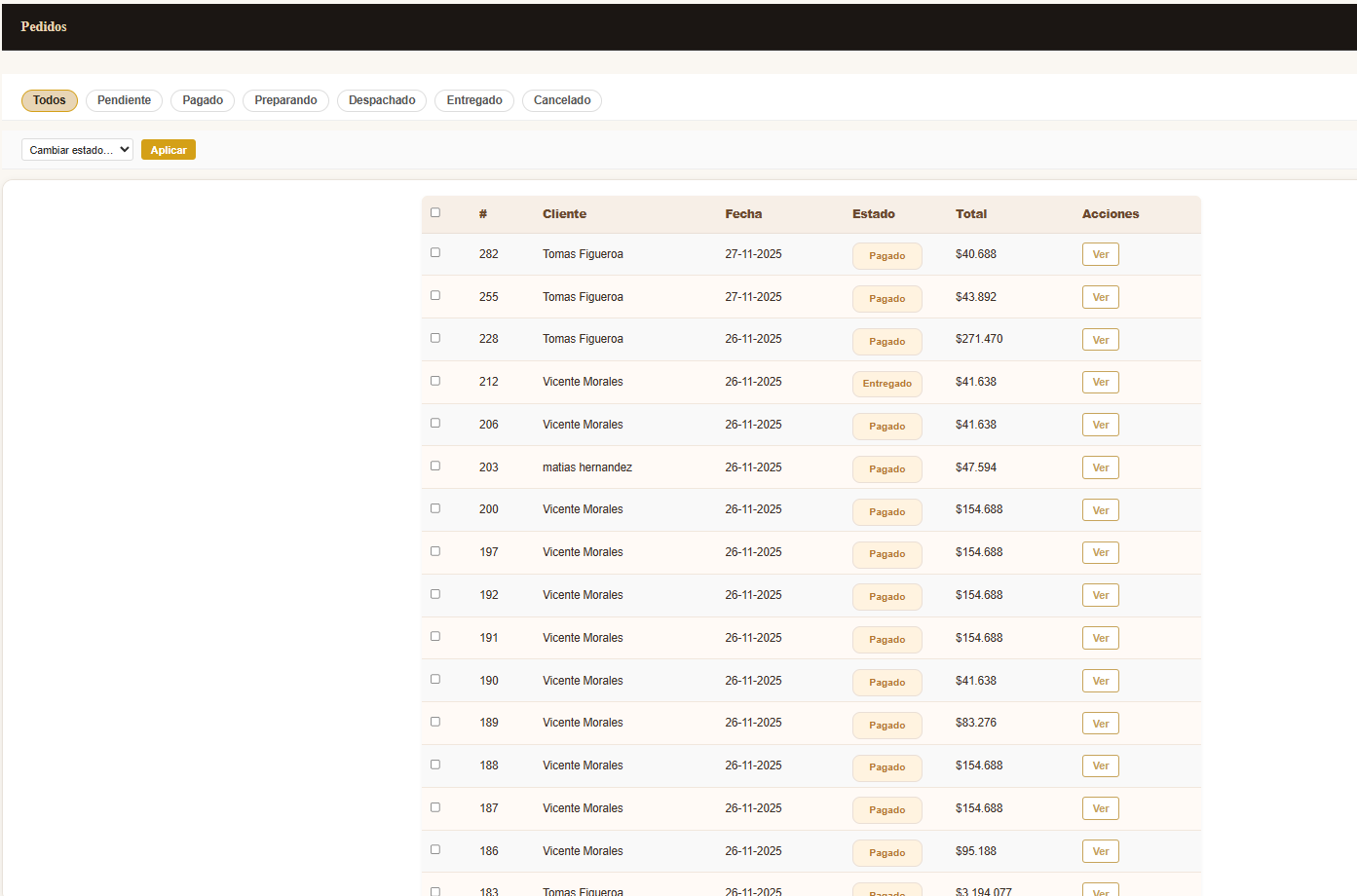
En el apartado de Panel se muestra un gráfico de las ventas diarias en donde se puede ver por día, semana y mes.

El panel cuenta con un filtrado por sucursal, una “card” de las Ventas de Hoy, otra de los Pedidos Pendientes, Productos con bajo stock (que muestra una lista de todo aquel producto que actualmente cuente con menos de 20 unidades de stock) y una card de los nuevos clientes



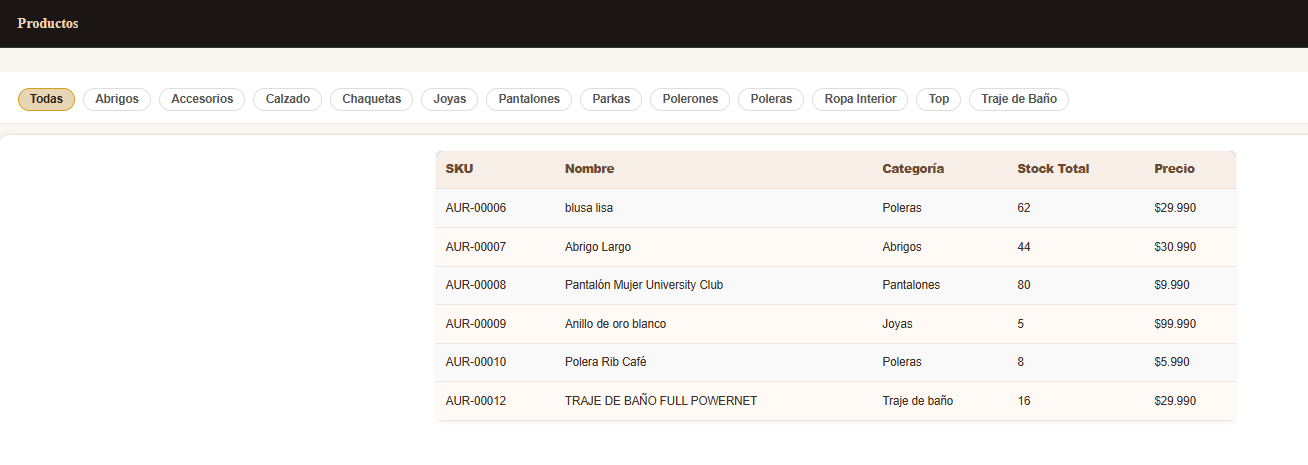
## **Pedidos (Panel de Administrador)**

Sección que muestra de todos los productos comprados por los clientes (y el detalle) y el estado de este; muestra si está **pendiente**, **pagado**, **en preparación**, **despachado**, **entregado o cancelado.** Cuenta con un filtrado amigable en la parte superior del listado de los pedidos. El listado cuenta con un botón para ver el detalle de cada pedido realizado, que cuenta con información de pago, quién realizó el pedido, cuánto se canceló, qué compró y cuánto compró, qué tallas compró el cliente, etc.



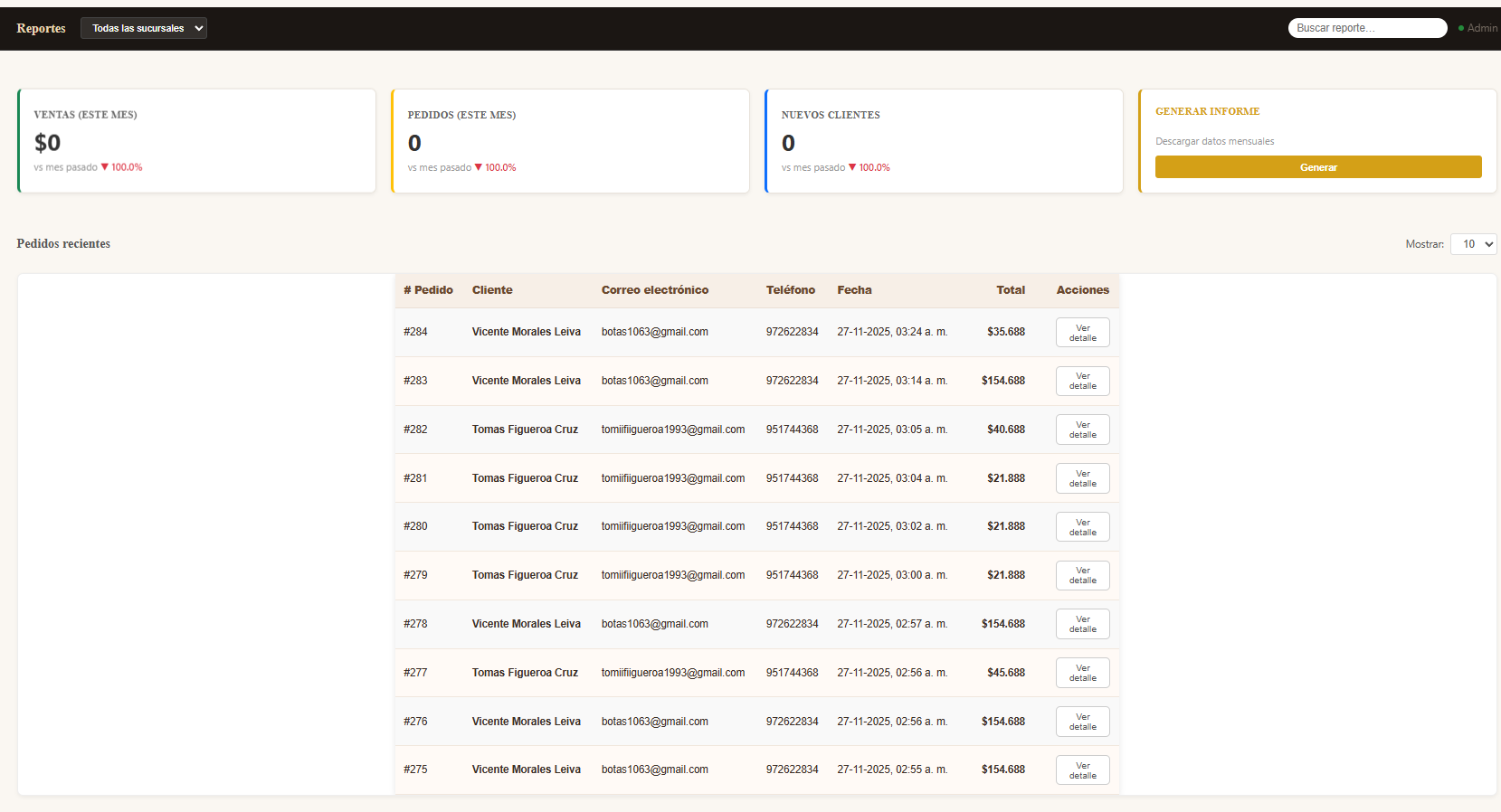
## **Productos (Panel de Administrador)**

Se visualizan todos los productos disponibles en la página con su respectivo SKU, Nombre del producto, Categoría, Stock total y Precio. Cuenta con un filtrado amigable en la parte superior del listado de productos.



## **Reportes (Panel de Administrador)**

Apartado de reportes, donde se pueden apreciar las ventas con un sistema de filtrado por día, semana y mes y cuadros comparativos dentro de las “Card”. Tiene una opción de generar informes en formato **.csv** sobre ventas, pedidos y nuevos clientes por el presente mes o el mes pasado.

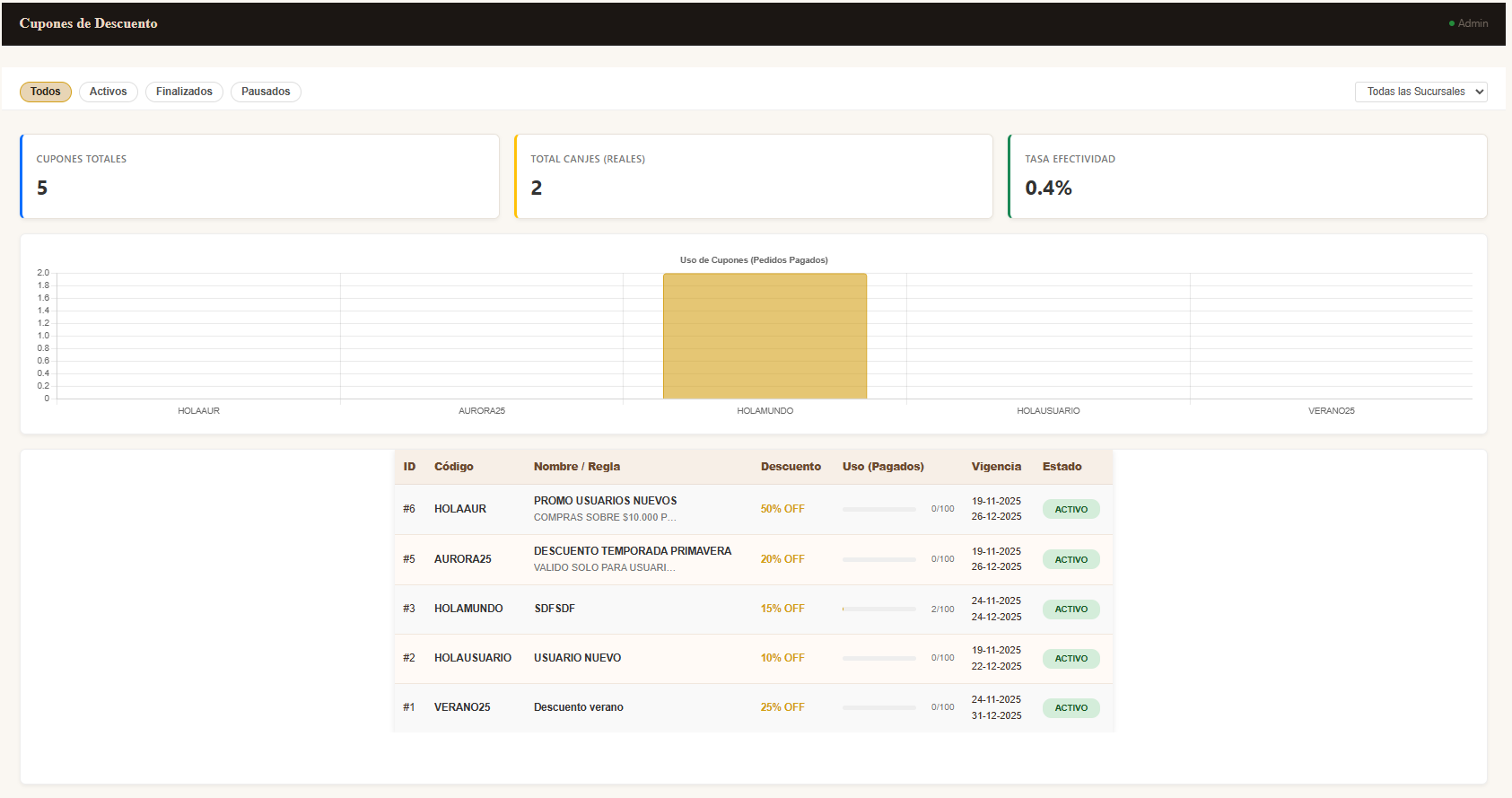


## 

## 

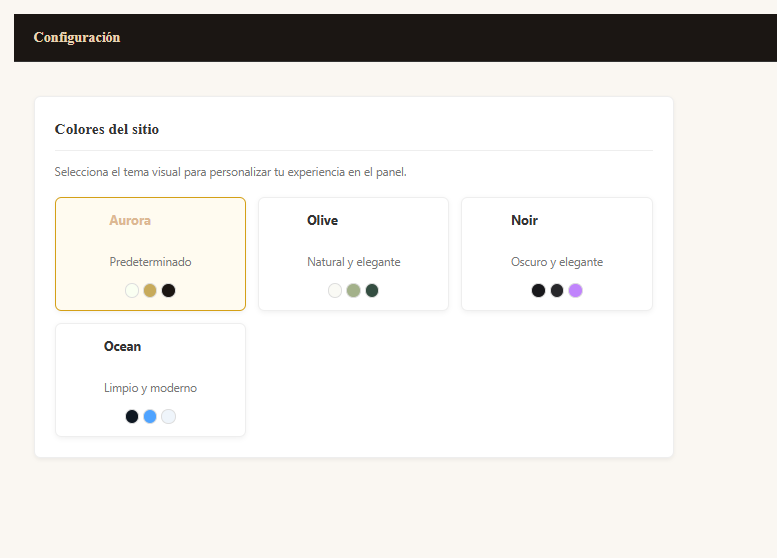
## **Códigos de descuento (Panel de Administrador)**

Se muestran los cupones de descuento que están activos y los que más se han utilizado, con parámetros y gráficos comparativos que evidencian el porcentaje de uso, tasa de efectividad, el total de cupones creados con su detalle, etc.

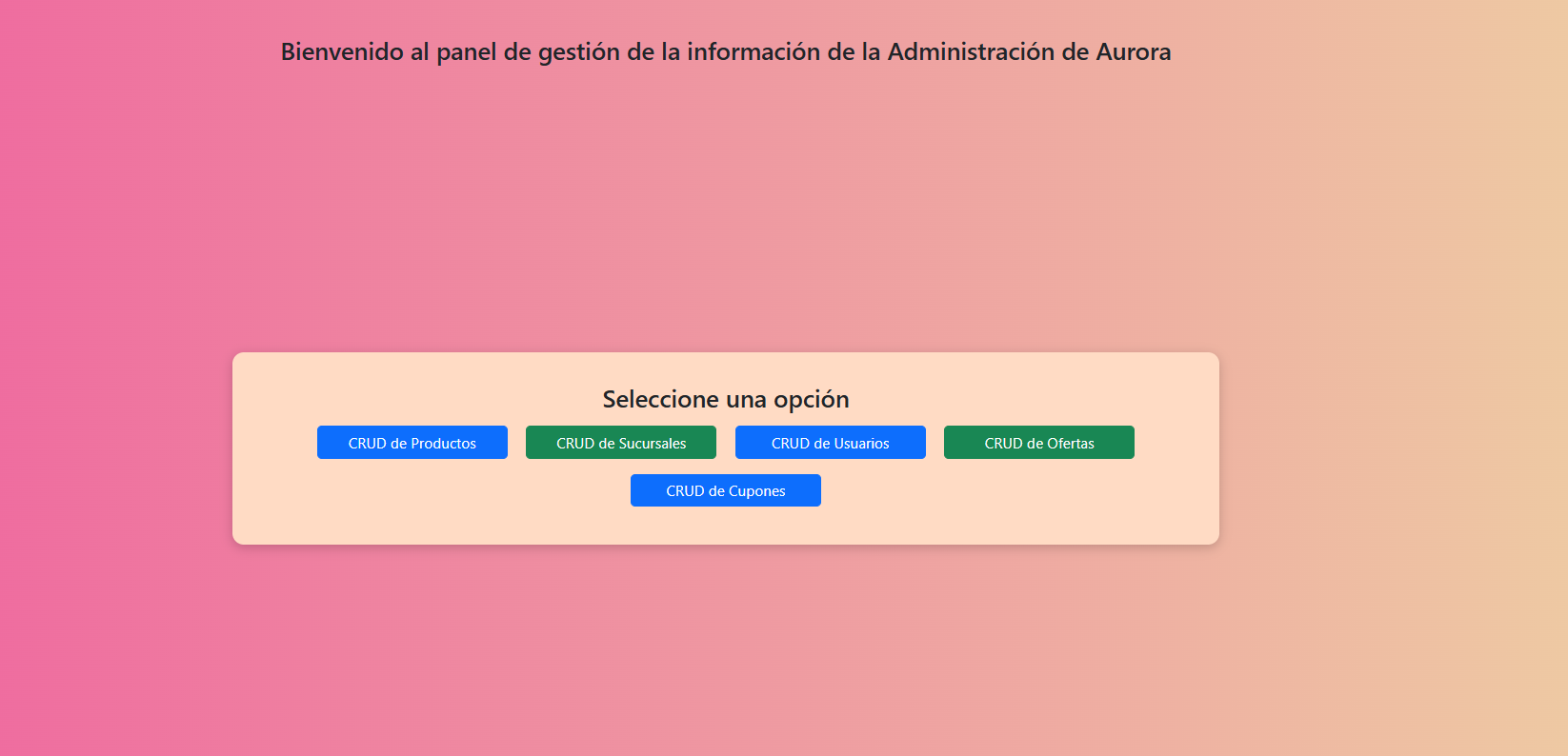


## **Configuración**

Panel que solo tiene como fin cambiarle los colores a la página. La baraja de opciones es: Color Predeterminado, Olive, Noir y Ocean.

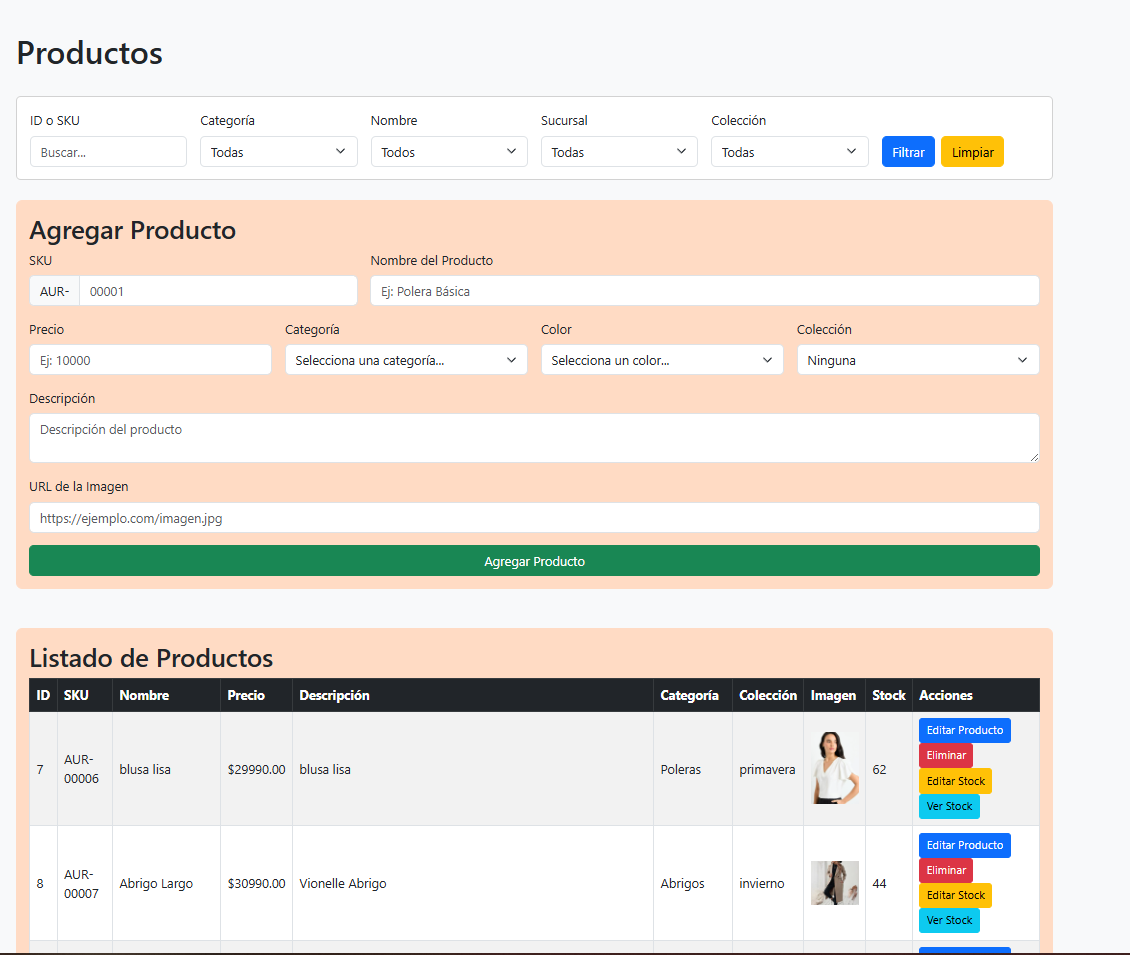


## **Vista principal del panel de gestión de la información de la administración de Aurora**

Página principal donde se alojan los “**CRUD**”. Se visualizan cinco opciones para gestionar: **Productos**, **Sucursales**, **Usuarios**, **Ofertas** y **Cupones**. Se puede navegar libremente dentro de las opciones ofrecidas. Dentro de cada sección también se puede navegar entre paneles, haciendo la misma función que en este apartado.

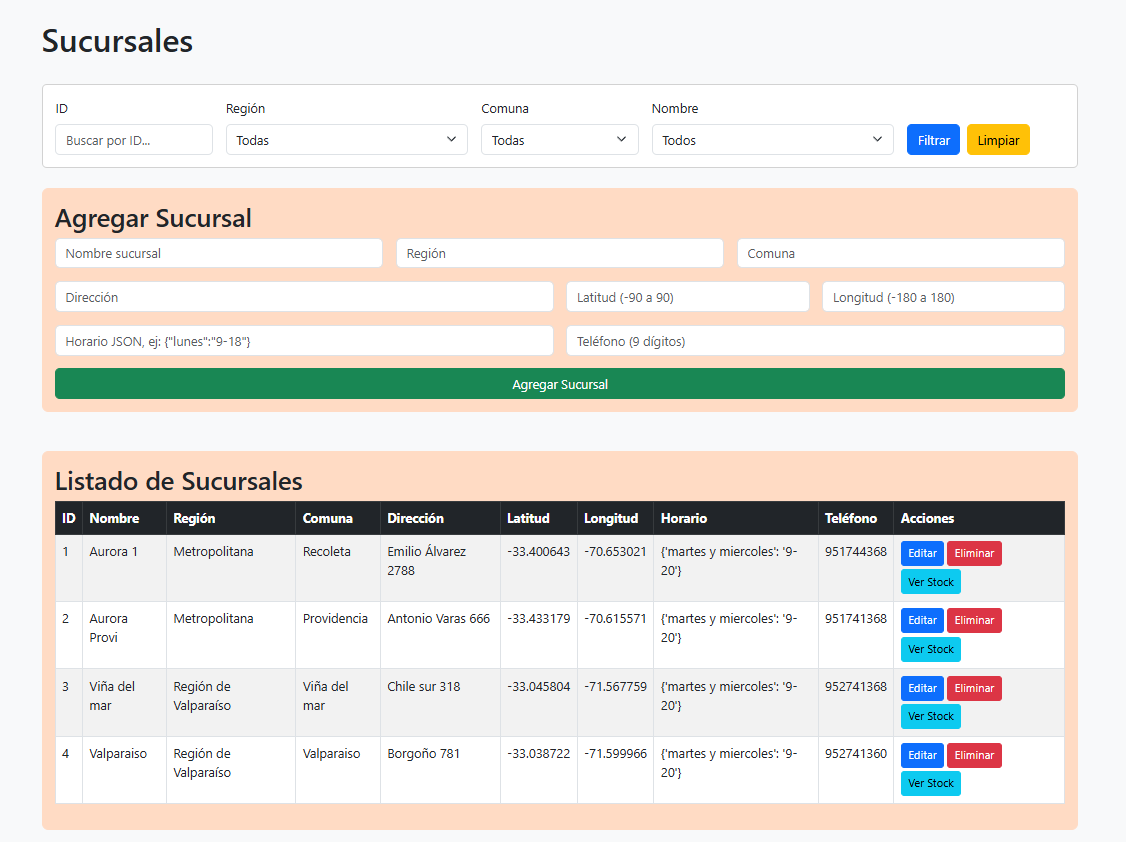
## **CRUD de Productos**

Sección para **agregar los productos con su SKU, precio, categoría, color, colección, descripción e imágenes del producto en cuestión**. Contiene un Listado de Productos, el cual muestra todos los productos agregados con su respectivo ID, SKU, Nombre, Descripción, Categoría, Imagen y Stock. En esta lista se puede ver el stock de cada producto por sucursal y se puede gestionar este mismo ítem. Cuando un producto es adquirido por un cliente con una cantidad ‘x’ de unidades para el producto en cuestión, en el listado, se va restando dependiendo el stock general y stock por tienda en específico.



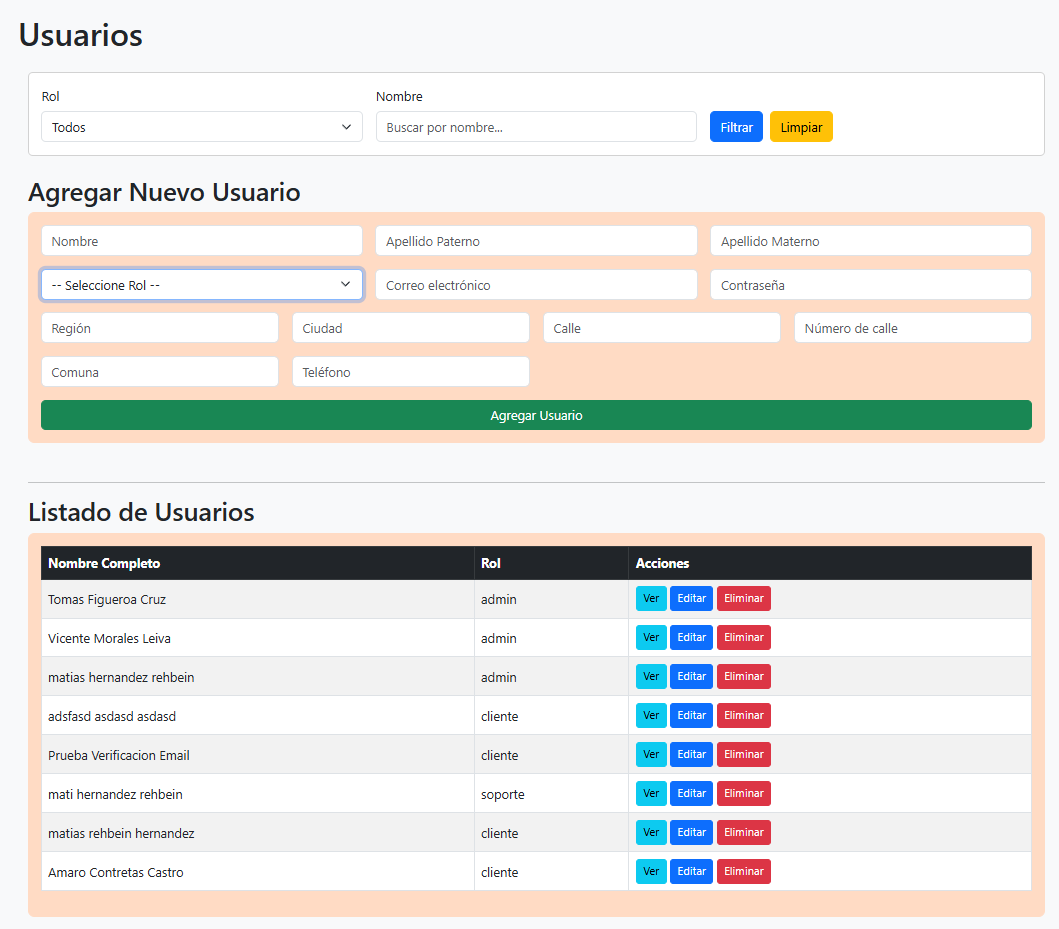
**CRUD de Sucursales**

Sección donde se pueden **agregar, editar o eliminar sucursales, el cual contiene toda su información (nombre de la sucursal, región, comuna, dirección, horario de la sucursal, latitud, longitud, teléfono)**. Contiene un listado de Sucursales que recoge la información de las sucursales creadas a partir del formulario. También cuenta con un botón para ver el stock por sucursales de los productos. Cuenta con un filtro.



**CRUD de Usuarios**

Una de las secciones principales. En este apartado se crean usuarios y se les asignan los roles a las personas desde una vista del administrador. **Al usuario creado se le puede otorgar el rol de Admin, Soporte o Cliente y modificar cuantas veces se necesite. Cuenta con un Listado de Usuarios que muestra tantos los usuarios creados por esta sección como los creados en el Formulario de Registro de la vista del cliente.** El listado cuenta con un botón para ver todos los detalles del cliente creado. Cuenta con un filtro.



**CRUD de Ofertas**

Sección donde se crean promociones a los productos existentes con tiempo en específico de duración. En el formulario de creación, se puede crear una oferta de cualquier producto creado anteriormente en el CRUD de Productos, definir **cuánto stock se requiere para la oferta a crear, agregar un título para la oferta al momento de mostrarse en la página, descripción de la oferta, cuánto descuento en porcentaje aplicarle a la oferta y definir la fecha de vigencia de la promoción**. Cuenta con un listado que otorga la visualización de **cada oferta creada con todos sus detalles y el estado de la oferta, también editable**. Cuenta con un filtro en la parte superior.



**CRUD de Cupones**

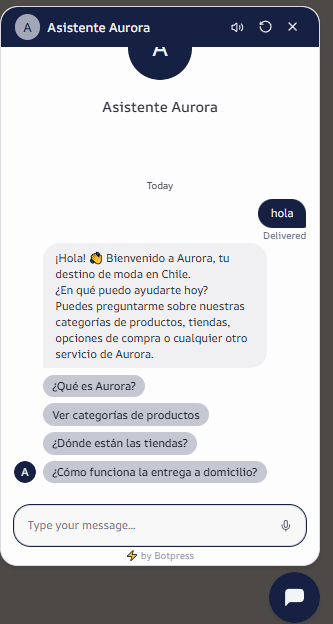
Sección diseñada para la creación de cupones, los cuales serán aplicados por los mismos clientes y , así, recibir un descuento dependiendo las reglas de cada cupón creado. El formulario de creación de cupones de descuento cuenta con la definición del **código a escribir por parte del usuario, nombre del cupón, descripción del cupón, tipo de descuento (descuento en porcentaje o un monto en específico), valor del descuento dependiendo si es en porcentaje o es un monto en particular, mínimo de compra para hacer válido el cupón, “stock” o límite de usos para el cupón en cuestión y la definición de la fecha para hacer uso del cupón de descuento.** Cuenta con un listado que recoge la información específica de cada cupón creado: ID, Código, Nombre del código, Descripción del Código, Descuento, Cantidad de usos, Fecha de vigencia y Estado actual del código. Se puede editar, pausar y eliminar el código en caso de requerirse. Cuenta con un filtro.



**ChatBot con IA**

La funcionalidad “fuerte”. Se encuentra en la página principal, el cual ayuda al cliente con preguntas o información de la tienda, ya sea, algún producto, stock, descuento o cupón vigente.

Fue añadido con el fin de ayudar al equipo de soporte para que los clientes puedan responder todas sus dudas posibles. Responde cualquier cosa acerca de la tienda. El bot se alimenta en base a información específica de la tiene que se le ingesta previamente para que pueda responder coherentemente y de una forma personalizada a cada usuario que lo utilice.



# 

# 

# **Intereses y proyecciones profesionales**

Respecto a una reflexión sobre los intereses, este proyecto fue un cable a tierra muy positivo. Como equipo de trabajo, coincidimos en que nos sirvió para confirmar lo mucho que nos gusta el desarrollo y que es un trabajo que culmina aquí, con un proyecto semestre con muchas altas y muchas bajas pero que supimos sacarlo adelante con convencimiento, sobre todo por la parte en la que integramos diferentes sistemas y los unimos a uno solo. Ver cómo el código que uno diseña y escribe puede solucionar el dolor real de alguien, de un negocio o de un problema común (como lo fue la falta de stock o la atención lenta en Aurora) es gratificante para cada uno de nosotros. Nuestros intereses evolucionaron a lo largo de toda la carrera y no dejaron de crecer ahora, con el trabajo, la programación y el uso de lo que es la inteligencia artificial incluida en un bot, respondiendo personalizadamente lo que uno va requiriendo y entender cómo usar estos modelos para crear valor real en una empresa.

Respecto a las proyecciones laborales, el equipo de trabajo está de acuerdo en lo siguiente: nos gustaría seguir interiorizándonos más en el área de la inteligencia artificial y manejo de datos. Nos proyectamos trabajando en eso luego de lo que fue la experiencia de resolver los problemas que proponía la Tienda Aurora. Nos gustaría profundizar en áreas de Desarrollador Full-Stack o en áreas de Ingeniería de datos con IA. Sentimos que quedamos preparados, no solo para diseñar y proponer una solución con código, sino para entender el ciclo completo de un producto digital, desde la base de datos hasta que el usuario final interactúa con el bot.

# **Conclusión**

Al llegar al final de este camino con nuestro proyecto “ShopIA”, más que simplemente entregar el software final ya terminado, como equipo hemos cerrado una etapa crucial de nuestra formación profesional. Una de las más importantes en nuestras vidas. Mirando hacia atrás, lo que comenzó como un desafío académico para ayudar a la Tienda Aurora se convirtió en una experiencia de aprendizaje real, fructífera y profunda.

Este proyecto nos enseñó que la tecnología no tiene ningún tipon de sentido si es que no resuelve problemas reales, de gente real. Nuestro objetivo no era solo “conectar una base de datos” o “escribir código en un lenguaje en específico”, sino que se trató de aliviar la carga de trabajo de un equipo de soporte que se veía sobrepasado, saturado y sin respuestas inmediatas para salir a flote. Nuestro equipo logró mejorar la experiencia de compra de clientes que buscaban respuestas rápidas y personalizadas y también redujimos la carga de todo el personal en Aurora, entregándoles un software bien estructurado y robusto para lo que necesiten. Ver al chatbot funcionando, luego de todo el proceso que conlleva llegar hasta esta funcionalidad, respondiendo dudas sobre stock y pedidos por sí solo fue la prueba tangible de que logramos entender y solucionar esa necesidad que tenían en Aurora.

El recorrido no fue para nada sencillo. Como equipo de trabajo tuvimos momentos de frustración que nos obligó a atrasarnos y parar a descansar varios minutos, especialmente cuando las piezas no encajaban, como lo fue por ejemplo cuando el sitio web y el servidor perdían la sesión al momento de redireccionarnos a otro lado. Sin embargo, fueron justamente esos obstáculos los que nos hicieron replantearnos las cosas, tomar el descanso debido y crecer en el proceso.

Hoy, con este documento dado por cerrado como lo es nuestra etapa como estudiantes y aprendices en este mundo, el proyecto queda como un testimonio de nuestra capacidad para integrar distintas tecnologías y microservicios y crear algo robusto tecnológicamente y funcional para el cliente y sus dolores. Pero, sobre todo, nos llevamos la certeza de que estamos preparados y bien formados de aquí en adelante para enfrentar el mundo laboral como corresponde. ShopIA fue nuestro campo de entrenamiento final, y nos demostró que tenemos las herramientas para trabajar en equipo, bajo presión algunas veces, y que somos capaces de enfrentar cualquier desafío tecnológico que se nos presente en el futuro.