Logotipo

Descripción generada automáticamente

Proyecto Mechanic-Kings (Taller Mecánico)

Integrantes: Manuel Collao (Desarrollador - DBA)

Yzan Baez (Scrum Máster - Desarrollador)

Anibal Perez (Desarrollador Front – Qa tester)

Benjamin Vidal (Product Owner – Desarrollador)

Índice

Contenido

[Razón del proyecto. 3](#_heading=h.2et92p0)

[**Objetivos Generales 3**](#_heading=h.oixsnf600y8)

[**Alcances del proyecto 3**](#_heading=h.lw9pomnkx56f)

[**Limitaciones del proyecto 4**](#_heading=h.oyfrkdydzdo7)

[Metodología que se ocupa. 4](#_heading=h.4d34og8)

[Arquitectura de Software 5](#_heading=h.g5j5rxo7tshi)

[**Modelo de Datos. 6**](#_heading=h.59j50s14m4fc)

[**Modelo de las tablas 8**](#_heading=h.vr4wovlt39x)

[Proyecto Final 18](#_heading=h.2s8eyo1)

[**Vista administrador 28**](#_heading=)

[**Conclusión General 28**](#_heading=)

# 

# 

# 

# **Razón del proyecto.**

Crear un Sistema para el Registro y Control del servicio de mantenimiento en taller mecánico, el cual pueda registrar los datos del cliente e ir administrando las mantenciones del vehículo en base al correo o el número telefónico del cliente. A su vez se venderán productos automotrices.

# **Objetivos Generales**

1. Facilitar la gestión integral de citas para el mantenimiento y revisión de vehículos de clientes.
2. Optimizar la venta y administración de productos automotrices relacionados con el mantenimiento.

**Objetivos Específicos**

1. Registrar y gestionar la información de clientes y sus vehículos.
2. Llevar un control eficiente de los clientes registrados.
3. Registrar y gestionar el historial de mantenimientos de los vehículos.
4. Organizar y administrar las citas para mantenimiento o revisión de vehículos.
5. Controlar y administrar el stock de productos automotrices.

# **Alcances del proyecto**

* Funcionalidades principales :
  + Agendamiento de servicios : Clientes pueden programar citas para mantenimiento automotriz.
  + Gestión de productos y servicios : Se permite administrar productos(agregar, editar y eliminar) y servicios ofrecidos por el taller
  + Chatbot integrado : Asistencia para realizar agendamientos.
  + Gestión de clientes : Perfiles de usuarios con historial de compras, servicios , vehículos registrados.
  + Sistemas de pagos : Integración con Transbank para realizar pagos de manera segura y eficiente.
* Interfaz de usuario:
  + Sistema responsivo : Interfaz diseñada para funcionar en dispositivos móviles, tablets y desktops.
  + Visualización en tiempo real : Actualización dinámica del carrito de compras
* Integraciones externas :
  + Transbank: Procesamiento de pagos para los productos y servicios.
  + Twilio: Envío de notificaciones y comunicación vía WhatsApp.

# **Limitaciones del proyecto**

* Dependencias de servicios externos:
  + El funcionamiento del chatbot y las notificaciones depende de la disponibilidad y estabilidad de Twilio
  + Los pagos están a través de Transbank, por lo que problemas en su API podrían afectar a la funcionalidad del sistema
* Presión al tiempo:
  + En el sprint 3 tuvimos percances respecto al tiempo ya que algunas tareas nos llevaron más tiempo de lo pensado.

# **Metodología que se ocupa.**

El proyecto se llevó a cabo con la metodología scrum ya que realizamos un proyecto de varias partes por lo cual era fundamental dividir cada etapa de los avances en sprints. Cada sprint que desarrollamos lo comentamos en reuniones a través de discord para hacer los respectivos avances por sprint y a su vez reportar sobre los avances de las tareas en cada sprint. Siempre que se termina cada sprint realizamos una respectiva reunión para identificar los obstáculos y así tratar de evitarlos para los sprints futuros. Con esta metodología logramos realizar nuestro proyecto en conjunto, analizando cada etapa para mejorar la forma de trabajo en el equipo.

### **Arquitectura de Software**

la arquitectura **MTV (Modelo-Template-Vista)** proporcionada por Django. Esta arquitectura organiza el sistema en tres capas principales: modelo, lógica de negocio y presentación, lo que permite una clara separación de responsabilidades. Gracias a esta estructura, se facilita el desarrollo, mantenimiento y escalabilidad del sistema.

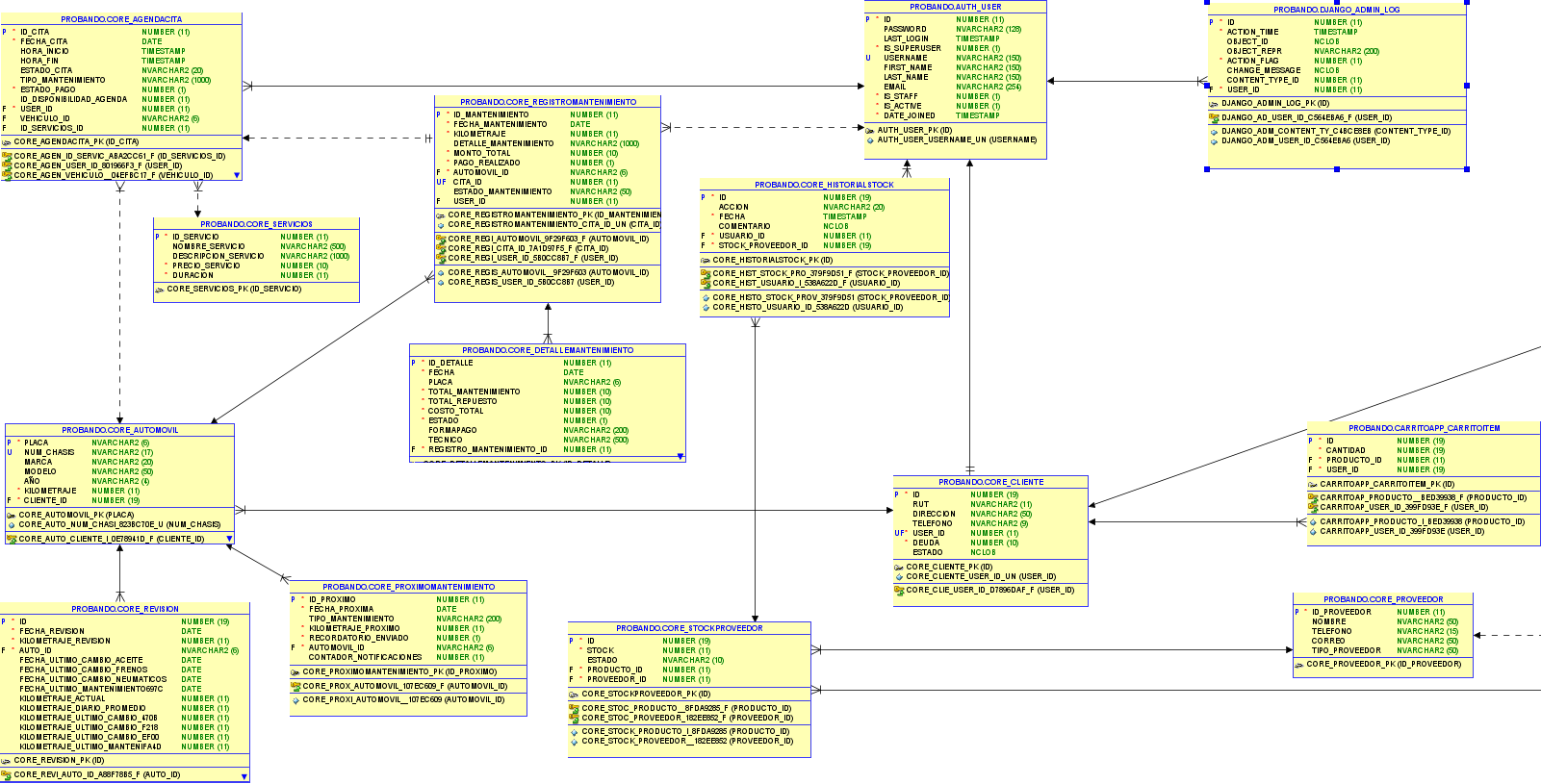
#### **Componentes de la Arquitectura MTV**

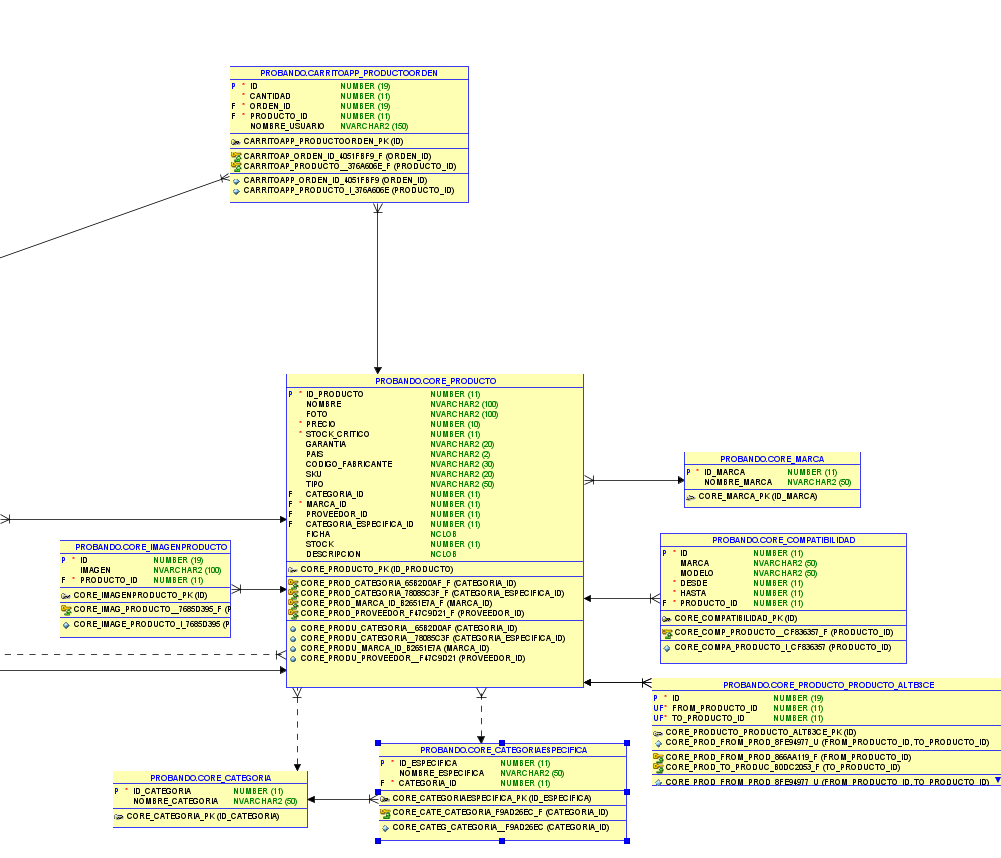
1. **Modelo:**La capa de modelo define la estructura de la base de datos y las relaciones entre las entidades principales del sistema, como *Cliente*, *Automóvil*, *Producto* y *Mantenimiento*. Esto permite gestionar datos fundamentales, incluyendo historiales de mantenimientos, inventarios y transacciones, asegurando que la información esté organizada y estructurada de forma eficiente.
2. **Vista:**En esta capa se implementa la lógica de negocio, que incluye el procesamiento de solicitudes de los usuarios y la interacción con los modelos para recuperar o actualizar datos. Por ejemplo, las vistas controlan la programación de mantenimientos, la gestión de inventarios o la realización de transacciones de compra. Estas vistas pueden ser de dos tipos:
   * **Vistas basadas en funciones (FBV)** para lógica específica.
   * **Vistas basadas en clases (CBV)** para reutilizar comportamientos comunes, como creación o edición de registros.
3. **Plantilla:**La capa de plantillas se encarga de la interfaz de usuario. Utiliza HTML dinámico y el Django Template Language (DTL) para mostrar datos de manera interactiva, como facturas, listados de productos o registros de clientes. Esto asegura una experiencia de usuario clara y eficiente.

#### **Beneficios de la Arquitectura MTV**

* **Escalabilidad y Mantenimiento:**
  + La separación de responsabilidades permite que el sistema crezca con nuevas funcionalidades sin afectar las partes existentes. Por ejemplo, se puede integrar un módulo de pagos en línea sin modificar la presentación o el modelo de datos.
  + El mantenimiento del sistema se simplifica, ya que los cambios en una capa no repercuten directamente en las otras. Por ejemplo, ajustar una lógica de negocio en las vistas no requiere alterar las plantillas ni los modelos.
* **Soporte para Procesos Complejos:**La arquitectura soporta la implementación de procesos complejos, como la integración con APIs externas (por ejemplo, Transbank para pagos) o la generación de reportes. La capa de vistas es ideal para manejar estas interacciones y, en proyectos más avanzados, se puede añadir una capa de servicios para desacoplar lógica compleja y facilitar integraciones externas.
* **Reutilización de Código:**Las vistas y plantillas pueden ser diseñadas para reutilizarse en múltiples partes del sistema. Esto optimiza los tiempos de desarrollo y reduce redundancias. Por ejemplo, una misma vista puede manejar tanto la creación de registros de productos como su actualización.
* **Facilidad de Pruebas:**El diseño modular de MTV facilita la ejecución de pruebas en cada componente:
  + **Modelos**: Validación de relaciones y restricciones en los datos.
  + **Vistas**: Comprobación de la lógica de negocio.
  + **Plantillas**: Verificación de la presentación de los datos.  
    Esto permite un desarrollo ágil y confiable, especialmente útil en entornos donde se sigue la metodología Scrum, con entregas frecuentes de incrementos funcionales.

# **Modelo de Datos.**





# 

# 

# 

# 

# **Modelo de las tablas**

AGENDARCITA

Texto

Descripción generada automáticamente

**Relaciones**:

Cliente (FK): Relaciona la cita con el usuario que la agenda a través del campo user.

Vehículo (FK): Asocia la cita con un vehículo específico del cliente.

Servicios (FK): Permite asignar un servicio específico de mantenimiento.

**Descripción**: La tabla AgendaCita está diseñada para gestionar la programación de citas de mantenimiento de vehículos de los clientes. Cada cliente puede registrar uno o más vehículos, y esta tabla permite asociar citas específicas con vehículos, servicios y horarios disponibles. Además, facilita el seguimiento de información clave relacionada con la cita, como el estado, el tipo de mantenimiento requerido y el estado de pago.

El objetivo principal de esta tabla es centralizar los datos esenciales para organizar, gestionar y realizar mantenimientos vehiculares, optimizando la comunicación entre clientes y el taller. Incluye información como fechas y horas específicas, el tipo de servicio solicitado, el vehículo involucrado, y el cliente responsable.

REGISTROMANTENIMIENTO

Texto

Descripción generada automáticamente

**Relaciones**:

Automóvil (FK): Relaciona el mantenimiento con el vehículo al que se le realizó el servicio.

Usuario (FK): Relación con el cliente al cual le realizamos este mantenimiento.

Cita (FK): Relación opcional con la tabla AgendaCita para asociar el mantenimiento con una cita específica.

**Descripción**: Esta tabla está creada para llevar un registro detallado de los mantenimientos o reparaciones realizados en los vehículos de los clientes. Este registro incluye información clave sobre el mantenimiento. Además, está diseñada para asociar cada registro con una cita específica y con el vehículo y cliente correspondientes.

La finalidad de esta tabla es centralizar y documentar el historial de mantenimientos realizados, asegurando una trazabilidad clara de los servicios prestados. Esto facilita tanto el seguimiento de los vehículos como la gestión interna del taller, permitiendo también responder a consultas de los clientes de manera eficiente.

SERVICIOS

Texto, Tabla

Descripción generada automáticamente

**Relación**: Ninguna

Descripción: Esta tabla tiene la finalidad de almacenar la información sobre los servicios que ofrece el taller mecánico. Su propósito es centralizar y mantener un catálogo actualizado de los diferentes tipos de servicios, incluyendo detalles sobre su descripción, duración estimada y precio. Esta tabla no tiene relaciones directas con otras tablas, ya que funciona como un registro independiente.

DETALLEMANTENIMIENTO

Tabla

Descripción generada automáticamente

**Relación:**

RegistroMantenimiento (FK): Relación con registro de mantenimiento, permitiendo asociar este detalle con la información del trabajo realizado.

**Descripción:** La tabla DetalleMantenimiento está diseñada para generar un registro detallado de los mantenimientos realizados, similar a una boleta o factura. Contiene información clave como el costo total, los costos desglosados por mantenimiento y repuestos, el técnico responsable, y la forma de pago. Además, está relacionada con el registro de mantenimiento específico para garantizar la trazabilidad del registro realizado.

PROXIMOMANTENIMIENTO

Tabla

Descripción generada automáticamente

**Relación**:

Automóvil (FK): Cada registro está vinculado a un vehículo específico, lo que permite gestionar los próximos mantenimientos de forma individualizada.

**Descripción**: La tabla ProximoMantenimiento está diseñada para llevar un control sobre los mantenimientos futuros de los vehículos de los clientes. Esta tabla genera un registro de los próximos mantenimientos que un automóvil necesita realizar, basándose en el tipo de servicio y el kilometraje estimado para cada mantenimiento. Además, incluye un sistema de notificación para recordar al cliente acerca de los mantenimientos pendientes. Esta tabla es fundamental para la planificación y la automatización de mantenimientos.

REVISION

Tabla

Descripción generada automáticamente

**Relación**:

Automóvil (FK): Cada revisión está asociada con un vehículo específico, lo que facilita el seguimiento del historial de mantenimiento.

**Descripción**: La tabla Revisión tiene como objetivo registrar datos clave sobre el uso y mantenimiento de los vehículos, permitiendo automatizar el cálculo de fechas y kilometrajes estimados para futuros mantenimientos. Al capturar datos como kilometraje diario promedio y últimas fechas de servicios, esta tabla se convierte en un soporte fundamental para la tabla ProximoMantenimiento.

CLIENTE

Tabla

Descripción generada automáticamente

**Relación**:

User (FK): Relación uno a uno con el modelo User de Django para autenticación.

**Descripción**: La tabla Cliente está diseñada para almacenar información detallada sobre los clientes del taller mecánico. Esto incluye sus datos personales, contacto, estado de sus conversaciones en WhatsApp.

AUTOMOVIL

Tabla

Descripción generada automáticamente

**Relación**:

Cliente (FK): Relación con el cliente propietario del vehículo.

**Descripción**: La tabla Automovil almacena información de los vehículos registrados de cada cliente. Los clientes pueden tener múltiples vehículos asociados, permitiendo un seguimiento detallado de cada uno.

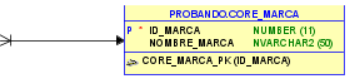
PRODUCTO

Imagen que contiene Texto

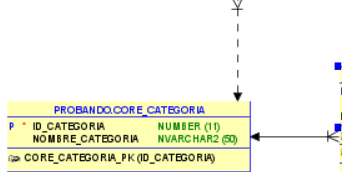
Descripción generada automáticamente

**Relación**:

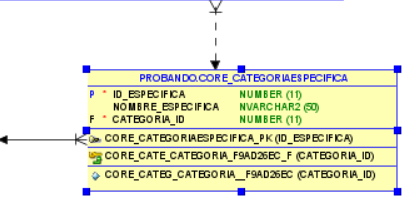
Marca (FK): Relaciona el producto con su marca.



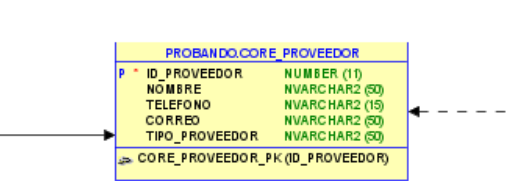
Categoría (FK): Categoría general del producto.



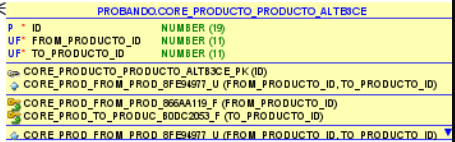
Categoría Específica (FK): Subcategoría específica del producto.



Proveedor (FK): Relación con el proveedor del producto.



Producto Alternativo (ManyToMany): Relación con otros productos que pueden ser alternativos o sustitutos.



**Descripción:** La tabla Producto se utiliza para registrar los productos que el taller mecánico vende o utiliza en sus servicios. Esto incluye repuestos, accesorios, y herramientas.

IMAGEN PRODUCTO

Tabla

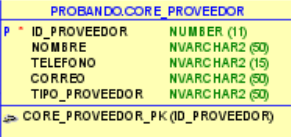
Descripción generada automáticamente

**Relación:**

Producto (FK): Relación con la tabla Producto, que vincula cada imagen a un producto específico.

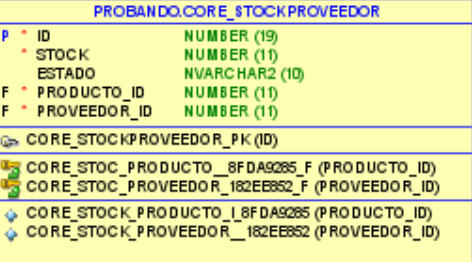
**Descripción:** La tabla Imagen Producto está diseñada para almacenar imágenes asociadas a un producto en particular. Esto permite crear una galería de imágenes para cada producto, enriqueciendo su descripción visual.

PROVEEDOR (con sus derivados)



**Descripción:** La tabla Proveedor se utiliza para tener un registro de a qué proveedor pertenece el stock del producto en base a STOCKPROVEEDOR.

STOCKPROVEEDOR



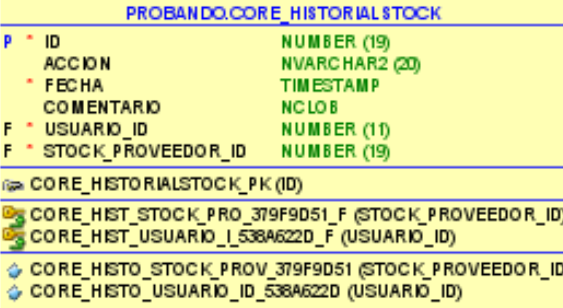
**Relación:**

Producto (FK): Relación con la tabla Producto, que vincula cada stock agregado a un producto específico.

Proveedor (FK): Relación con la tabla Proveedor, que vincula cada stock agregado a un Proveedor específico.

**Descripción:** La tabla STOCKPROVEEDOR se utiliza en conjunto a la tabla proveedor para crear nuevos proveedores y poder añadir nuevo stock con esos proveedores, al igual que se puede añadir más stock con el proveedor que ya tiene incluido.

HISTORIALSTOCK



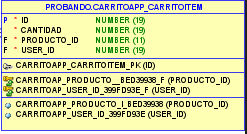
**Relación:**

StockProveedor (FK): Relación con la tabla StockProveedor, esta relación se usa para poder extraer los datos del usuario staff que creó ese proveedor y añadió o rechazo stock con algunos de los proveedores registrados.

Usuario (FK): Relación con la tabla Usuario, esto se ocupa mediante una validación de que el usuario tiene que ser staff para ver los stock agregados con sus respectivos datos (Acción (Aceptada, Rechazada), Fecha y Comentario)

**Descripción:** La finalidad de esta tabla es tener la información al momento de ver si se añade stock o se rechaza en una tabla con sus respectivos datos visualizados (un control).

CARRITO ITEM

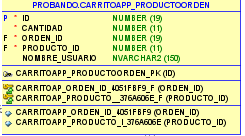


**Relación :**

Se crea una llave foránea de usuario ID para que cada cliente tenga un carro de compras propio.

**Descripción :** La tabla producto está diseñada para almacenar todos los productos que el cliente vaya agregando, no afectando a los demás clientes ya que está validado por usuario, que cada carro sea diferente y guarde los datos del respectivo cliente.

PRODUCTOORDEN



Descripción : Se creó la tabla producto orden para que registre las órdenes de cada cliente al comprar productos según su cantidad, orden id, nombre de usuario, producto id. La tabla está directamente relacionada con la tabla cliente.

ORDENCOMPRA



Descripción : Tabla orden compra donde se almacenarán las compras realizadas por el cliente, con su monto total, el método de pago, la fecha de pago, cliente id (al usuario que realizó la compra) con su nombre de usuario que se muesran en la base de datos.

# 

# 

# 

# 

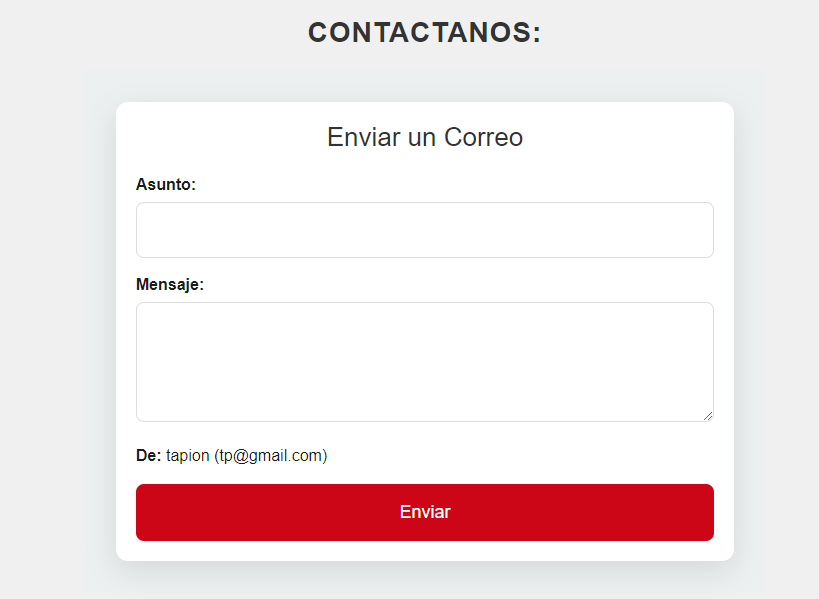
# Proyecto Final

**Página Home.**



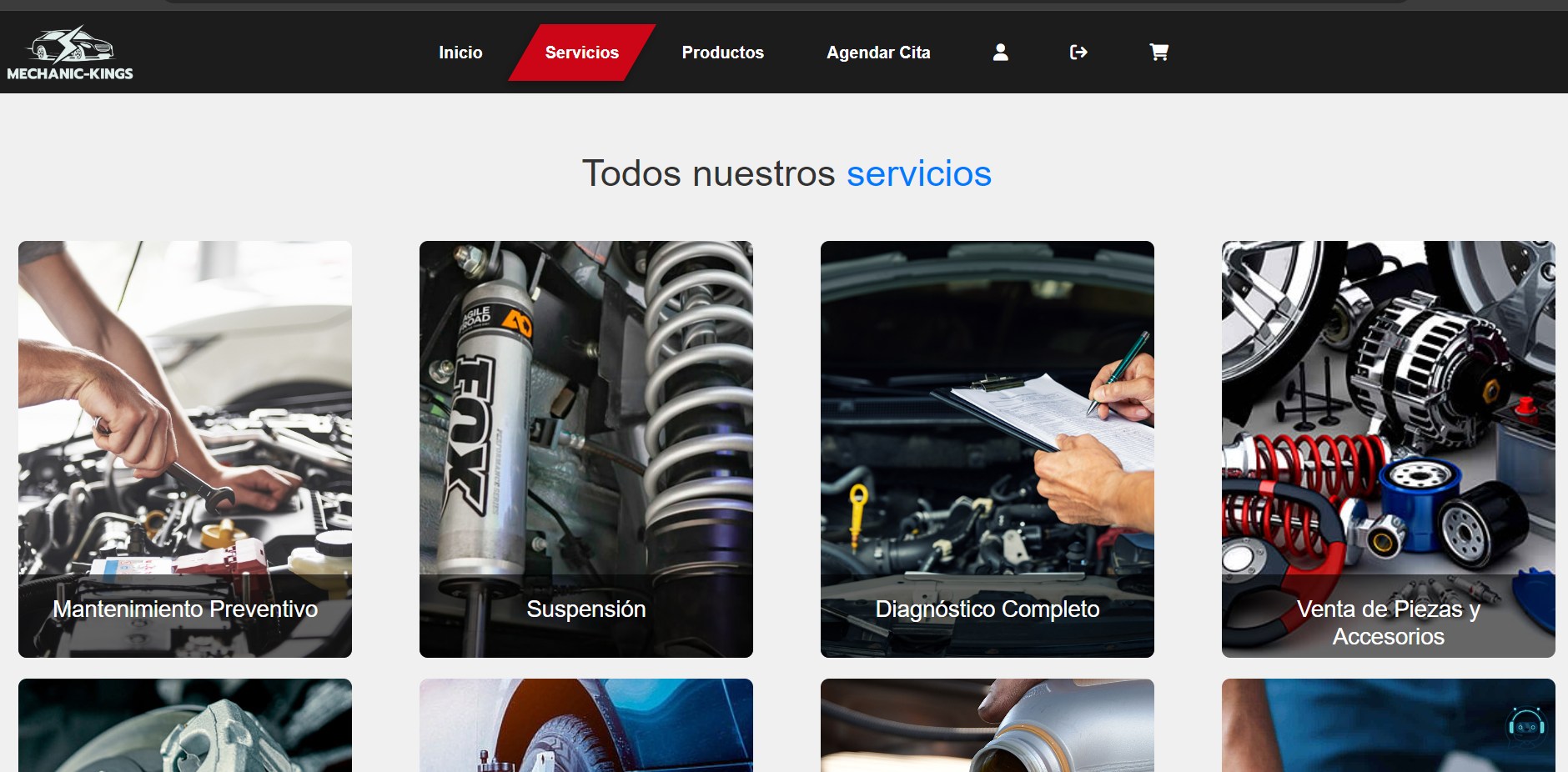
Página de inicio al usuario donde se muestran las marcas con las que se trabaja, los servicios que entregamos y una breve reseña acerca de nuestro servicio.

**Apartado contacto.**

****

El apartado contacto se encuentra en el home abajo de los servicios, donde el cliente puede realizar consultas acerca de servicios/horarios, etc. Al estar con la sesión iniciada se indica el correo del usuario actual, evitando tener que ingresar el correo.

**Página Servicios.**



Página de servicios donde se enfatiza y se recalcan los servicios que se ofrecen en el taller mecánico.

**Página Productos**

Captura de pantalla de un celular

Descripción generada automáticamente

Esta es la página de los productos, donde se mostrarán todos los productos disponibles para la venta.

**Filtro de productos (Precio, categoría, subcategoría, marca).**

Interfaz de usuario gráfica, Sitio web

Descripción generada automáticamente

Para hacer una búsqueda más detallada de los productos que se requieren según el usuario, se incorporó un filtro de productos donde va por el precio, la categoría, la subcategoría de este y la marca de dicho producto.

**Página Detalle Productos.**

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Página donde se detallarán los aspectos del producto, tanto el precio, descripción, SKU, etc.

**Carrito de compra - Lateral**

Interfaz de usuario gráfica, Sitio web

Descripción generada automáticamente

Carrito lateral donde se mostrarán los productos que el cliente irá agregando al carrito, junto con la cantidad agregada y el total de precio de todos.

**Página Agendar Cita (Cliente)**

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Para agendar cita primero debes tener un vehículo registrado, de lo contrario te dirigirá al perfil del usuario para agregar un vehículo.

**Página Agendar Cita (Cliente)**

Interfaz de usuario gráfica

Descripción generada automáticamente

Donde se agendan las citas del cliente, indicando su servicio y la hora disponible que hay, ya que las horas ya ocupadas no se mostraran al agendar otra cita.

**Resumen Cita**

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

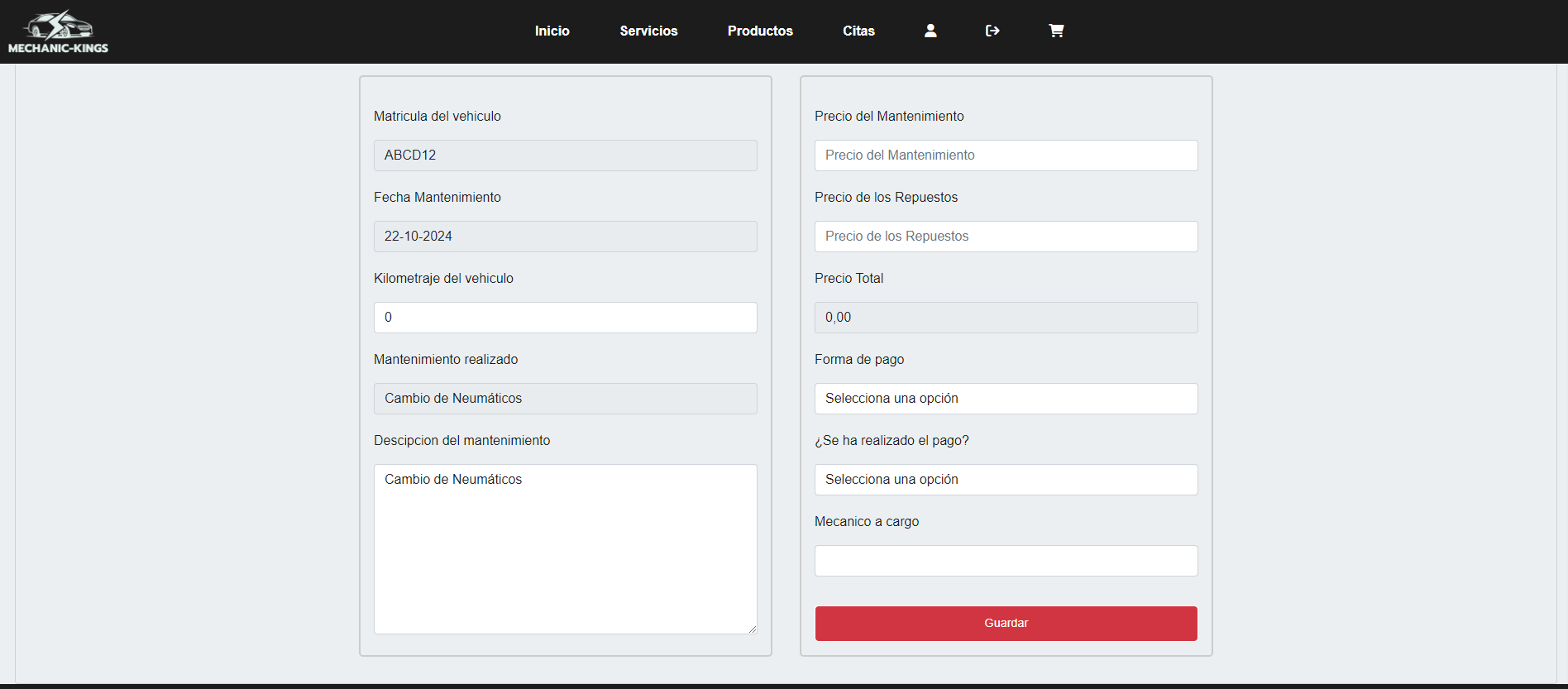
Se muestra el resumen de la cita con su servicio, vehículo, precio, duración, fecha y la hora.

**Página Agendar Cita (Admin)**



Logueado como administrador, en citas se visualiza las citas agendadas por cliente para tener un control al día.

**Página Agendar cita Detalle**



Al darle clic al botón actualizar se mostrará más a detalle la cita agendada.

**Página Perfil de usuario**

Interfaz de usuario gráfica

Descripción generada automáticamente

Perfil del usuario logueado junto con los Vehículos que tiene registrado, se mostraran las compras realizadas por parte del usuario. Además de las citas agendadas por el usuario, se dispondrá de los mantenimientos realizados junto con su boleta.

**Página Editar perfil**

Captura de pantalla de computadora

Descripción generada automáticamente

Página editar perfil del usuario, donde se podrá actualizar la información del perfil, se verán los vehículos registrados y las revisiones agendadas.

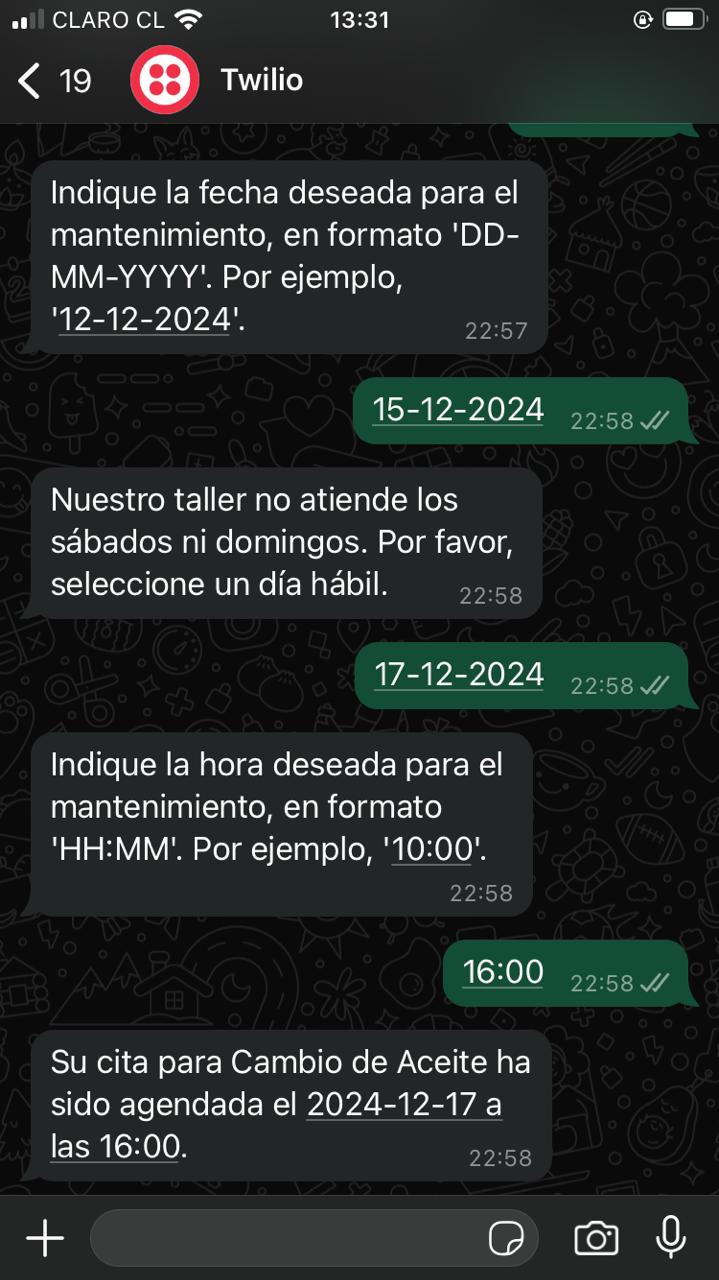
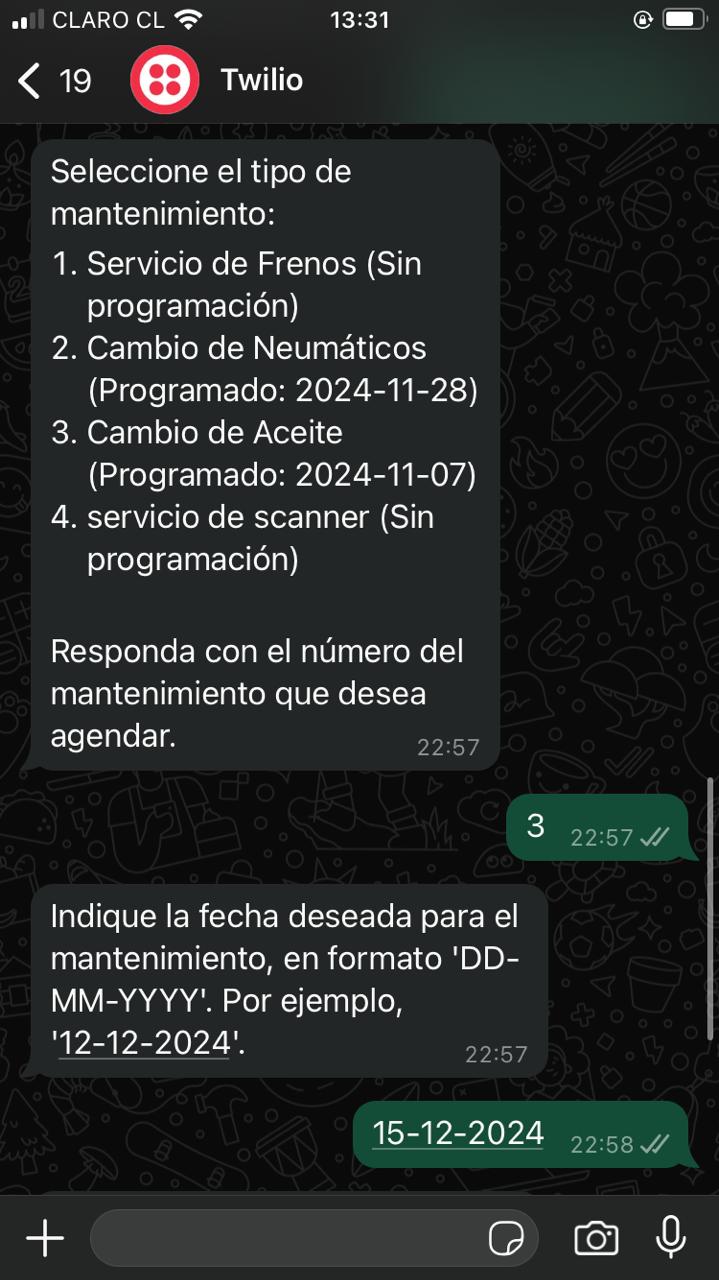
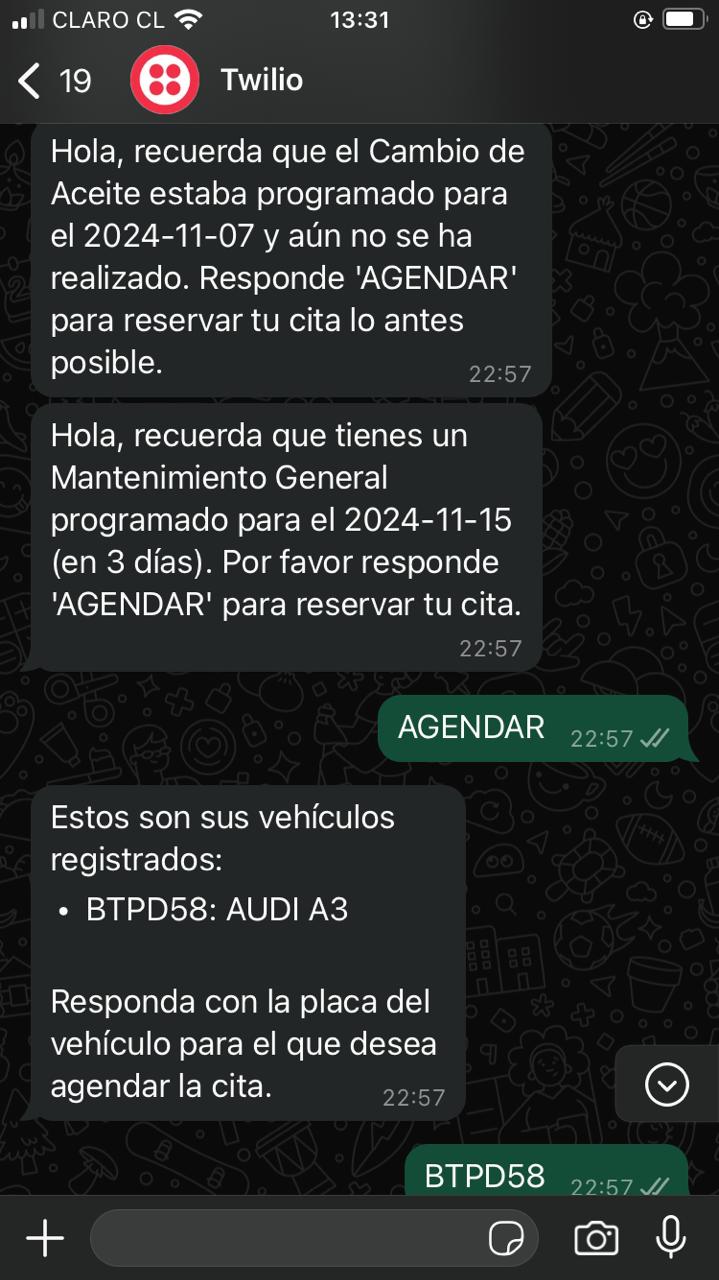
**Chatbot**

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

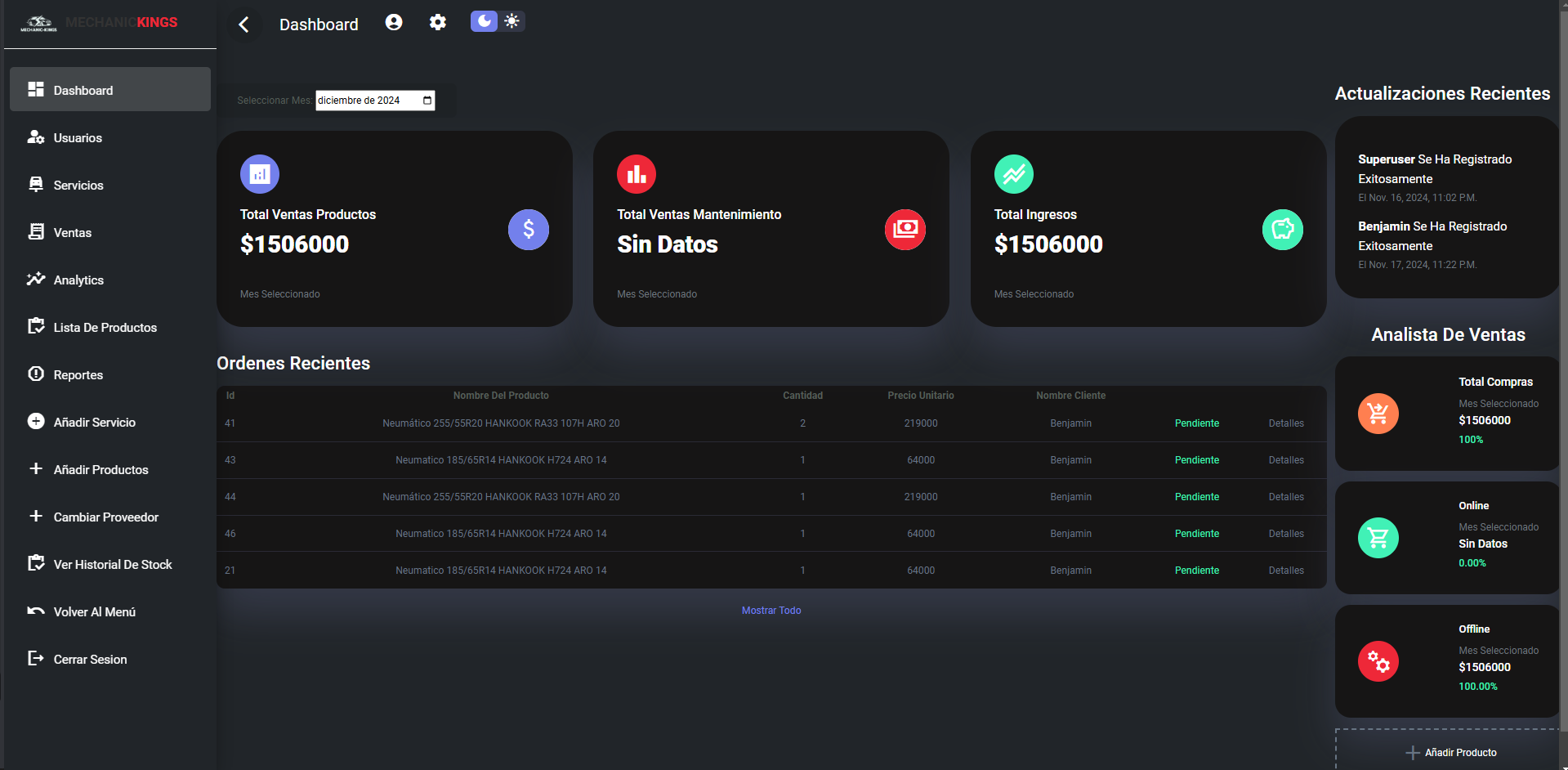
En el apartado del chatbot se mostrará las opciones: buscar un producto, ver servicios disponibles, contacto y contactar por WhatsApp. El contacto por WhatsApp está configurado a través de Twilio, donde se podrán ver las horas agendadas por el cliente además de agendar un mantenimiento.

Apartado Contactar por Whatsapp



A través de twilio el cliente puede agendar su hora para realizar el servicio que desee, y si previamente haya seleccionado un servicio, dia y hora para realizarlo.

# Vista administrador



Es el dashboard del administrador donde se pueden ver las órdenes recientes , el total de ventas de los productos por mes, total mantenimientos y los ingresos en general del taller mecánico.

# **Conclusión General:**

Para cerrar , cabe decir que el proyecto de Taller Mecánico Mechanic-King ha sido un largo proceso en el cual se han llevado a cabo conocimientos técnicos para elaborar una plataforma web que permita realizar agendamiento de citas con diferentes servicios ofrecidos, facilita la gestión de servicios de mantenimientos vehiculares y a la venta de productos automotrices. También cabe destacar las incorporaciones de servicios externos como Transbank y Twilio , que eran sumamente importantes y relevantes para el éxito de este proyecto grupal.

La metodología SCRUM fue una buena elección ya que nos permitió ir mejorando en lo que avanzaba el proyecto en cuanto a efectividad de desarrollo y también identificar las falencias que teníamos a la hora de terminar cada sprint , siendo esto de gran ayuda para que el proyecto saliera adelante como teníamos planeado al principio de todo, con todas las funcionalidades.