



Universidad Autónoma Gabriel Rene Moreno

‘Facultad de Ingeniería en Ciencias de la Computación y Telecomunicaciones’

**Sistema de Información para la Gestión de Procesos
Administrativos, Clínicos y de Comunicación en Clínicas
Veterinarias ‘VetCare’.**

Numero de Grupo: 2

Materia: Sistemas de Información 1

Grupo Materia: SA

Docente: Msc. Ing. Angélica Garzón Cuellar

Gestión: 2/2025

INTEGRANTES
Auad Castillo Miguel Andres
Cervantes Arancibia Roberto Carlos
Crespo Cardenas Gabriela Belen
Melgar Gushi Diego
Ortiz Montero Luis Enrique
Rojas Rivero Romer

Contenido

1.- PERFIL.....	5
1.1. INTRODUCCIÓN	5
1.2. .-ANTECEDENTES	6
1.3. .-JUSTIFICACION.....	8
1.4. -DESCRIPCIÓN DEL PROBLEMA.....	8
1.4.1. Gestión de citas y turnos.....	9
1.4.2. Historias clínicas dispersas	10
1.4.3. Control de inventario de medicamentos e insumos	11
1.4.4. Procesos administrativos y facturación	12
1.4.5. Comunicación con los clientes	13
1.5. .- FORMULACIÓN DEL PROBLEMA.....	14
1.6. – OBJETIVOS	15
1.6.1. - OBJETIVO GENERAL.....	15
1.6.2. - OBJETIVOS ESPECIFICOS.....	15
1.7. - ALCANCE.....	16
2. ELEMENTOS DEL SISTEMA BASADO EN COMPUTADORAS	22
2.1. Hardware	22
2.1.1. Servidor	22
2.1.2. Cliente	23
2.1.3. Medios de comunicación	23
2.1.4. Otros dispositivos.....	23
2.2. Software	24
2.2.1.1 Cliente	24
2.2.1.2 Otro software adicional	24
2.3. Datos.....	24
2.4. Procesos.....	27
2.5. Gente/Usuario	28
2.6. Documento	28
3. TECNOLOGÍA PARA EL DESARROLLO DEL SOFTWARE	29
3.1. ESTRATEGIA PARA EL DESARROLLO DEL SOFTWARE	29
3.2. METODOLOGÍA PARA EL DESARROLLO DEL SOFTWARE	29
3.2.1. CARACTERÍSTICAS DEL PUDS.....	29

3.2.2. CARACTERÍSTICAS DE UML.....	32
3.3. HERRAMIENTAS DE DESARROLLO	33
3.3.1. SOFTWARE	33
3.3.2. HARDWARE	34
4. POSIBLES COSTOS.....	36
4.1. COSTO DE HERRAMIENTAS DE DESARROLLADORES	36
4.2. COSTO DE DESARROLLO	37
4.3. COSTO DE IMPLEMENTACIÓN	37
5. POSIBLES BENEFICIOS	38
5.1. TIEMPO	38
5.2. ESFUERZO	38
5.3. COSTOS	39
6. POSIBLES CLIENTES	39
7. Marco teórico.....	40
CAPITULO 1: METODO DE ISHIKAWA	41
1.-Identificar Problemas.....	41
1.1.- Lista de problemas.....	41
1.2.- Depurar Problemas.....	42
1.3.- Lista Final de Problemas	43
1.4.- Propietarios De Problemas	43
1.5.- Análisis de problemas	46
1.6.- Estimación y cuantificación del problema	47
1.7.- Alternativas de Cambio.....	48
1.8.- Conclusión y Recomendación	49
1.9.- Diseñar el Diagrama de Ishikawa.....	50
2.- Identificar las principales categorías	51
3.- Identificar las causas.....	51
4.- Analizar y discutir el diagrama.....	52
MODELO DE NEGOCIO.....	54
Diagrama de actividades.....	54
CAPITULO 2.- FLUJO DE TRABAJO: CAPTURA DE REQUISITOS.....	56
2.1.- Encontrar Actores y Casos de uso:	56
2.2.- Priorizar Casos de Uso:	60

2.3.- Detallar Casos de uso:.....	62
2.4.- Prototipar la Interfaz de Usuario:.....	84
2.5.- Estructurar Modelos de Casos de Uso:.....	94
CICLO 1.-.....	94
CICLO 2.-.....	95
CICLO 3.-.....	96
CAPITULO 3 FLUJO DE TRABAJO: ANALISIS.....	97
3.1.- Análisis de Arquitectura:.....	97
3.1.1.- Identificar Paquetes:.....	97
3.1.2.- Relacionar Paquetes y Casos de Uso:	98
3.1.3.- Vista de Paquetes	101
3.2.- Analizar Casos de Uso	105
3.2.1.- Diagrama de Comunicación	105
3.3.- Analizar una Clase	108
3.3.1.- Ciclo 1.....	108
3.3.2.- Ciclo 2.....	110
3.3.3.- Ciclo 3.....	112
3.4.- Analizar un Paquete.....	116
CAPITULO 4.- FLUJO DE TRABAJO: DISEÑO.....	117
4.1.- Diseño de Arquitectura.....	117
Arquitectura Física (Diagrama de Despliegue).....	117
Arquitectura Lógica (Diagrama de paquete).....	118
4.2.- Diseño de Datos	119
4.2.1.- Diseño de Datos Lógico.....	119
4.2.2.- Diseño De Datos Físico.....	122
4.3.- Diseño de Caso de Uso	154
Diagrama de Secuencia.....	154
CAPITULO 5.- Flujo de Trabajo: Implementación	161
5.1 Elección de plataforma de desarrollo del Software.....	161
5.1.1 Lenguaje de programación	161
5.1.2 Base de Datos.....	161
5.1.3 Sistemas Operativos.....	162
5.1.4 Otros	163

5.2.- Arquitectura Subsistema	163
P1.- GESTION DE USUARIO.....	163
P2.- GESTION CLINICA	164
P3.- GESTION DE INVENTARIO Y PRODUCTOS.....	164
CONCLUSIÓN	165
RECOMENDACIÓN	165
ANEXO.....	166

1.- PERFIL

1.1.INTRODUCCIÓN

El sector veterinario ha experimentado un crecimiento significativo en los últimos años, impulsado principalmente por el aumento de la tenencia responsable de mascotas y la necesidad de garantizar su bienestar. Las clínicas veterinarias no solo brindan servicios médicos, sino que también ofrecen un acompañamiento integral que abarca controles periódicos, vacunaciones, tratamientos especializados y cirugías. Esto las convierte en un pilar fundamental tanto para la salud animal como para la tranquilidad de los propietarios.

Sin embargo, muchas de estas instituciones aún enfrentan dificultades en la organización de sus procesos internos, ya que dependen de registros manuales, agendas físicas o sistemas aislados que no se comunican entre sí. Esta falta de centralización genera problemas como pérdidas de información, duplicidad de datos, retrasos en la atención y errores en historiales clínicos o

facturación. A ello se suma la administración manual de inventarios, que incrementa el riesgo de desabastecimiento o sobrecompra, afectando directamente la calidad del servicio.

En respuesta a estas problemáticas surge VetCare, un sistema de información diseñado para optimizar la gestión de una clínica veterinaria mediante una plataforma web integral. El proyecto busca digitalizar los procesos clave de la institución, facilitando la programación de citas, el control de historiales médicos, la administración de inventarios y el manejo de facturación. Al centralizar la información en un solo entorno, VetCare mejora la eficiencia operativa, garantiza mayor seguridad en el manejo de los datos y contribuye a ofrecer un servicio más ágil, confiable y transparente a los dueños de las mascotas, respondiendo así a las exigencias actuales del sector y al avance de las tecnologías de información.

1.2.. -ANTECEDENTES

Las clínicas veterinarias han desempeñado un papel cada vez más relevante debido al crecimiento sostenido de la tenencia de mascotas y a la demanda de servicios médicos especializados. Sin embargo, en Bolivia gran parte de estos establecimientos aún dependen de procedimientos manuales para la gestión de sus operaciones, lo que genera limitaciones en eficiencia, control y calidad de servicio.

Experiencias documentadas de clientes de diversas clínicas veterinarias han evidenciado que la persistencia de este modelo de gestión genera consecuencias recurrentes: duplicación de citas, pérdida de información, retrasos en la atención, fallas en el seguimiento post-tratamiento y desabastecimiento de insumos médicos esenciales.

Datos Comerciales de la Clínica Veterinaria

Nombre Comercial: ServetCorp.

Ubicación: Radial 13, Av. El Triunfo Esq. San Antonio.

Teléfono: 33596396

Celular: 75593333

Web: Facebook - servetcorp

Instagram - servetcorp

Antecedentes de la Clínica Veterinaria ServetCorp

La clínica veterinaria **ServetCorp**, ubicada en la ciudad de Santa Cruz de la Sierra, constituye un ejemplo representativo de la situación. A pesar de contar con un equipo multidisciplinario y una gama de servicios que abarca consultas, vacunación, cirugías menores, peluquería y venta de productos básicos, la administración de los procesos internos continúa realizándose de manera manual. El registro de citas se efectúa en un cuaderno físico, las historias clínicas se archivan en fichas de cartulina y el inventario de insumos se controla de manera visual, sin mecanismos de actualización ni alertas automáticas.

En el caso particular de **ServetCorp**, la información recabada en entrevistas al personal administrativo confirma la existencia de estos problemas, además de identificar a la recepción como el principal punto de congestión, dado que concentra simultáneamente la atención telefónica, la programación de citas y la búsqueda de historias clínicas.

Por lo tanto, los antecedentes recopilados permiten concluir que la problemática no es aislada, sino parte de una tendencia generalizada en el sector veterinario boliviano. Esta situación justifica la necesidad de implementar soluciones tecnológicas que contribuyan a optimizar los procesos administrativos y médicos, garantizando una atención más eficiente, segura y confiable para los propietarios y sus mascotas.

1.3.. -JUSTIFICACION

La implementación de un sistema de información en clínicas veterinarias responde a la necesidad de superar las limitaciones de los procesos manuales que, si bien fueron funcionales en etapas iniciales, resultan obsoletos ante el crecimiento sostenido de la demanda de servicios.

En este sentido, un sistema como **VetCare** permitirá centralizar y digitalizar los procesos claves de la clínica veterinaria **ServetCorp**. Adicionalmente, el contexto actual exige que las instituciones de servicios incorporen tecnologías de información que respondan a las expectativas de los usuarios, quienes demandan inmediatez, transparencia y accesibilidad, convirtiéndose en un referente de innovación y modernización en el sector.

1.4.-DESCRIPCIÓN DEL PROBLEMA

La clínica veterinaria ServetCorp, al igual que muchas otras en el país, atraviesa una serie de dificultades que obstaculizan su crecimiento y reducen su capacidad de ofrecer un servicio de calidad. Aunque ha experimentado un aumento sostenido en la cantidad de pacientes atendidos y ha ampliado su portafolio de servicios, sus procesos internos aún se sostienen en prácticas manuales, poco estandarizadas y fragmentadas que no responden a las necesidades actuales del sector.

En sus primeros años de funcionamiento, cuando la cantidad de clientes era reducida, el uso de agendas físicas, libretas de apuntes y hojas de cálculo dispersas parecía suficiente para organizar la gestión diaria. Sin embargo, con el crecimiento acelerado de la demanda, estas herramientas se volvieron obsoletas, ineficientes y riesgosas, exponiendo a la clínica a errores, pérdidas de información, retrasos y desorganización general.

Hoy en día, la falta de un sistema centralizado constituye un problema estructural que impacta en cuatro dimensiones clave: la eficiencia del personal, la calidad del servicio, la satisfacción del cliente y la rentabilidad económica. A continuación, se detallan los principales focos de ineficiencia, acompañados de ejemplos reales y comparativos que evidencian la urgencia de una solución digital.

1.4.1. Gestión de citas y turnos

En la actualidad, el registro de citas se realiza en una agenda física ubicada en la recepción. Cada vez que un cliente solicita una cita, la secretaria anota manualmente los datos, lo cual conlleva una alta probabilidad de errores humanos. Esta dinámica genera múltiples problemas:

- Citas duplicadas en un mismo horario, lo que provoca retrasos y quejas de los clientes.
- Omisión de turnos, cuando la secretaria olvida anotar una cita solicitada por teléfono o de manera presencial.
- Dificultad para reprogramar, ya que no existe un registro histórico digital ni posibilidad de confirmar horarios de forma rápida.

En varias ocasiones, se han reportado situaciones en las que un cliente llega puntualmente y descubre que su cita no fue registrada. Esto no solo genera malestar inmediato, sino también pérdida de confianza en la clínica.

Adicionalmente, al no contar con recordatorios automáticos, muchos clientes simplemente olvidan sus citas y no se presentan. Estos "espacios vacíos" en la agenda representan tiempos muertos para el veterinario y pérdidas económicas, ya que se desaprovecha la capacidad instalada de atención.

En contraste, las clínicas modernas que ya implementaron sistemas digitales permiten a sus clientes reservar citas en línea, recibir recordatorios vía correo electrónico o WhatsApp y reprogramar fácilmente. Este tipo de funcionalidades no solo incrementa la eficiencia, sino que proyecta una imagen de profesionalismo y modernidad. La falta de un sistema de este tipo en coloca a la clínica en clara desventaja competitiva frente a instituciones que ya digitalizaron sus procesos.

1.4.2. Historias clínicas dispersas

El manejo de los historiales clínicos constituye otro de los puntos críticos de ineficiencia. Actualmente, la clínica archiva la información en carpetas físicas organizadas alfabéticamente. Si bien este método parece ordenado en teoría, en la práctica se convierte en un obstáculo constante:

- Buscar un historial médico entre cientos de carpetas puede tomar varios minutos, un tiempo valioso en situaciones de emergencia.
- Los documentos físicos están expuestos a daños por humedad, extravío, deterioro o confusión entre carpetas.
- La información no puede ser consultada por más de un médico a la vez, lo que dificulta la coordinación de casos.

Existen antecedentes de expedientes mal archivados que ocasionaron errores en tratamientos y diagnósticos, generando riesgos para la salud de los animales y comprometiendo la reputación de la clínica.

En clínicas digitalizadas, un solo clic permite acceder al historial completo del paciente: antecedentes, vacunas aplicadas, alergias, tratamientos previos, resultados de laboratorio y radiografías digitalizadas. Esta diferencia no solo se traduce en rapidez, sino también en seguridad médica y calidad del servicio.

En la clínica, la falta de digitalización en historiales clínicos representa un riesgo crítico que limita la capacidad de respuesta y aumenta las probabilidades de cometer errores médicos.

1.4.3. Control de inventario de medicamentos e insumos

El inventario de medicamentos y suministros médicos se lleva de forma manual en una hoja de cálculo, actualizada únicamente por el encargado de farmacia. Este método rudimentario provoca varios inconvenientes:

- No existe información en tiempo real sobre las existencias.
- No se generan alertas automáticas de stock mínimo o de vencimientos próximos.
- Se han producido desabastecimientos de fármacos esenciales como antibióticos, analgésicos o vacunas, lo cual compromete seriamente la atención de los pacientes.
- También se han registrado pérdidas económicas por medicamentos vencidos que no fueron rotados ni utilizados a tiempo.

La falta de control adecuado obliga al personal a invertir horas en conteos manuales frecuentes, lo que aumenta la carga administrativa y genera fricciones entre veterinarios y encargados de insumos cuando un producto solicitado no se encuentra disponible.

En sistemas de gestión modernos, el inventario se actualiza automáticamente con cada entrada o salida, emite alertas de stock mínimo, permite planificar compras con anticipación y genera reportes estadísticos de consumo. La ausencia de estas herramientas mantiene a la clínica en un estado de vulnerabilidad permanente en cuanto a insumos médicos.

1.4.4. Procesos administrativos y facturación

En la parte administrativa, la clínica todavía emite recibos en papel y almacena los registros en carpetas físicas. Este método acarrea múltiples desventajas:

- Mayor probabilidad de errores contables.
- Riesgo de duplicidad en los cobros o extravío de documentos.
- Necesidad de elaborar reportes financieros de forma manual, consumiendo largas horas de trabajo.

La ausencia de información consolidada en tiempo real dificulta que la gerencia tenga un panorama claro sobre el rendimiento económico. Esto impide la toma de decisiones estratégicas oportunas, como ajustes de precios, promociones o inversión en equipamiento.

Además, la clínica queda expuesta a problemas de cumplimiento tributario, ya que la elaboración manual de reportes aumenta la posibilidad de inconsistencias frente a auditorías.

En cambio, un sistema digital permitiría emitir facturas electrónicas, registrar automáticamente las cuentas por cobrar, consolidar ingresos y generar reportes instantáneos. Esto no solo ahorra tiempo y reduce errores, sino que incrementa la transparencia y confiabilidad administrativa.

1.4.5. Comunicación con los clientes

Actualmente, la relación con los dueños de las mascotas se limita a llamadas telefónicas para recordatorios de citas o vacunas. Este método, además de ser poco eficiente, depende completamente del tiempo disponible del personal administrativo.

En la práctica, muchas de estas llamadas no llegan a concretarse, se realizan de manera tardía o simplemente no se hacen. Como resultado:

- Los clientes olvidan acudir a sus citas.
- Se interrumpen tratamientos médicos por falta de continuidad.
- Se pierde la oportunidad de fidelizar al cliente mediante un servicio moderno y personalizado.

En la era digital, los clientes esperan recibir notificaciones automáticas, acceso en línea al historial de sus mascotas y la posibilidad de descargar facturas o comprobantes desde un portal web. La falta de estas opciones genera frustración y una percepción de servicio anticuado.

Esto representa un riesgo real de pérdida de clientes, quienes podrían migrar hacia clínicas más tecnológicas que les brinden la comodidad y eficiencia que demandan.

Consecuencias generales del problema:

- La combinación de todos estos factores crea un entorno de trabajo desorganizado e ineficiente, con impactos directos en:
- Eficiencia operativa: tiempos muertos, sobrecarga administrativa y duplicación de tareas.

- Calidad de servicio: errores en historiales, falta de medicamentos y atención lenta.
- Satisfacción del cliente: quejas frecuentes, pérdida de confianza y baja fidelización.
- Rentabilidad: pérdidas por citas fallidas, medicamentos vencidos y reportes manuales ineficientes.
- Competitividad: clara desventaja frente a clínicas que ya modernizaron su gestión.
- Necesidad de una solución

Ante este panorama, la implementación de un sistema de información integral no es una opción, sino una necesidad estratégica. Dicho sistema debe:

- Digitalizar y automatizar la gestión de citas, con recordatorios automáticos.
- Incorporar historiales clínicos electrónicos accesibles en tiempo real.
- Permitir el control de inventario con alertas de stock y vencimiento.
- Integrar facturación digital y generación inmediata de reportes financieros.
- Establecer canales digitales de comunicación con los clientes.

En conclusión, el problema central identificado es la ineficiencia estructural ocasionada por la ausencia de un sistema de información centralizado, lo que limita el crecimiento de la clínica, deteriora la calidad del servicio y compromete su competitividad en un mercado cada vez más exigente y tecnológico.

1.5.. - FORMULACIÓN DEL PROBLEMA

Antes del desarrollo del software, **ServetCorp** dependía completamente de procesos manuales. Las citas se anotaban en agendas, los historiales clínicos se archivaban en papel, los medicamentos se controlaban en hojas de cálculo y la facturación era escrita. Esta práctica generaba descoordinación interna, errores frecuentes, pérdida de información y limitaba la comunicación efectiva con los dueños de los animales.

1.6.- OBJETIVOS

1.6.1. - OBJETIVO GENERAL

Desarrollar un Sistema de Información para la Gestión de Procesos Administrativos, Clínicos y de Comunicación en Clínicas Veterinarias ‘VetCare’ .

1.6.2. - OBJETIVOS ESPECIFICOS

- Realizar el diseño conceptual de la base de datos utilizando UML para definir entidades, relaciones y reglas de negocio.
- Crear la base de datos física en PostgreSQL, asegurando la integridad y consistencia de los datos.
- Desarrollar el backend utilizando Python (Flask o Django) para implementar la lógica de negocio y la gestión de la información.
- Implementar el frontend con HTML, JavaScript y React, garantizando una interfaz amigable y funcional para los usuarios.
- Integrar el backend y frontend con la base de datos, asegurando la comunicación eficiente y segura entre todas las capas del sistema.
- Probar y validar todas las funcionalidades del sistema para asegurar su correcto funcionamiento, usabilidad y confiabilidad antes de su despliegue.

1.7.- ALCANCE

El sistema de gestión para la clínica veterinaria **ServetCorp** tiene como objetivo principal centralizar y automatizar los procesos internos de la institución, facilitando la gestión de citas, historiales clínicos, inventario, facturación y la comunicación con los clientes.

El sistema estará compuesto por los siguientes módulos y funciones principales:

Módulo de Gestión de Citas:

- Registrar, modificar y eliminar citas de manera rápida y confiable.
- Generar recordatorios automáticos para los clientes por correo electrónico o mensaje de texto.
- Evitar conflictos de horarios mediante control de disponibilidad del personal y recursos.

Módulo de Historias Clínicas:

- Digitalizar y almacenar todos los datos médicos de los pacientes.
- Permitir la consulta rápida y simultánea de información por parte del personal autorizado.
- Adjuntar resultados de laboratorio, imágenes y notas médicas de manera organizada.

Módulo de Inventario de Medicamentos e Insumos:

- Registrar entradas y salidas de medicamentos y materiales médicos.
- Controlar fechas de vencimiento y niveles mínimos de stock.

- Generar alertas automáticas de reabastecimiento y reportes de consumo.

Módulo de Facturación y Administración:

- Emitir facturas electrónicas y controlar pagos realizados.
- Generar reportes financieros periódicos para la toma de decisiones.
- Registrar de manera confiable todas las transacciones económicas de la clínica.

Módulo de Comunicación con Clientes:

- Enviar notificaciones de citas, vacunas y tratamientos pendientes.
- Permitir la consulta de información relevante de las mascotas vía portal web o aplicación.
- Facilitar la interacción directa entre la clínica y los propietarios de los animales.

El alcance del sistema se limita a la gestión interna de la clínica y la comunicación con los clientes, sin incluir funciones externas como telemedicina avanzada, pagos en línea ni integración con otras plataformas externas, aunque puede considerarse su incorporación en futuras versiones.

ENTREVISTAS PARA OBTENCIÓN DE REQUERIMIENTOS

Objetivo de la entrevista: Recopilar información sobre el funcionamiento actual de la clínica, identificar los problemas de la falta de un sistema de información y definir los requisitos clave para el diseño y desarrollo de un sitio web funcional.

Entrevista #1

Fecha	23/08/2025
Duración	20 min
Lugar	Clinica Veterinaria ServetCorp
Dirección	5to Anillo - Radial 13. C/ San Antonio

Datos de la empresa	
Nombre	ServetCorp
Tipo	Privada
Datos del entrevistado	
Nombre	Carlos Guzman
Cargo	Repcionista
Datos del entrevistador	
Nombre	Diego Melgar Gushi

Aspectos por abordar

a) Información General y Operativa de la Clínica

a.1. ¿Cuántos veterinarios y personal de apoyo trabajan en la clínica y cuáles son sus roles principales?

R: Un equipo de 6 personas. Está el Dr. Gutiérrez, que es el veterinario jefe y cirujano principal. Lo apoya la Dra. Sofía Rojas en consultas generales. Dos auxiliares veterinarios: Ana, que es la principal y asiste en el quirófano, y Luis, que se encarga del área de internación. Luego está María, nuestra peluquera de tiempo completo, y finalmente Carlos, encargado de toda la recepción y administración.

a.2. ¿Cuál es el horario de atención y cómo manejan las emergencias fuera de ese horario?

R: Atendemos de lunes a sábado de 9 am a 7 pm. Las emergencias se manejan de forma informal a través del celular personal del doctor, no hay un sistema formal.

a.3. ¿Qué gama de servicios ofrecen actualmente?

R: Consultas, vacunación, cirugías menores, peluquería y venta de algunos productos. Derivamos los análisis complejos y ecografías a otros centros.

a.4. ¿Cuáles son los días o las horas de mayor afluencia de clientes y cómo gestionan esos picos de trabajo?

R: Los sábados por la mañana y los días de semana después de las 5 pm. Gestionamos el flujo como podemos, pero el sistema manual nos genera desorden y demoras.

b) Procesos Clave y Flujo de Trabajo Actual (Sin Sistema)

b.1. Describa el paso a paso de cómo un cliente agenda una cita actualmente.

R: Por llamada telefónica, WhatsApp o presencial. Todas las citas se anotan manualmente en un único cuaderno.

b.2. ¿Cuál es el procedimiento desde que un paciente ingresa a la clínica hasta que se retira?

R: Se registra al cliente (creando una ficha de cartulina si es nuevo o buscando la existente), espera su turno, pasa a consulta, y al final se realiza el cobro en recepción.

b.3. ¿Cómo se crea, archiva y consulta el historial clínico de una mascota?

R: Se usan fichas de cartulina que se archivan en gavetas por orden alfabético. La consulta es manual y a veces lenta por errores de archivo.

b.4. ¿Cómo gestionan el inventario de medicamentos, vacunas y alimentos?

R: Es un control visual. Cuando vemos que algo se está acabando, lo anotamos en una pizarra para hacer el pedido. No es preciso y a veces nos quedamos sin stock.

c) Gestión de Clientes y Comunicación

c.1. ¿Qué canales utilizan para comunicarse con los clientes para recordatorios o seguimientos?

R: No tenemos un sistema formal. Ocasionalmente, si hay tiempo, hacemos una llamada o enviamos un WhatsApp. El seguimiento post-operatorio lo hace el doctor directamente.

c.2. ¿Implementan alguna estrategia para la fidelización de clientes?

R: No, ninguna estrategia formal. La fidelización se basa únicamente en la confianza y el buen trato del doctor.

c.3. Aparte de los datos para la factura, ¿qué información del cliente y de la mascota consideran esencial registrar?

R: Del dueño: nombre y celular. De la mascota: nombre, especie, raza, sexo, edad y, muy importante, alergias o condiciones preexistentes.

c.4. ¿Cuáles son las preguntas más frecuentes que reciben que podrían resolverse con un sitio web?

R: Horarios, costo de la consulta, si ofrecemos un servicio específico y la ubicación de la clínica.

d) Gestión de Servicios y Pagos

d.1. ¿Qué métodos de pago aceptan actualmente?

R: Efectivo, QR y transferencia bancaria. No aceptamos tarjetas.

d.2. ¿Cómo es el proceso de facturación?

R: Se emite factura fiscal a través del sistema de Impuestos. El detalle de los servicios se explica verbalmente.

d.3. ¿Ofrecen paquetes de servicios?

R: Sí, un plan de vacunación para cachorros. Se controla con una tarjeta sellada, pero no lo promocionamos activamente.

e) Manejo de Información y Desafíos Actuales

e.1. ¿Cuál es el principal desafío o "cuello de botella" en la gestión diaria de la clínica?

R: La recepción: la gestión simultánea de llamadas, citas y búsqueda de historiales manuales crea el mayor caos y retraso.

e.2. ¿Cómo garantizan la confidencialidad de los historiales en su sistema de archivo físico?

R: El archivador está en un área restringida al personal. Es una seguridad básica.

e.3. ¿Han tenido problemas específicos por no tener un sistema digital?

R: Sí: citas duplicadas, demoras críticas en encontrar historiales de emergencia y olvido en el seguimiento de refuerzos de vacunas, es decir los clientes muchas veces pierden la cartilla de seguimiento de su mascota.

f) Necesidades y Expectativas del Nuevo Sistema Web

f.1. Si pudieran solucionar un solo problema con esta nueva web, ¿cuál sería el más importante?

R: El sistema de agendamiento de citas online.

f.2. De una lista de posibles funciones, ¿cuáles consideran indispensables y cuáles serían "un lujo"?

R: Indispensables: Citas online, historiales clínicos digitales. Un lujo: Pagos en línea o un blog de contenidos y recordatorios automáticos.

f.3. ¿Tienen alguna preocupación sobre la implementación de un sistema web?

R: Sí, dos: la adaptación de los clientes mayores al sistema online y el tiempo que nos tomará digitalizar todos los historiales físicos existentes.

2. ELEMENTOS DEL SISTEMA BASADO EN COMPUTADORAS

2.1. Hardware

2.1.1. Servidor

Nombre	Versión	Característica
Procesador	Intel Core i7-12400F	4.1GHz
Memoria RAM		16GB
Disco Duro	Kingston	1TB
Gráfica		

2.1.2. Cliente

Nombre	Versión	Característica
Procesador	Intel i3 12th o superior	-
Memoria RAM	-	8gb
Sistema Operativo	Windows 10 o superior	

2.1.3. Medios de comunicación

El medio de comunicación que se utilizará para transferir la información será a través de red internet-wifi.

2.1.4. Otros dispositivos

Nombre	Características
Impresoras o escáneres	Si el sistema necesita imprimir facturas, recibos o documentos administrativos.
Cámaras de seguridad	Para monitorear las instalaciones, verificación de usuarios o control de acceso.
UPS (Sistema de alimentación ininterrumpida)	Para garantizar que los servidores y computadoras funcionen incluso ante cortes de energía.
Tablets o dispositivos móviles	Para el sistema web que se desarrollará en este proyecto.

2.2.Software

2.2.1.1 Cliente

NOMBRE	VERSIÓN
Sistema Operativo	Windows 11.
Navegador	Google Chrome, Firefox y otros.
Documentación del proyecto	Office

2.2.1.2 Otro software adicional

NOMBRE	CARACTERÍSTICAS
Visual Studio Code	Herramientas de Desarrollo.
Sistemas Operativos	Windows Server o cualquier IDE necesario para la programación.
Firewalls, antivirus o software para proteger el sistema.	Software de seguridad.
Google Cloud o Azure	Servicios en la nube.
JasperReports, Crystal Reports	Software de reportes.

2.3.Datos

Módulo involucrado	Entidad	Dato	Descripción
Mod. Gestión de Citas	Tutor (Cliente)	Nombre	Nombre completo del propietario.
		C.I./NIT	Identificación personal o fiscal.
		Celular / Email	Contactos para confirmación y recordatorios.
	Paciente	Nombre	Nombre de la mascota.
		Especie / Raza / Sexo	Clasificación del paciente.
		Fecha de nacimiento	Cálculo de edad / plan preventivo.
		Microchip	Identificador único (si existe).

	Cita	Fecha/Hora (inicio–fin)	Franja reservada para atención.
		Motivo	Razón de la consulta.
		Estado	pendiente, confirmada, reprogramada, cancelada, atendida, no_show}.
	Recurso	Sala / Equipo	Consultorio o equipo asignado.
	Notificación	Canal	Email / SMS / WhatsApp habilitado.
		Fecha envío / Resultado	Seguimiento del recordatorio.
Mod. Historias Clínicas	Historia	Fecha de apertura	Inicio del expediente clínico.
		Resumen	Antecedentes relevantes.
	Consulta	Motivo / Anamnesis	Información clínica declarada.
		Examen físico	Hallazgos objetivos.
		Diagnóstico / Plan	Conclusiones y tratamiento.
		Adjuntos	Imágenes / PDFs / resultados vinculados.
	Receta	Indicaciones	Medicamentos, dosis, frecuencia, duración.
		Firma digital	Huella/hash de integridad y autoría.
	Vacuna/Tratamiento	Producto	Biológico/medicamento aplicado.
		Lote / Vencimiento	Trazabilidad sanitaria.
		Dosis / Vía / Fecha	Detalle de administración.
		Próxima fecha	Programación de refuerzo/seguimiento.
Mod. Inventario de Medicamentos e Insumos	Producto	Tipo / Unidad	{medicamento, insumo, alimento} y unidad de medida.
		Exige lote	Control obligatorio de lote/vencimiento.
		Código de barras	Identificación operativa.
	Lote	Nº de lote	Identificador del lote recibido.
		Vencimiento	Fecha límite de uso.
	Kardex	Tipo de movimiento	{ingreso, egreso, ajuste}.
		Cantidad / Costo	Movimiento y valorización.

		Motivo / Usuario / Fecha	Trazabilidad del cambio.
	Alerta	Stock mínimo	Umbral para reabastecer.
		Próximo vencimiento	Fecha para acciones preventivas.
Mod. Facturación y Administración	Venta	Fecha / Total	Momento y monto de la transacción.
		Método de pago	{efectivo, QR, transferencia}.
		Correlativo	Numeración fiscal/administrativa.
	Ventaltem	Producto / Lote	Ítem vendido y lote (si aplica).
		Cantidad / Precio / Impuesto	Detalle por ítem.
	Caja	Apertura / Cierre	Control diario de caja.
	Reporte	Período	Rango de fechas del informe.
		Métricas clave	Ingresos, margen, rotación.
	Preferencia	Opt-in	Consentimiento del tutor para envíos.
	Plantilla	Tipo / Contenido	Formatos (cita, vacuna, seguimiento).
	Canal	SMTP / WhatsApp	Medio de notificación.
	Mensaje	Estado	Enviado / Entregado / Error.
	Portal	Último acceso	Fecha/hora y dispositivo usado.

2.4. Procesos

Módulo	Proceso	Entidades involucradas	Descripción
Mod. Gestión de Citas	Registrar / Modificar / Eliminar cita	Tutor, Paciente, Cita, Recurso	Crear, actualizar o cancelar citas controlando disponibilidad por profesional y sala/equipo para evitar conflictos de horario.
	Control de disponibilidad en línea.	Cita, Recurso, Usuario (profesional)	Validar solapamientos y horarios; bloquear reservas fuera de agenda o en recursos ocupados.
Mod. Historias Clínicas	Digitalizar y almacenar expediente	Historia, Paciente, Tutor	Crear y mantener la historia clínica única por paciente; guardar antecedentes y actualizaciones.
	Consulta clínica	Consulta, Historia, Paciente, Adjunto	Registrar motivo, análisis, examen, diagnóstico y plan; adjuntar resultados/archivos; cerrar episodio.
	Gestión de adjuntos	Adjunto, Consulta/Historia	Cargar, validar y vincular imágenes, laboratorios y notas (PDF/imagen).
Mod. Inventario de Medicamentos e Insumos	Registro de entradas	Proveedor, Compra, Producto, Lote, Kardex	Ingresar compras/ajustes con lotes y vencimientos; actualizar stock y costo.
	Registro de salidas	Producto, Lote, Kardex, Consulta/Venta	Descontar stock por consumo clínico o venta (PEPS por defecto) con trazabilidad por lote.
	Control de vencimientos y mínimos	Producto, Lote.	Monitorear mínimos y fechas próximas a vencer.
	Reportes de consumo	Producto, Kardex, Reporte	Emitir reportes de consumo/rotación para reabastecimiento.
Mod. Facturación y Administración	Emisión de facturas y cobros	Venta, Ventaltem, Caja, Tutor	Armar venta (servicios/productos), cobrar (efectivo/QR/transferencia) y emitir factura; reflejar en caja.
	Reportes financieros	Venta, Compra, Caja, Reporte	Generar informes periódicos (ingresos, margen, rotación, CxC) para decisiones.
	Registro de transacciones	Venta, Caja, Bitácora	Guardar toda operación económica con correlativo y auditoría.

Mod. Comunicación con Clientes	Vacunas	Cita, Evento.	Registrar entregas y reintentos.
	Portal web/app	Portal, Tutor, Paciente	Permitir consulta de citas, vacunas y tratamientos; descarga de comprobantes; interacción básica con la clínica.
	Interacción directa	Mensaje, Tutor, Usuario	Comunicación puntual (confirmaciones, aclaraciones) mediante canales habilitados.

2.5.Gente/Usuario

El equipo está conformado por:

INTEGRANTES
Ortiz Montero Luis Enrique
Auad Castillo Miguel Andrés
Melgar Gushi Diego
Llanos Vinaya Rodrigo
Crespo Cárdenas Gabriela Belén
Cervantes Arancibia Roberto Carlos

2.6.Documento

Se utilizarán modelos UML dentro del Proceso Unificado de Desarrollo de Software (PUDS) para visualizar, validar y comunicar el sistema por módulos: Gestión de Citas, Historias Clínicas, Inventario, Facturación/Administración y Comunicación con Clientes.

3. TECNOLOGÍA PARA EL DESARROLLO DEL SOFTWARE

3.1. ESTRATEGIA PARA EL DESARROLLO DEL SOFTWARE

Para realizar el sistema se utilizaron 2 estrategias para el desarrollo del software: El Lenguaje Unificado de Modelado (UML) y El Proceso Unificado de Desarrollo de Software (PUDS).

Se utilizará UML ya que se requiere un lenguaje de modelado para especificar o describir modelos o procesos de nuestros sistemas, además se necesita de un lenguaje de modelado unificado (UML) para los requerimientos del sistema a través de diagramas.

Asimismo, se utilizará el proceso iterativo PUDS (Proceso Unificado de Desarrollo de Software), ya que permite desarrollar software a gran escala al tener pruebas constantes y obtener respuestas con base a dichas pruebas para garantizar la calidad de nuestros resultados.

3.2. METODOLOGÍA PARA EL DESARROLLO DEL SOFTWARE

3.2.1. CARACTERÍSTICAS DEL PUDS

El Proceso Unificado de Desarrollo de Software es un proceso iterativo, un enfoque iterativo propone una comprensión incremental del problema a través de refinamientos sucesivos y un crecimiento incremental de una solución efectiva a través de varios ciclos.

Este proceso puede organizarse en cuatro fases: inicio, elaboración, transición y construcción. Las fases se dividen en sus conjuntos de iteraciones, en las que se desarrollan los flujos de trabajo: requerimientos, análisis, diseño, implementación y pruebas.

El proceso unificado está dirigido por casos de uso

Para desarrollar un sistema de manera efectiva, es crucial entender lo que los usuarios potenciales buscan y requieren. La palabra usuario se refiere no solo a personas, sino también a otros sistemas.

Así, el término usuario implica cualquier entidad que se relaciona con el sistema que estamos creando.

Un caso de uso es una parte de la funcionalidad del sistema que ofrece al usuario un resultado significativo. Los casos de uso son indicativos de las necesidades funcionales. La totalidad de los casos de uso forma el modelo de casos de uso, que ilustra la funcionalidad completa del sistema.

El proceso unificado está centrado en la arquitectura

El Proceso Unificado asume que no existe un único modelo que abarque todas las dimensiones del sistema. Por esta razón, se emplean diversos modelos y perspectivas que delinean la arquitectura del software de un sistema.

Enfocado en los riesgos

El Proceso Unificado demanda que el grupo de trabajo del proyecto dirija su atención hacia la identificación de los riesgos significativos en las fases iniciales del ciclo de vida. Los productos generados en cada iteración, especialmente durante la etapa de Elaboración, deben ser elegidos en una secuencia que garantice la priorización de los riesgos más relevantes.

- Se basa en los siguientes principios de desarrollo:
- Desarrollo Iterativo del Software
- Administración de Requerimientos
- Uso de Arquitecturas Basadas en Componentes
- Modelado Visual del Software
- Verificación de la Calidad del Software

- Control de Cambios.

Por otra parte, la arquitectura debe ser flexible, fácil de modificar, intuitivamente comprensible y de fácil reutilización de componentes.

Flujos de Trabajo

Captura de requisitos: Es el proceso de determinar, generalmente en situaciones complicadas, lo que se debe desarrollar.

Análisis: Analizamos los requisitos que se detallaron en el documento de captura de requisitos, llevándolos a un proceso de refinamiento y organización. La finalidad de esta actividad es alcanzar una comprensión más exacta de los requerimientos y elaborar una descripción que sea sencilla de gestionar, contribuyendo así a la organización del sistema en su totalidad.

Diseño: En la fase de diseño, creamos un modelo del sistema y determinamos su estructura para que cumpla con todas las exigencias, abarcando tanto los requisitos no funcionales como otras limitaciones inherentes.

Implementación: La finalidad primordial de la ejecución es crear la estructura y el sistema en su totalidad.

Prueba: Se lleva a cabo una revisión del resultado de la implementación evaluando cada una de las estructuras, abarcando tanto las construcciones internas como las intermedias, además de las versiones finales del sistema que se entregarán a terceros.

Fases del Proceso

Fase de Inicio: Se elabora una definición del producto final basada en una especificación adecuada. Se reconocen y clasifican los riesgos más significativos, se especifica la etapa de producción, y se realiza una estimación preliminar del proyecto.

Fase de Elaboración: Se describe con precisión la mayor parte de los escenarios de utilización del producto y se elabora la estructura del sistema. Se crean los escenarios de uso más esenciales que fueron detectados en la etapa previa.

Fase de Construcción: El producto se desarrolla de acuerdo con los casos de uso que han sido consensuados por la dirección y el cliente para esta versión. Aunque esta versión puede no estar totalmente finalizada, los errores menores que se encuentren serán solucionados en la próxima etapa.

Fase de Transición: En esta fase el producto se transforma en su versión beta. Implica tareas como la producción, capacitación del cliente, ofrecer servicio de soporte y asistencia, así como la rectificación de los errores detectados.

3.2.2. CARACTERÍSTICAS DE UML

El Lenguaje Unificado de Modelado (UML), es un lenguaje visual de modelado que se utiliza con el fin de definir, ilustrar, construir y documentar los elementos de un sistema de software. Registra decisiones y conocimientos sobre los sistemas que necesitan ser desarrollados. Su propósito es facilitar la comprensión, el diseño, la revisión, la configuración, el mantenimiento y la gestión de la información relacionada con dichos sistemas. Se elabora para aplicarse en todos los métodos de desarrollo, fases del ciclo de vida, áreas de aplicación y modalidades.

UML es más que una notación visual. Los modelos UML tienen la capacidad de generar código y pruebas. Esto requiere un perfil UML apropiado, la utilización de herramientas compatibles con la plataforma de destino y la selección entre diversos elementos de implementación.

UML debe ser lo bastante claro y efectivo para abordar todos los conceptos presentes en un sistema contemporáneo, tales como la concurrencia y la distribución.

El Lenguaje Unificado de Modelado ofrece un conjunto de diagramas, siendo los dos más destacados:

Estructurales: Diagramas de clases, componentes, despliegue, objetos, paquetes, perfiles y estructura compuesta.

De comportamiento: Diagrama de actividades, casos de uso, máquina de estados e interacción.

3.3.HERRAMIENTAS DE DESARROLLO

3.3.1. SOFTWARE

Para el desarrollo de este proyecto se requieren los siguientes recursos informáticos:

NOMBRE	VERSIÓN
Sistema Operativo	Windows 11
Gestor de Base de Datos	PostgreSQL
Frontend	React, Tailwind, CSS
Backend	Python, JavaScript, Flask
Plataforma de Modelado Visual	UML

Repositorio Remoto	GitHub
Editor	Visual Studio Code
Servidor en la Nube	Railway

3.3.2. HARDWARE

Ordenador 1: Desktop Personalizada

Nombre	Versión	Característica
Procesador	Intel Core i5-9400F	4.1GHz
Memoria RAM	HyperX	16GB
Disco Duro	Crucial	1TB
Gráfica	Nvidia GT1030	4GB

Ordenador 2: Lenovo Thinkpad

Nombre	Versión	Característica
Procesador	AMD Ryzen 7 Pro 8840HS	5.1 GHz
Memoria RAM	HyperX	32GB
Disco Duro	Kingston	1TB NVMe
Gráfica	Radeon 780M Graphics	3.30 GHz

Ordenador 3: Desktop Personalizada

Nombre	Versión	Característica
Procesador	Intel Core i5-9400F	2.90 GHz
Memoria RAM	HyperX	16 GB
Disco Duro	Western Digital	1 TB
Gráfica	Nvidia GT1030	2GB

Ordenador 4: Desktop Personalizada

Nombre	Versión	Característica
Procesador	AMD Ryzen 5 3500X	3.59 GHz
Memoria RAM	HyperX	16 GB
Disco Duro	Kingston	1 TB
Gráfica	Nvidia GTX 1650	4 GB

Ordenador 5: Desktop Personalizada

Nombre	Versión	Característica
Procesador	Intel Core i5-12400	2.50 GHz
Memoria RAM	Corsair	16 GB
Disco Duro	WD Black	500 GB

Gráfica	Nvidia RTX 4060 Ti	8 GB
---------	--------------------	------

Ordenador 6: HP Laptop

Nombre	Versión	Característica
Procesador	Intel Core i7-1165G7	2.80GHz
Memoria RAM	Crucial	16 GB
Disco Duro	Western Digital	500 GB
Gráfica	Intel Iris Xe	-

4. POSIBLES COSTOS

4.1.COSTO DE HERRAMIENTAS DE DESARROLLADORES

Herramienta de Trabajo	Costo (USD)
Desktop Personalizada	600
Lenovo Thinkpad	900
Desktop Personalizada	550
Desktop Personalizada	800
Desktop Personalizada	1100
HP Laptop	750

Total	4700
--------------	------

4.2.COSTO DE DESARROLLO

ÍTEM	COSTO (Bs)
Conexión a Internet (6 conexiones, Bs 200/Mes)	2 400
Horas de Trabajo (Bs 20 hora/Desarrollador)¹	15 000
Material de Escritorio	250
Depreciación de Equipos de Computación del Equipo de Desarrollo²	690
Total	18.340

5 hrs*25 días*6 personas = 750 horas de trabajo.

considerando un costo promedio de Bs 5500 para las 6 computadoras del equipo, la depreciación mensual de cada uno de los equipos es Bs 5500/4 años/12 meses = Bs 115/mes.

4.3.COSTO DE IMPLEMENTACIÓN

ÍTEM	COSTO ANUAL (Bs)
Dominio Web	405
Servidor en la Nube	504
Total	909

5. POSIBLES BENEFICIOS

La implementación del sistema que permita digitalizar los procesos administrativos y clínicos para la veterinaria, genera los siguientes posibles beneficios:

5.1. TIEMPO

Agendamiento rápido de citas, permitiendo registrar, modificar y consultar citas en segundos, evitando la búsqueda manual en agendas físicas.

Procesos automatizados, la generación de reportes se hace de manera automática, reduciendo horas de trabajo administrativo.

Rápido acceso a historiales, posibilidad de consultar historiales clínicos y de vacunación de una mascota con solo ingresar el nombre.

5.2.ESFUERZO

Menor carga administrativa, evitando el llenado repetitivo de formularios en papel.

Organización centralizada, todos los datos se encuentran en un mismo sistema, reduciendo el esfuerzo de búsqueda y control.

Alertas automáticas, recordatorios de citas, avisos de stock bajo en inventario eliminan la necesidad de revisiones manuales continuas.

5.3.COSTOS

Menor gasto en insumos físicos, reduciendo el uso de papel, archivadores y espacio de almacenamiento, disminuyendo los costos de material de oficina.

Prevención de errores costosos, la digitalización disminuye la probabilidad de citas duplicadas, pérdida de históricos o errores en el inventario que puedan generar gastos adicionales.

Optimización del recurso humano, el personal dedica menos horas a tareas repetitivas, enfocándose en la atención al cliente y los servicios médicos, incrementando la productividad.

6. POSIBLES CLIENTES

Este sistema de información está diseñado para cubrir todas las necesidades administrativas de una clínica veterinaria, entre las cuales tenemos:

CLIENTE	DIRECCIÓN
Clínica Veterinaria ServetCorp	5to Anillo, Av. Radial 13 c/ San Antonio.
Clinica Veterinaria Mi Huesito	4to Anillo, c/ Penoco.
Clínica Veterinaria Amor Animal	2do Anillo, c/ Pisagua.

7. Marco teórico

CAPITULO 1: METODO DE ISHIKAWA

1.-Identificar Problemas

1.1.- Lista de problemas

P1. Ineficiencia en gestión de citas manuales

P2. Ineficiencia en la comunicación con los clientes.

P3. Sobrecarga laboral al personal de recepción debido a llamadas.

P4. Demora en la búsqueda de historias clínicas y antecedentes.

P5. Demora y errores en el registro de tratamientos, vacunas y desparasitaciones.

P6. Gestión ineficiente y riesgosa de adjuntos clínicos.

P7. Riesgo de pérdida/daño de registros físicos (fichas y consentimientos).

P8. Demora en el registro de datos del tutor y de la mascota (paciente).

P9. Gestión de inventario ineficiente (ingreso por lote, control de vencimientos).

P10. Gastos innecesarios en insumos físicos.

P11. Ineficiencia en el control de gastos.

P12. No hay un canal de comunicación formal para emergencias.

P13. Capacitación del personal.

P14. Mala ubicación de la clínica en un mercado competitivo.

P15. Mobiliario y equipo obsoleto.

1.2.- Depurar Problemas

P1. Ineficiencia en gestión de citas manuales

P2. Ineficiencia en la comunicación con los clientes.

P3. Sobrecarga laboral al personal de recepción debido a llamadas.

P4. Demora en la búsqueda de historias clínicas y antecedentes.

P5. ~~Demora y errores en el registro de tratamientos, vacunas y desparasitaciones.~~

P6. Gestión ineficiente y riesgosa de adjuntos clínicos.

P7. Riesgo de pérdida/daño de registros físicos (fichas y consentimientos).

P8. Demora en el registro de datos del tutor y de la mascota (paciente).

P9. Gestión de inventario ineficiente (ingreso por lote, control de vencimientos).

P10. ~~Gastos innecesarios en insumos físicos.~~

P11. Ineficiencia en el control de gastos.

P12. No hay un canal de comunicación formal para emergencias.

P13. ~~Capacitación del personal.~~

P14. ~~Mala ubicación de la clínica en un mercado competitivo.~~

P15. ~~Mobiliario y equipo obsoleto.~~

1.3.- Lista Final de Problemas

P1. Ineficiencia en gestión de citas manuales

P2. Ineficiencia en la comunicación con los clientes.

P3. Sobrecarga laboral al personal de recepción debido a llamadas.

P4. Demora en la búsqueda de historias clínicas y antecedentes.

P5. Riesgo de pérdida/daño de registros físicos (fichas y consentimientos).

P6. Demora en el registro de datos del tutor y de la mascota (paciente).

P7. Gestión de inventario ineficiente (ingreso por lote, control de vencimientos).

P8. Ineficiencia en el control de gastos.

P9. No hay un canal de comunicación formal para emergencias.

1.4.- Propietarios De Problemas

▪ **Dueño/Gerente de la Clínica:** Es responsable de la toma de decisiones estratégicas de la clínica. Su rol es asegurar que el sistema se alinee con los objetivos de negocio, optimice los recursos y mejore la calidad del servicio. Es el principal beneficiario de las mejoras en la eficiencia operativa.

▪ **Médico Veterinario:** Encargado de la atención médica de los pacientes. Su rol es crítico para garantizar un acceso rápido y preciso a los historiales clínicos, resultados de laboratorio y otros datos para un diagnóstico y tratamientos efectivos.

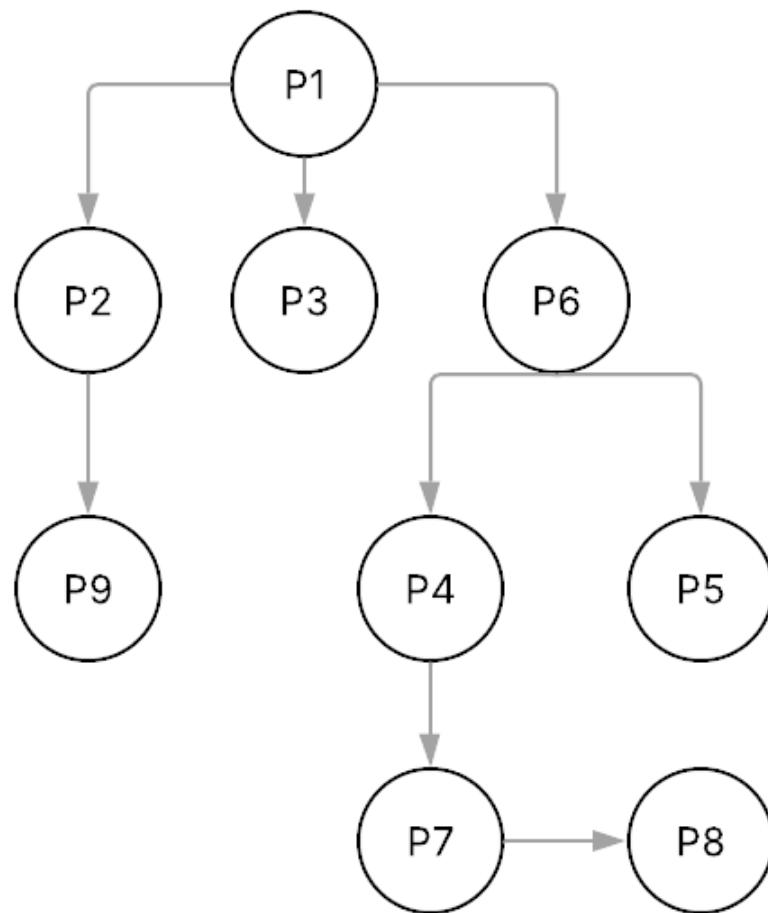
▪ **Personal de Recepción:** Responsable de la gestión de citas y la atención inicial al cliente. Su rol es esencial en la operación diaria, ya que un sistema de información debe reducir su carga administrativa y mejorar la comunicación con los clientes.

▪ **Tutor/Cliente:** Es el dueño de la mascota. Aunque no es parte del personal de la clínica, es un propietario del problema porque la ineficiencia en la gestión de citas, la comunicación y el acceso a la información afecta su experiencia. El software debe solucionar estos puntos para mejorar su satisfacción.

▪ **Personal Administrativo (Contabilidad/Almacén):** Se encarga de la gestión de inventario y los registros financieros. Su rol es crucial para asegurar que el sistema automatice el control de stock y facilite la generación de reportes contables, minimizando errores y pérdidas.

Problemas	Dueño	Recepcionista	Veterinario	Cliente	Bodeguero
P1: Ineficiencia en gestión de citas manuales	✓	✓	✓	✓	
P2: Ineficiencia en la comunicación con los clientes	✓	✓		✓	
P3: Sobrecarga laboral al personal de recepción debido a llamadas	✓	✓		✓	
P4. Demora en la búsqueda de historias clínicas y antecedentes.		✓	✓	✓	
P5. Riesgo de pérdida/daño de registros físicos (fichas y consentimientos).	✓	✓	✓	✓	
P6. Demora en el registro de datos del tutor y de la mascota (paciente).	✓	✓	✓		
P7. Gestión de inventario ineficiente (ingreso por lote, control de vencimientos).	✓		✓		✓
P8. Ineficiencia en el control de gastos.	✓				✓
P9. No hay un canal de comunicación formal para emergencias.	✓		✓	✓	

1.5.- Análisis de problemas



- **P1:** Ineficiencia en gestión de citas manuales
- **P2:** Ineficiencia en la comunicación con los clientes.
- **P3:** Sobrecarga laboral al personal de recepción debido a llamadas.
- **P4:** Demora en la búsqueda de historias clínicas y antecedentes.
- **P5:** Riesgo de pérdida/daño de registros físicos (fichas y consentimientos).
- **P6:** Demora en el registro de datos del tutor y de la mascota (paciente).

- **P7:** Gestión de inventario ineficiente (ingreso por lote, control de vencimientos).
- **P8:** Ineficiencia en el control de gastos.
- **P9:** No hay un canal de comunicación formal para emergencias.

1.6.- Estimación y cuantificación del problema

Problema	Justificación (Resumen Unificado)
P1: Ineficiencia en gestión de citas manuales	La digitalización de la agenda mejora la organización y la experiencia del cliente. Al reducir las citas duplicadas, aumenta la eficiencia operativa y los ingresos.
P2: Ineficiencia en la comunicación con los clientes	La automatización de notificaciones y recordatorios incrementa la satisfacción y fidelidad del cliente, a la vez que reduce la carga laboral del personal y optimiza los recursos.
P3: Sobrecarga laboral al personal de recepción debido a llamadas	Al automatizar tareas, se reduce el tiempo dedicado a la gestión manual y se permite que el personal se enfoque en una atención de mayor calidad, lo que mejora la productividad y el ambiente de trabajo.
P4: Demora en la búsqueda de historias clínicas y antecedentes	El acceso inmediato a los historiales clínicos no solo mejora la calidad del diagnóstico, sino que también reduce el tiempo de consulta, permitiendo atender a más pacientes y mejorando el flujo de trabajo.
P5: Riesgo de pérdida/daño de registros físicos	La digitalización elimina el riesgo de perder información crucial por daños o extravío, asegurando la confiabilidad de los datos y eliminando los costos y el tiempo de recrear la información.

P6: Demora en el registro de datos del tutor y de la mascota (paciente)	Agilizar el proceso de admisión mejora la primera impresión del cliente y reduce su tiempo de espera. Esto se traduce en una mayor eficiencia operativa y una mejor experiencia para el usuario.
P7: Gestión de inventario ineficiente	Un control de inventario automatizado previene pérdidas económicas por productos vencidos y asegura la disponibilidad de insumos clave. Esto mejora la organización y optimiza los recursos de la clínica.
P8: Ineficiencia en el control de gastos	Un sistema digital permite un seguimiento detallado de las transacciones, lo que facilita la toma de decisiones y la identificación de áreas de ahorro, lo que se traduce en una mejor salud financiera para la clínica.
P9: No hay un canal de comunicación formal para emergencias	Formalizar este proceso a través del software asegura un registro y seguimiento adecuados. Esto mejora la calidad de la atención en situaciones críticas y protege a la clínica de posibles errores costosos.

1.7.- Alternativas de Cambio

Alt1. Implementar un módulo de gestión de citas que permita registrar, modificar y eliminar turnos de manera digital. Este sistema permitirá a la clínica organizar su agenda de forma centralizada y eficiente.

Alt2. Desarrollar un módulo de gestión de comunicación interna que permita al personal registrar las acciones de contacto con los clientes (por ejemplo, llamadas, mensajes de seguimiento o correos) y programarlas.

Alt3. Integrar un sistema de gestión de tareas que automatice la asignación y el seguimiento de las actividades de recepción. Esto reducirá la carga administrativa del personal, lo que le permitirá enfocarse en tareas más estratégicas.

Alt4. Digitalizar las historias clínicas y antecedentes de los pacientes, permitiendo que el personal autorizado acceda a toda la información de forma rápida y simultánea.

Alt5. Implementar un sistema de almacenamiento digital para todos los registros y adjuntos clínicos. Esto protegerá los documentos de daños o pérdida, y permitirá consultas más eficientes.

Alt6. Automatizar el registro de datos de tutores y mascotas para reducir el tiempo de espera en la recepción y eliminar la necesidad de fichas en papel.

Alt7. Utilizar un módulo de inventario digital que controle las entradas y salidas, monitoree los niveles de stock y registre las fechas de vencimiento y reabastecimiento.

Alt8. Crear un sistema de facturación y administración que permita emitir recibos electrónicos y generar reportes financieros periódicos de manera automática para un mejor control de gastos.

Alt9. Establecer un módulo de comunicación formal para emergencias, el cual registre y formalice la atención fuera del horario habitual.

1.8.- Conclusión y Recomendación

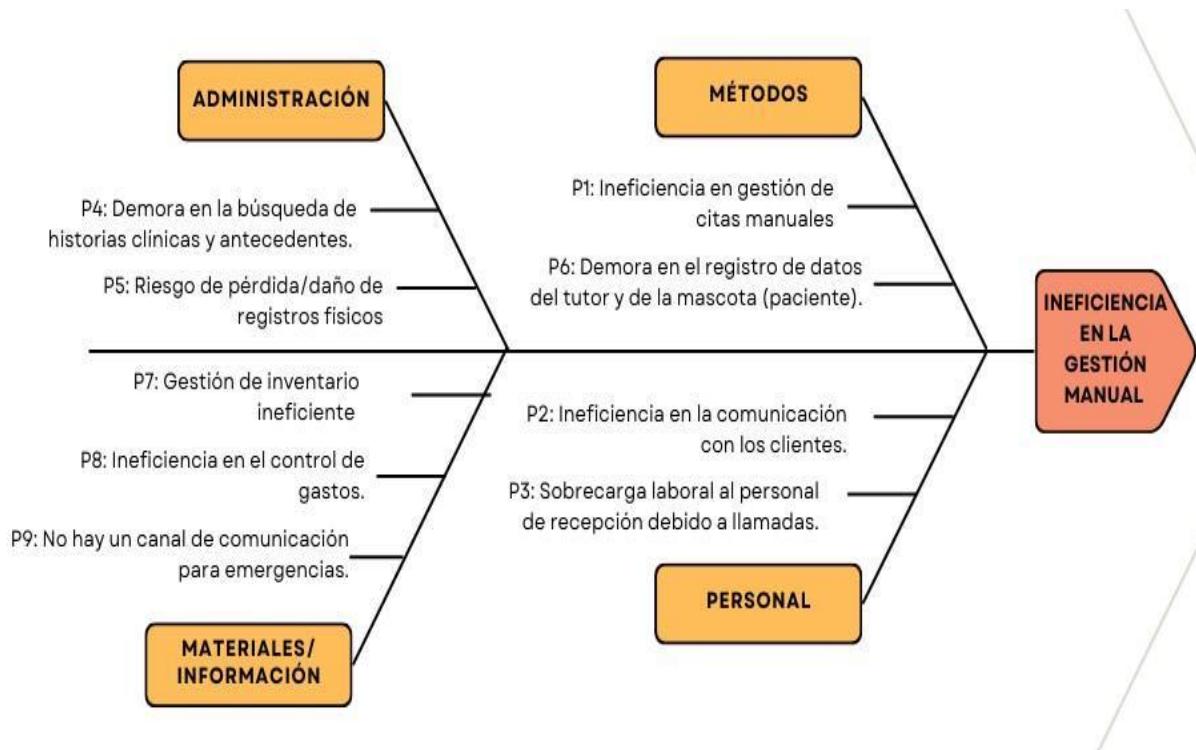
La implementación del Sistema de Información **VETCARE** se presenta como la solución estratégica para los problemas operativos y de gestión identificados en la clínica veterinaria. El sistema abordará directamente la ineficiencia, el riesgo de pérdida de

información y la sobrecarga laboral, transformando los procesos manuales y obsoletos en flujos de trabajo eficientes y digitalizados.

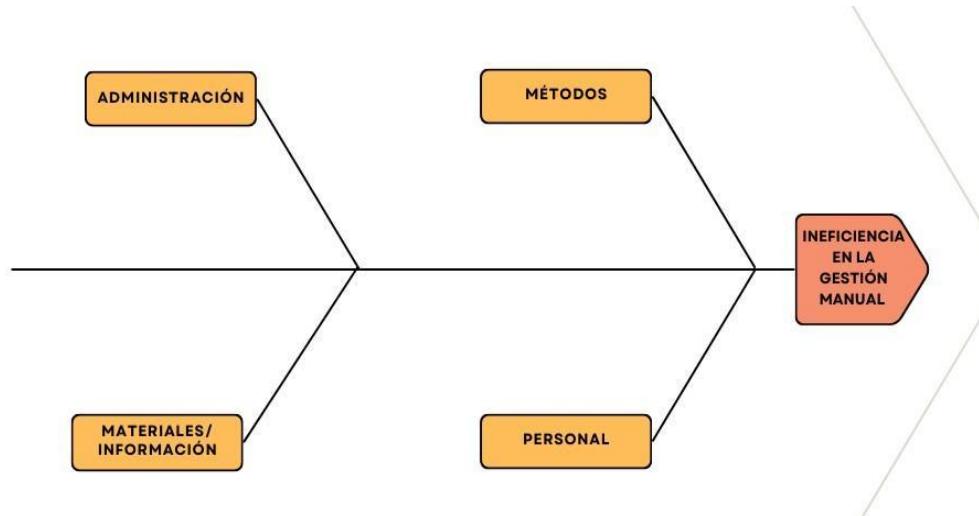
Las recomendaciones se basarán en 3 puntos:

- Digitalización de la información: Permite el acceso a los datos desde un solo lugar, eliminando el riesgo de pérdida.
- Automatización de procesos: Reduce el trabajo manual en la gestión de citas e inventario.
- Optimización de recursos: Ahorra en costos de papel y permite que el personal se enfoque en la atención de calidad.

1.9.- Diseñar el Diagrama de Ishikawa



2.- Identificar las principales categorías



3.- Identificar las causas

- Ineficiencia en gestionar citas manuales.
- Ineficiencia en la comunicación con los clientes.
- Demora en registro de datos del tutor y la mascota.
- Sobrecarga laboral.
- Demora en la búsqueda de historia clínicas y antecedentes.
- Gestión de inventario ineficiente.
- Riesgo de pérdida/daño de registros físicos.
- Ineficiencia en control de gastos.
- No hay un canal de comunicación formal para emergencias.

4.- Analizar y discutir el diagrama

1. Personal: El personal de recepción sufre una **sobrecarga operativa** debido al volumen de llamadas telefónicas para agendar citas, responder consultas y proporcionar información. Esta saturación genera:

- Comunicación inconsistente con los clientes.
- Tiempos de espera prolongados en llamadas.
- Posible deterioro en la calidad del servicio.
- Estrés laboral y posibles errores humanos.

2. Métodos: Los procesos manuales existentes presentan múltiples deficiencias:

- Agenda física propensa a duplicados, conflictos de horarios y pérdida de información
- Registro en papel de datos de tutores y mascotas que consume tiempo valioso.
- Falta de estandarización en los procedimientos de registro.
- Dificultad para acceder a información histórica rápidamente.

3. Materiales/ Información: La gestión documental física representa un riesgo crítico:

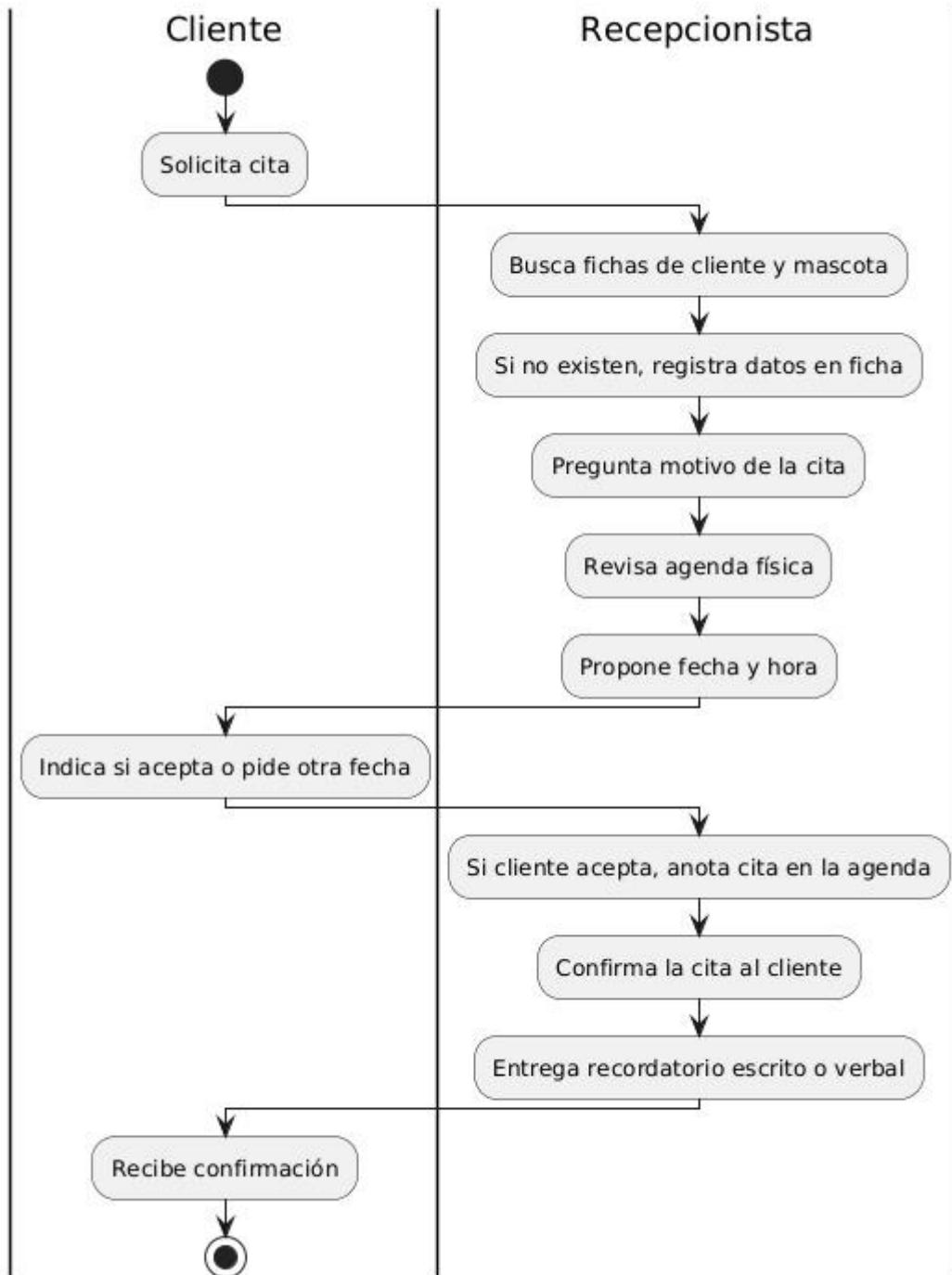
- Historias clínicas en papel vulnerables a pérdida, daño por agua/incendio, o extravío.
- Tiempo excesivo en localización de expedientes físicos.
- Imposibilidad de acceso simultáneo a los registros por múltiples usuarios.
- Deterioro natural de documentos en papel con el tiempo.
- Falta de backups o copias de seguridad de información crítica.

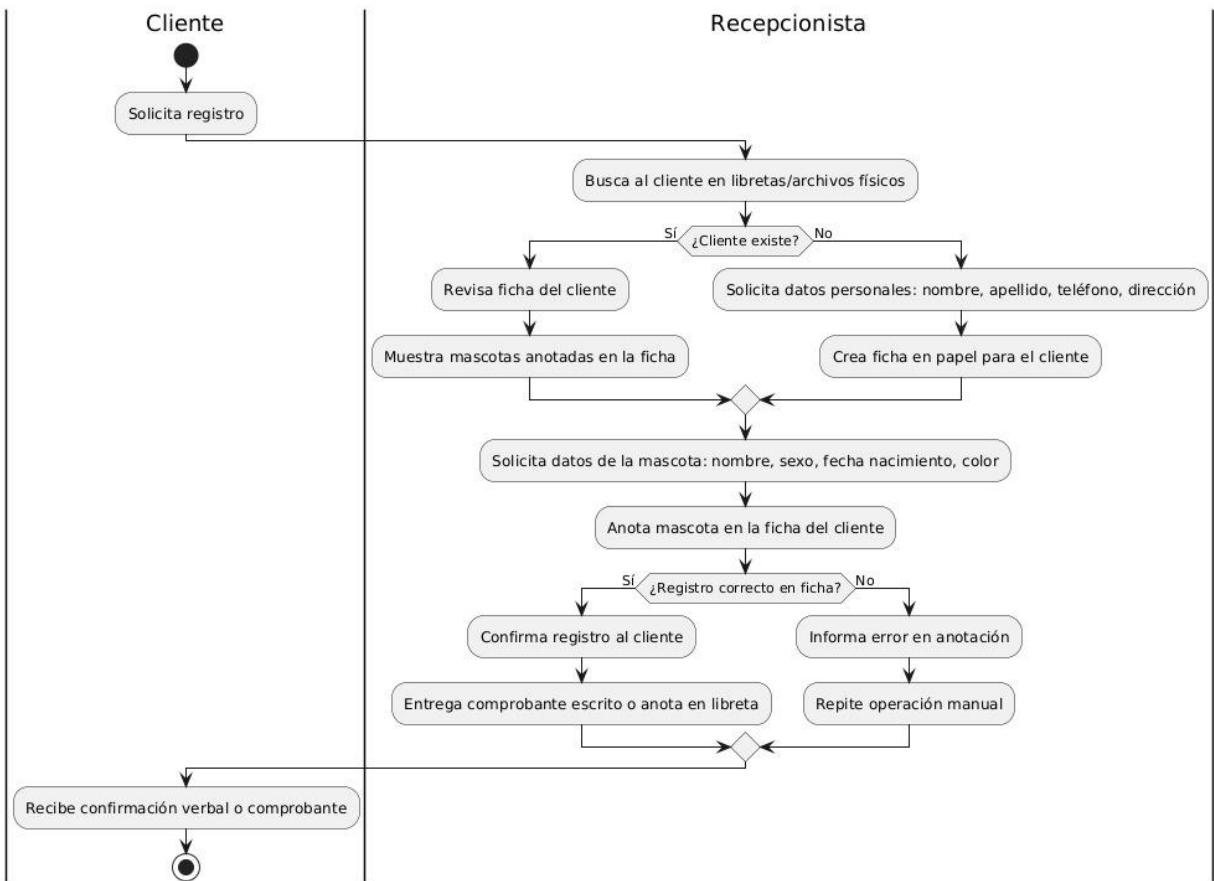
4. Administración: La gestión administrativa carece de sistemas adecuados:

- Inventario manual propenso a errores de conteo, stock outs o excesos.
- Control financiero rudimentario sin indicadores claros de gastos/ingresos.
- Falta de protocolos para comunicaciones de emergencia con clientes.
- Ausencia de planificación basada en datos históricos.
- Decisiones reactivas en lugar de proactivas por falta de información.

MODELO DE NEGOCIO

Diagrama de actividades





CAPITULO 2.- FLUJO DE TRABAJO: CAPTURA DE REQUISITOS

2.1.- Encontrar Actores y Casos de uso:

Tabla de Actores:

Actor	Rol	Función en el sistema
Dueño/Gerente de la Clínica	Responsable de la gestión estratégica y toma de decisiones.	Supervisar y acceder al dashboard para indicadores de negocio, generar reportes de ventas y productos, reportes de clientes con mayor gasto, controlar estadísticas de veterinarios y tomar decisiones sobre servicios y precios.
Médico Veterinario	Encargado de la atención médica de las mascotas.	Gestionar consultas y tratamientos, registro y aplicación de vacunas, ver y actualizar historial clínico, acceder a estadísticas de sus consultas, colaborar en agendamiento de citas.
Personal de Recepción	Atención inicial al cliente y gestión de citas.	Gestionar registro de clientes y mascotas, agendamiento y actualización de citas, inicio/cierre de sesión, asistencia en comunicación con clientes (recordatorios de citas, pagos y vacunas).
Tutor/Cliente	Dueño de la mascota, usuario final externo.	Acceder al sistema para registro e inicio de sesión, visualizar historial de mascota, agendar citas, recibir recordatorios, consultar y pagar facturas.
Personal Administrativo (Contabilidad/Almacén)	Encargado de inventario y finanzas.	Gestionar inventario y control de stock, gestión de servicios y precios, facturación de productos y servicios, pagos de facturas, generar reportes financieros y de inventario.
Administrador del Sistema	Responsable de roles y permisos.	Gestionar inicio/cierre de sesión, roles y permisos, asegurar que los perfiles (Cliente, Veterinario, Admin) tengan acceso a las funciones correctas.

Casos de Uso:

- **CU1:** Gestionar Inicio/Cierre de Sesión.
- **CU2:** Gestionar Roles y Permisos.
- **CU3:** Registrar Cliente.
- **CU4:** Registrar Mascota.
- **CU5:** Ver y Actualizar Historial Clínico.
- **CU6:** Agendar Citas.
- **CU7:** Registrar Vacunas.
- **CU8:** Gestionar Consultas y Tratamientos.
- **CU9:** Gestionar Inventario.
- **CU10:** Gestionar Servicios y Precios.
- **CU11:** Facturación de Productos y Servicios.
- **CU12:** Pagos de Facturas.
- **CU13:** Reporte de Historial Clínico.
- **CU14:** Reportes de Ventas y Productos.
- **CU15:** Reporte de Clientes con Mayor Gasto.
- **CU16:** Recordatorios de Citas, Pagos y Vacunas.
- **CU17:** Consultar y Pagar Facturas.
- **CU18:** Estadísticas de Veterinarios.
- **CU19:** Registro y Visualización de Vacunas.
- **CU20:** Control de Stock.
- **CU21:** Acceder a Dashboard de Indicadores de Negocio.
- **CU22:** Gestión de Proveedores.
- **CU23:** Gestión de Compras / Reabastecimiento de Insumos.
- **CU24:** Auditoría de Acciones del Sistema (Bitácora).
- **CU25:** Comunicación Interna (Notificaciones del Sistema).
- **CU26:** Mantenimiento de Catálogos (Especies, Razas, Vacunas, Servicios).
- **CU27:** Configuración del Sistema (Impuestos, Moneda, Horarios, Métodos de Pago).
- **CU28:** Integración con Pasarela de Pago.

TABLA DE CASOS DE USO:

CU1	Gestionar Inicio/Cierre de Sesión	Falta de un control de acceso seguro y autenticado para diferentes usuarios.
CU2	Gestionar Roles y Permisos	Inexistencia de un sistema para controlar los accesos y funcionalidades de cada tipo de usuario (veterinario, administrativo, etc.).
CU3	Registrar Clientes	Proceso manual y desorganizado para el registro de nuevos clientes.
CU4	Registrar Mascotas	Proceso manual y desorganizado para el registro de nuevas mascotas de acuerdo a sus respectivos dueños/propietarios.
CU5	Ver y Actualizar Historial Clínico	Historias clínicas dispersas en múltiples archivos físicos, dificultando el acceso rápido y preciso.
CU6	Agendar Citas	Gestión de citas y turnos ineficiente y propensa a errores.
CU7	Registrar Vacunas	Falta de un registro centralizado y fácil de consultar de las vacunas de cada mascota.
CU8	Gestionar Consultas y Tratamientos	Ausencia de un sistema que permita un registro ordenado y detallado de las atenciones médicas.
CU9	Gestionar Inventario	Falta de un control preciso del inventario de medicamentos e insumos, lo que genera escasez o pérdidas.
CU10	Gestionar Servicios y Precios	Inexistencia de un método para centralizar y mantener actualizada la información de los servicios y sus costos.
CU11	Facturación de Productos y Servicios	Procesos administrativos y de facturación manuales, lentos y propensos a errores de cálculo.
CU12	Pagos de Facturas	Ausencia de un registro digital y unificado de los pagos de clientes.
CU13	Reporte de Historial Clínico	Dificultad para generar reportes consolidados del historial de las mascotas para análisis.
CU14	Reportes de Ventas y Productos	Inexistencia de un sistema para generar reportes sobre ventas y productos de forma automática.
CU15	Reporte de Clientes con Mayor Gasto	Falta de información clara para identificar a los clientes más rentables.
CU16	Recordatorios de Citas, Pagos y Vacunas	Falta de un sistema automático para enviar recordatorios a los clientes, afectando la asistencia y el cobro.
CU17	Consultar y Pagar Facturas	Dificultad para los clientes de acceder y pagar sus facturas de forma remota.
CU18	Estadísticas de Veterinarios	Ausencia de métricas para evaluar el desempeño y la productividad de los médicos veterinarios.
CU19	Registro y Visualización de Vacunas	Proceso manual para el registro de vacunas y dificultad para que el cliente las consulte.
CU20	Control de Stock	Falta de visibilidad en tiempo real sobre la cantidad de productos disponibles en el almacén.
CU21	Acceder a Dashboard de Indicadores de Negocio	Ausencia de un panel de control con métricas clave para la toma de decisiones estratégicas.

CU22	Gestión de Proveedores.	Falta de un registro centralizado para almacenar y consultar datos de los proveedores, lo que dificulta el control de insumos y la relación con los mismos.
CU23	Gestión de Compras / Reabastecimiento de Insumos	Ausencia de un proceso automatizado para registrar compras y planificar reabastecimientos, lo que genera riesgo de desabastecimiento o sobrecompra de insumos.
CU24	Auditoría de Acciones del Sistema (Bitácora)	Inexistencia de un mecanismo para registrar cambios y operaciones realizadas por los usuarios, lo que dificulta el control y la trazabilidad en el sistema.
CU25	Comunicación Interna (Notificaciones del Sistema)	Falta de un canal digital de notificaciones internas que permita coordinar mejor las actividades entre el personal de la clínica.
CU26	Mantenimiento de Catálogos (Especies, Razas, Vacunas, Servicios)	Dificultad para mantener actualizada la información base de la clínica, provocando inconsistencias en registros de mascotas, vacunas y servicios.
CU27	Configuración del Sistema (Impuestos, Moneda, Horarios, Métodos de Pago)	Inexistencia de una interfaz que permita adaptar el sistema a las políticas internas de la clínica y normativas fiscales.
CU28	Integración con Pasarela de Pago	Falta de integración con medios de pago electrónicos en línea, lo que limita las opciones para los clientes y retrasa el proceso de cobro.

2.2.- Priorizar Casos de Uso:

ID	Caso de uso	Estado	Prioridad	Riesgo	Actores	Ciclo
CU1	Gestionar Inicio/Cierre de Sesión	Incluido	Crítica	Crítico	Usuario	C1
CU2	Gestionar Roles y Permisos	Incluido	Crítica	Crítico	Admin	C1
CU3	Registrar Clientes	Incluido	Crítica	Crítico	Recepción.	C1
CU4	Registrar Mascotas	Crítica	Crítico	Crítico	Usuario/ Veterinario	C1
CU5	Ver y Actualizar Historial Clínico	Incluido	Crítica	Crítico	Veterinario	C2
CU6	Agendar Citas	Incluido	Crítica	Crítico	Recepción.	C2
CU7	Registrar Vacunas	Incluido	Crítica	Crítico	Veterinario	C3
CU8	Gestionar Consultas y Tratamientos	Incluido	Crítica	Crítico	Veterinario	C2
CU9	Gestionar Inventory	Incluido	Crítica	Crítico	Admin/Rec.	C3
CU10	Gestionar Servicios y Precios	Incluido	Crítica	Crítico	Admin	C2
CU11	Pagos de Facturas	Incluido	Crítica	Crítico	Cliente	NE
CU12	Reporte de Historial Clínico	Incluido	Importante	Significativo	Veterinario / Admin	C3
CU13	Reportes de Ventas y Productos	Incluido	Importante	Significativo	Admin	C3
CU14	Reporte de Clientes con Mayor Gasto	Incluido	Importante	Ordinario	Admin	C3
CU15	Recordatorios de Citas, Pagos y Vacunas	Incluido	Importante	Ordinario	Cliente	NE
CU16	Consultar y Pagar Facturas	Incluido	Importante	Crítico	Cliente	NE
CU17	Visualización de Vacunas	Incluido	Importante	Significativo	Veterinario	NE
CU18	Control de Stock	Incluido	Crítica	Crítico	Recepción/Admin	C3
CU19	Acceder a Dashboard Estadístico.	Incluido	Secundaria	Significativo	Admin	C3
CU20	Gestión de Proveedores	Incluido	Crítica	Crítico	Admin	NE
CU21	Auditorias de Acciones del Sistema (Bitácora)	Incluido	Crítica	Crítico	Admin	C2
CU22	Mantenimiento de catálogos(Especies, Razas, Vacunas, Servicios)	Incluido	Crítica	Crítico	Admin	NE

CU23	Configuración del Sistema (Impuestos, Moneda, Horarios, métodos de Pago)	Incluido	Crítica	Crítico	Admin	C2
CU24	Integración con Pasarela de Pago	Incluido	Crítica	Crítico	Cliente / Admin	C2

TABLA CICLOS A PRESENTAR

ID	Caso de uso	Prioridad	Riesgo	Actores	Ciclo
CU1	Inicio/Cierre Sesión	Crítica	Crítico	Usuario	1
CU2	Gestionar Roles y Permisos	Crítica	Crítico	Usuario	1
CU3	Registrar Clientes	Crítica	Crítico	Admin	1
CU4	Registrar Mascotas	Crítica	Crítico	Usuario/ Veterinario	1
CU5	Ver y Actualizar Historial Clínico	Crítica	Crítico	Veterinario	2
CU6	Agendar Citas	Crítica	Crítico	Recepción.	2
CU8	Gestionar Consultas y Tratamientos	Crítica	Crítico	Veterinario	2
CU10	Gestionar Servicios y Precios	Crítica	Crítico	Admin	2
CU21	Auditorias de Acciones del Sistema (Bitácora)	Crítica	Crítico	Admin	2
CU23	Configuración del Sistema (Impuestos, Moneda, Horarios, métodos de Pago)	Crítica	Crítico	Admin	2
CU24	Integración con Pasarela de Pago	Crítica	Crítico	Cliente / Admin	2
CU7	Registrar Vacunas	Crítica	Crítico	Veterinario	C3
CU9	Gestionar Inventario	Crítica	Crítico	Admin	C3
CU12	Reporte de Historial Clínico	Crítica	Crítico	Admin/Vet	C3
CU13	Reportes de Ventas y Productos	Crítica	Crítico	Admin	C3
CU14	Reporte de Clientes con Mayor Gasto	Crítica	Crítico	Admin	C3
CU18	Control de Stock	Crítica	Crítico	Admin	C3
CU19	Acceder a Dashboard Estadístico.	Crítica	Crítico	Admin	C3

C1=Ciclo Uno

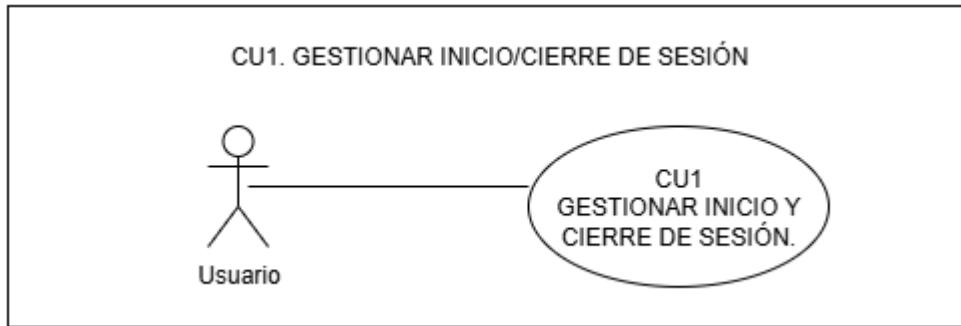
C2=Ciclo Dos

NE=No Especificado

2.3.- Detallar Casos de uso:

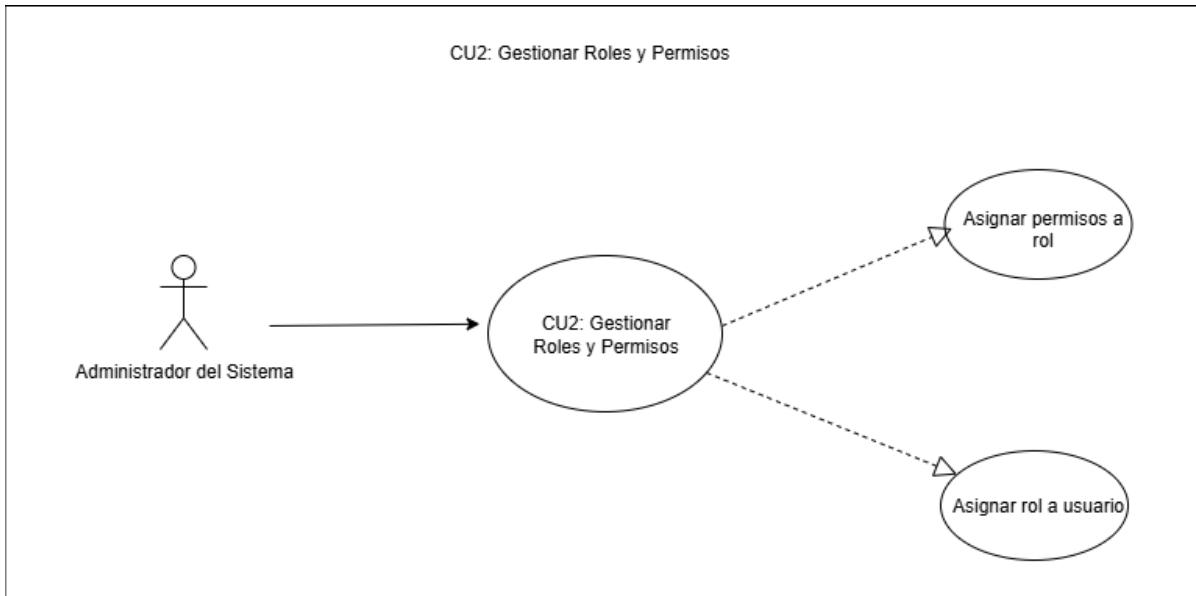
CICLO #1

CU1. Gestionar Inicio/Cierre Sesión



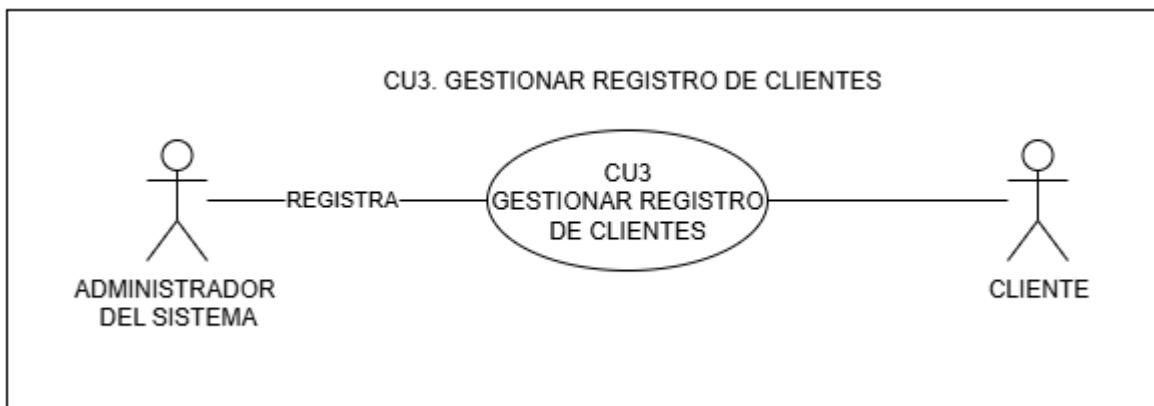
Nombre de CU	CU1. Gestionar inicio y cierre de sesión
Propósito	Permite iniciar sesión y cerrar sesión en la página web con sus respectivos usuarios.
Actores	Cliente / Administrador
Actor Principal	Cliente
Precondición	CU3. Registro de clientes y mascotas (opcional)
Flujo de trabajo	<p>Registro de Usuario.</p> <ul style="list-style-type: none">• Registro en la base de datos <p>Inicio de sesión.</p> <ul style="list-style-type: none">• Acceso con las credenciales <p>Cierre de sesión.</p>
Postcondición	Ninguna.
Excepción	Contraseña mínima de 6 caracteres.

CU2. Gestionar Roles y Permisos



Nombre de CU	CU2. Gestionar Roles y Permisos
Propósito	Permitir al Administrador del Sistema crear, modificar y asignar roles, así como otorgar permisos específicos a cada rol, garantizando el control de acceso de las funcionalidades del sistema.
Actores	Administrador del Sistema
Actor Principal	Administrador del Sistema
Precondición	El administrador debe haber iniciado sesión en el sistema Los roles y permisos deben estar definidos en la base de datos o existir la posibilidad de crearlos
Flujo de trabajo	El Administrador solicita gestionar roles y permisos. El sistema muestra la lista de roles existentes El Administrador puede: <ul style="list-style-type: none"> • Asignar permisos a un rol. • Asignar un rol a un usuario. El sistema valida los cambios. El sistema guarda los cambios y muestra confirmación.
Postcondición	Los usuarios quedan asociados a los roles asignados. Los permisos quedan vinculados a los roles correctamente.
Excepción	Si el rol ya existe, se notifica al administrador.

CU3.- Registrar Clientes



Nombre de CU	CU3. Gestión de clientes y mascotas
Propósito	Permitir el registro de nuevos clientes y médicos veterinarios en el sistema para habilitar su participación en los procesos de la clínica.
Actores	Cliente, Recepcionista, Administrador del Sistema
Actor Principal	Recepcionista
Precondición	Ninguna.
Flujo de trabajo	<p>El actor inicia el registro de usuario.</p> <p>El sistema solicita los datos obligatorios.</p> <p>El actor ingresa la información requerida.</p> <p>El sistema valida que no exista un registro duplicado.</p> <p>El sistema guarda la información en la base de datos.</p> <p>El sistema confirma el registro exitoso al actor.</p>
Postcondición	<p>El cliente queda registrado en la base de datos.</p> <p>CU1.-El usuario puede iniciar sesión y utilizar el sistema según sus permisos.</p>
Excepción	Si existe un usuario con datos duplicados, se notifica y no se realiza el registro.

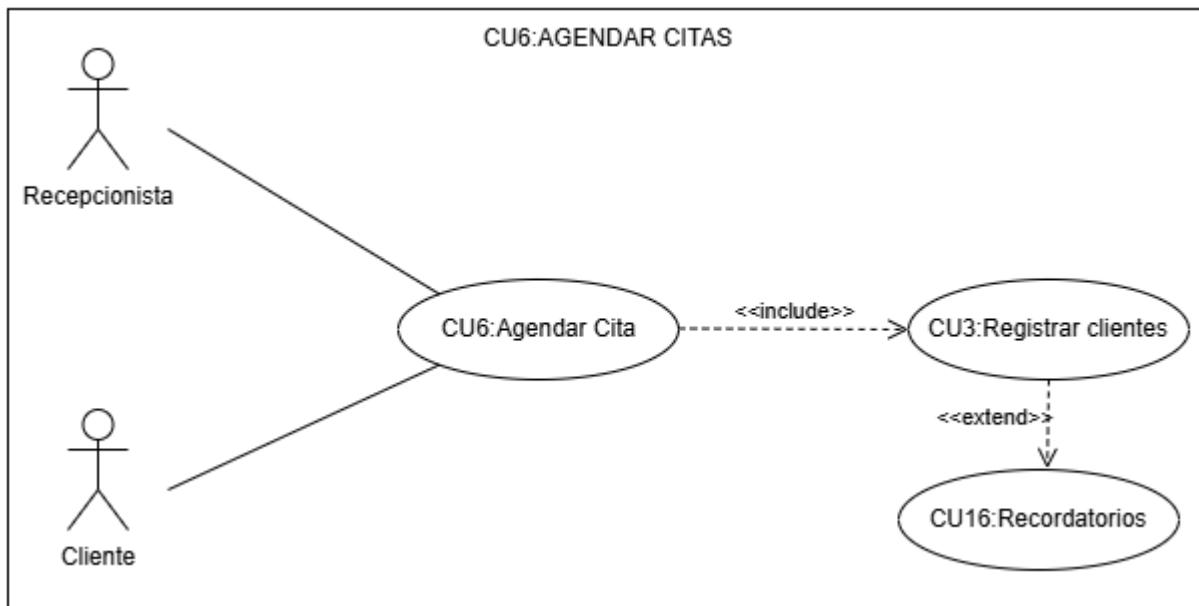
CU4. Registrar mascotas



Nombre de CU	CU4. Gestionar registro de mascotas
Propósito	Permitir el registro de nuevas mascotas asociadas a un cliente, garantizando que cada mascota quede vinculada a su propietario.
Actores	Cliente, Recepcionista, Veterinario.
Actor Principal	Veterinario, Cliente
Precondición	CU3. Registro de cliente en el sistema.
Flujo de trabajo	<ol style="list-style-type: none"> El actor solicita el registro de una nueva mascota. El sistema solicita datos de la mascota (nombre, especie, raza, edad, sexo, etc.). El actor ingresa la información requerida. El sistema valida que los datos sean correctos y que no exista un registro duplicado. El sistema guarda la información en la base de datos. El sistema confirma el registro exitoso.
Postcondición	<ul style="list-style-type: none"> - La mascota queda registrada y asociada al cliente correspondiente. - La mascota aparece disponible en el historial clínico y puede ser atendida en futuras consultas.
Excepción	<ul style="list-style-type: none"> - Si ya existe una mascota registrada con los mismos datos, se notifica al actor y no se crea un duplicado. - Si el cliente no está registrado, el sistema solicita primero ejecutar el CU3 (Registro de clientes).

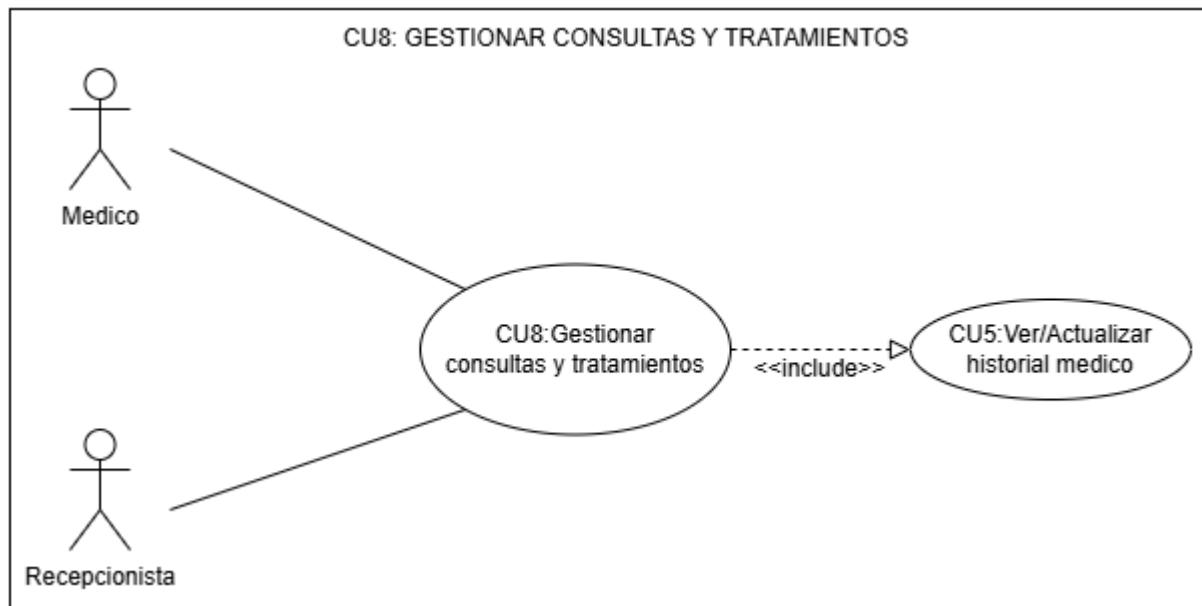
CU5.- Ver y Actualizar Historial Clínico

CU6.- Agendar Cita



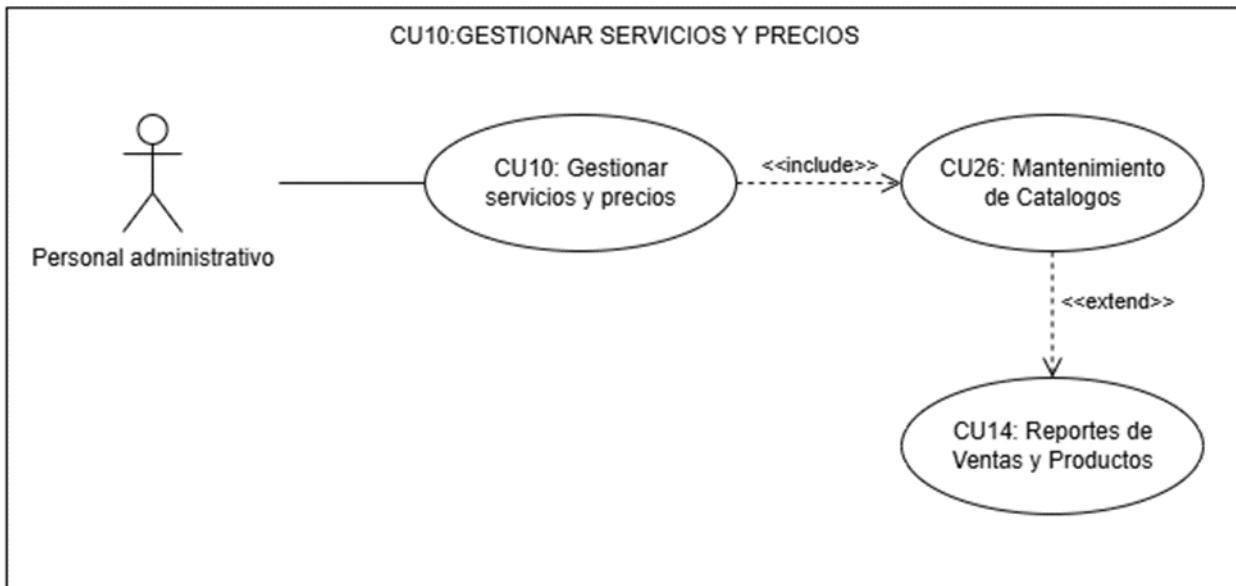
Nombre de CU	CU6. Agendar citas
Propósito	Permitir registrar citas médicas de mascotas de forma ordenada y automática, evitando choques de horario y errores manuales.
Actores	Cliente (dueño de mascota), Recepcionista
Actor Principal	Recepcionista
Precondición	El cliente y su mascota deben estar registrados en el sistema; el veterinario debe tener horario disponible.
Flujo de trabajo	<ul style="list-style-type: none">• El cliente solicita una cita indicando fecha, hora y motivo.• El sistema muestra la disponibilidad de veterinarios.• La recepcionista selecciona la opción deseada.• El sistema valida la disponibilidad.• Se registra la cita en la base de datos.• 6. El sistema genera un comprobante y programa recordatorios.
Postcondición	Cita registrada correctamente y recordatorio programado para el cliente.
Excepción	<ul style="list-style-type: none">• El cliente se debe registrar antes de continuar.• No hay horarios disponibles

CU8.- Gestionar Consultas y Tratamientos



Nombre de CU	CU8. Gestionar consultas y tratamientos
Propósito	Registrar y administrar las consultas veterinarias, diagnósticos, tratamientos y medicamentos indicados al paciente (mascota).
Actores	Veterinario, Recepcionista
Actor Principal	Veterinario
Precondición	La cita debe estar registrada y en estado “En Atención”; la mascota debe tener historial clínico.
Flujo de trabajo	<ul style="list-style-type: none"> • El recepcionista marca la llegada del cliente. • El sistema cambia el estado de la cita a “En Atención”. • El médico abre la ficha clínica y registra la evolución, diagnóstico y tratamiento. • El sistema guarda la información y actualiza el historial. • Si corresponde, se genera una receta o se agenda una cita de control.
Postcondición	<ul style="list-style-type: none"> • Consulta registrada con diagnóstico y tratamiento • historial actualizado.
Excepción	<ul style="list-style-type: none"> • No existe historial clínico, el sistema permite crear uno nuevo. • Falla al guardar datos, se muestra mensaje de error y se permite reintentar.

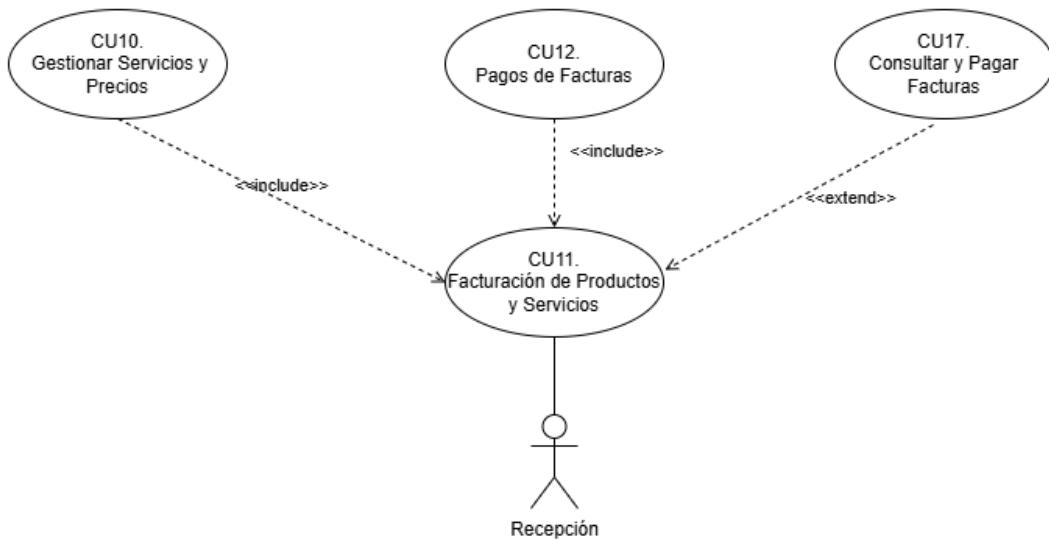
CU10.- Gestionar Servicios y Precios



Nombre de CU	CU10: Gestión de servicios y precios
Propósito	Permitir al personal administrativo registrar, modificar o eliminar servicios y productos, además actualizar sus precios en el sistema.
Actores	Personal Administrativo
Actor Principal	Personal Administrativo
Precondición	El usuario debe tener permisos administrativos y acceso al módulo de servicios.
Flujo de trabajo	<ul style="list-style-type: none"> • El personal administrativo accede al módulo de servicios. • Crea o edita un servicio. • Define o actualiza el precio y la vigencia. • El sistema valida y guarda los datos.
Postcondición	Los servicios y precios quedan registrados y actualizados correctamente en el sistema.
Excepción	<ul style="list-style-type: none"> • Servicio duplicado o precio inválido, el sistema muestra mensaje de error. • Fallo al guardar, el sistema no registra cambios.

CU11.- Facturación de Productos y Servicios

CU11. Facturación de Productos y Servicios

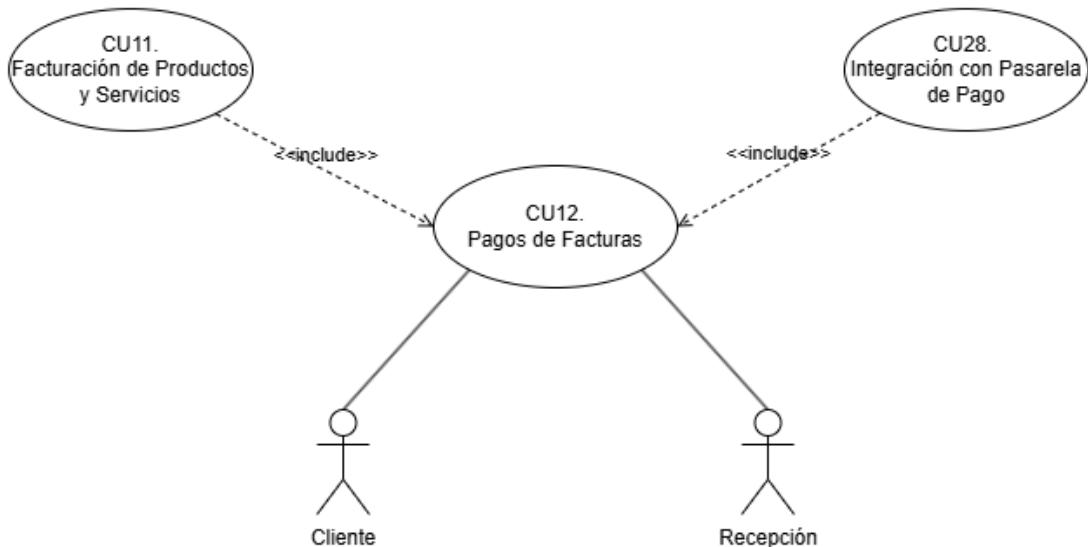


Nombre de CU	CU11. Facturación de Productos y Servicios.
Propósito	Permitir al personal de recepción generar facturas por la venta de productos y la prestación de servicios veterinarios, registrando los pagos y asociándolos a los clientes correspondientes.
Actores	Recepción / Cliente
Actor Principal	Recepción
Precondición	Los productos y servicios deben estar previamente registrados en el sistema (CU10). El cliente debe existir en la base de datos (CU3).
Flujo de trabajo	<p>El recepcionista accede al módulo de facturación.</p> <p>El sistema muestra la lista de productos y servicios disponibles con sus precios.</p> <p>El recepcionista selecciona los productos o servicios adquiridos por el cliente.</p> <p>El sistema calcula el total a pagar incluyendo impuestos si corresponde.</p> <p>El recepcionista confirma la venta y procede al registro del pago.</p> <p>El sistema genera la factura con la información del cliente, los productos/servicios y el monto total.</p> <p>El sistema guarda la factura en la base de datos y la asocia al cliente.</p> <p>El sistema ofrece la opción de imprimir o enviar la factura por correo electrónico.</p>

Postcondición	La factura queda registrada y disponible para consulta o pago posterior (CU17). Se actualiza el historial de ventas y pagos del cliente.
Excepción	Si el producto o servicio no está registrado, el sistema muestra un mensaje de error. Si el pago no se completa correctamente, la factura no se genera y el sistema solicita reintentar la operación.

CU12.-Pagos de Facturas

CU12. Pagos de Facturas

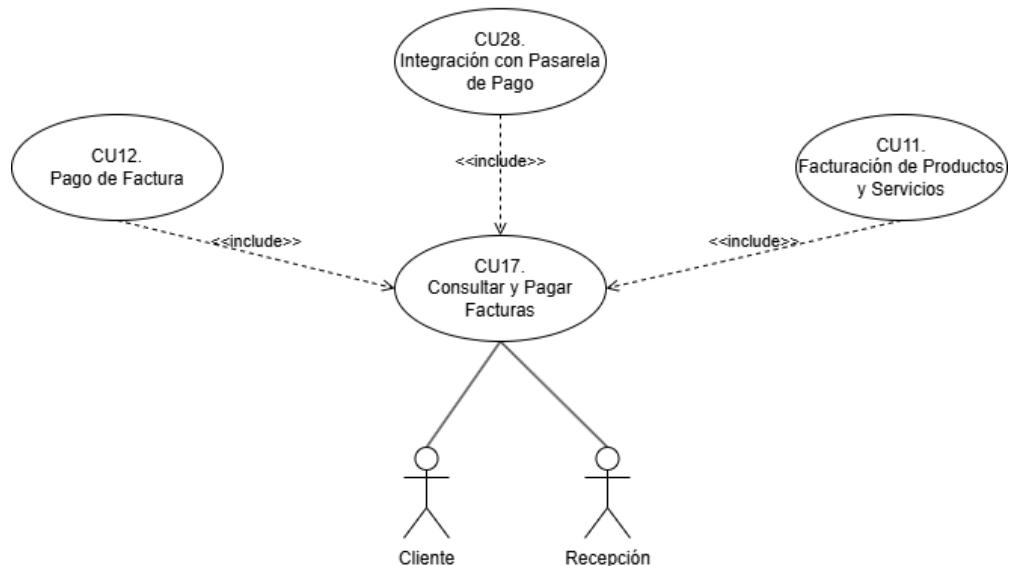


Nombre de CU	CU12. Pagos de Facturas.
Propósito	Permite al cliente o al personal de recepción registrar y procesar el pago de facturas generadas por productos o servicios, garantizando que el sistema actualice el estado de la factura y refleje el ingreso correspondiente.
Actores	Cliente / Recepción
Actor Principal	Cliente, cuando paga directamente. Recepción, cuando registra el pago en caja.
Precondición	Debe existir una factura generada previamente. El sistema de pagos debe estar operativo.
Flujo de trabajo	El actor accede al módulo de pagos. El sistema muestra la lista de facturas pendientes del cliente.

	<p>El actor selecciona la factura a pagar.</p> <p>El sistema muestra el detalle de la factura.</p> <p>El actor elige el método de pago.</p> <p>El sistema valida el monto ingresado y procesa el pago.</p> <p>Si el pago es exitoso, el sistema cambia el estado de la factura a “Pagada”.</p> <p>El sistema registra el pago en la base de datos y emite un comprobante o recibo.</p>
Postcondición	<p>La factura queda marcada como pagada.</p> <p>Se actualiza el historial financiero del cliente.</p> <p>El sistema registra la transacción en el módulo de facturación y contabilidad.</p>
Excepción	<p>Si el producto o servicio no está registrado, el sistema muestra un mensaje de error.</p> <p>Si el pago no se completa correctamente, la factura no se genera y el sistema solicita reintentar la operación.</p>

CU17.- Consultar Facturas

CU17. CONSULTAR Y PAGAR FACTURAS

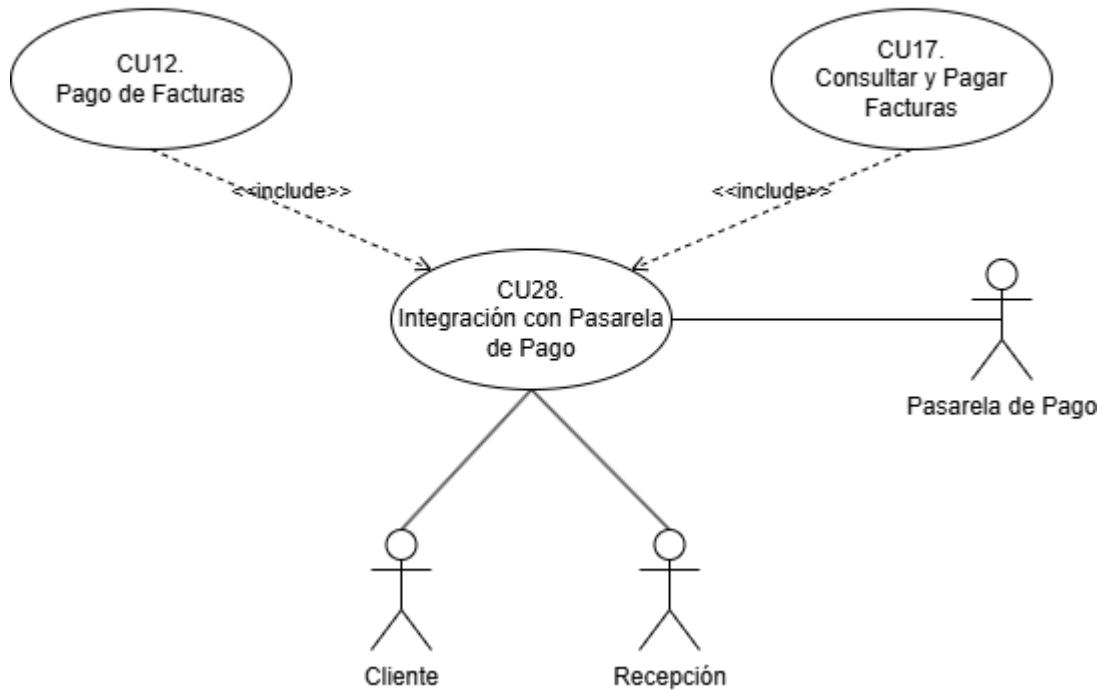


Nombre de CU	CU17. Consultar y Pagar Facturas
Propósito	Permite al cliente consultar las facturas generadas por la compra de productos o servicios veterinarios, verificar su estado (pendiente o pagada) y realizar el pago correspondiente a través del sistema.
Actores	Cliente / Recepción
Actor Principal	Cliente

Precondición	Debe existir al menos una factura generada previamente. El módulo de pagos debe estar disponible.
Flujo de trabajo	<p>El cliente accede al módulo “Facturas” del sistema.</p> <p>El sistema muestra la lista de facturas asociadas al cliente.</p> <p>El cliente selecciona una factura para ver su detalle.</p> <p>El sistema despliega los datos: productos o servicios facturados, monto total y estado de pago.</p> <p>Si la factura está pendiente, el cliente selecciona la opción “Pagar”.</p> <p>El sistema solicita el método de pago y confirma el monto.</p> <p>El cliente realiza el pago y el sistema procesa la transacción.</p> <p>Una vez confirmado el pago, el sistema actualiza el estado de la factura a “Pagada”.</p> <p>El sistema muestra un comprobante y ofrece la opción de descargar o imprimir.</p> <p>La recepción puede visualizar las facturas pagadas o pendientes de los clientes.</p>
Postcondición	<p>La factura cambia su estado a “Pagada”.</p> <p>El registro del pago se almacena en la base de datos del sistema.</p> <p>El cliente y la recepción pueden consultar el historial actualizado.</p>
Excepción	<p>Si no existen facturas registradas, el sistema muestra el mensaje “No se encontraron facturas disponibles”.</p> <p>Si el pago no se completa correctamente, el sistema mantiene el estado “Pendiente” y muestra un mensaje de error.</p>

CU28.- Integración con Pasarela de Pago

CU28. Integración con Pasarela de Pago

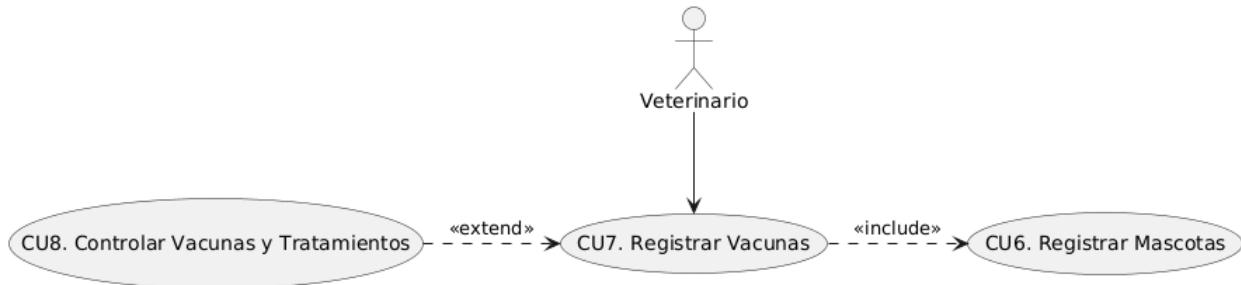


Nombre de CU	CU28. Integración con Pasarela de Pago
Propósito	Permite que el sistema VetCare se conecte con una pasarela de pago externa para procesar de forma segura los pagos de facturas realizados por clientes o registrados por el personal de recepción.
Actores	Pasarela de Pago / Cliente / Recepción
Actor Principal	Pasarela de Pago
Precondición	Debe existir al menos una factura generada previamente. El módulo de pagos debe estar disponible. El cliente o recepción deben haber iniciado el proceso de pago.
Flujo de trabajo	El cliente o la recepción seleccionan una factura para pagar. El sistema redirige la solicitud a la pasarela de pago. La pasarela de pago solicita los datos necesarios (método de pago, monto, tarjeta o QR). El usuario ingresa los datos y confirma la transacción. La pasarela valida la información y procesa el pago.

	<p>La pasarela envía la respuesta al sistema VetCare (aprobado o rechazado).</p> <p>Si el pago es exitoso, el sistema actualiza el estado de la factura a “Pagada”.</p> <p>El sistema genera y guarda el comprobante de pago.</p> <p>El cliente o la recepción pueden visualizar el resultado del pago y descargar el comprobante.</p>
Postcondición	<p>La factura cambia su estado a “Pagada”.</p> <p>Se almacena la confirmación de la transacción en la base de datos.</p> <p>El comprobante queda disponible para consulta o impresión.</p>
Excepción	<p>Si la conexión con la pasarela falla, el sistema muestra un mensaje de error y no registra el pago.</p> <p>Si los datos de la tarjeta o del método de pago son inválidos, la transacción es rechazada.</p> <p>Si el usuario cancela la operación antes de finalizar, el sistema mantiene la factura en estado “Pendiente”. Si el pago no se completa correctamente, el sistema mantiene el estado “Pendiente” y muestra un mensaje de error.</p>

CICLO #3

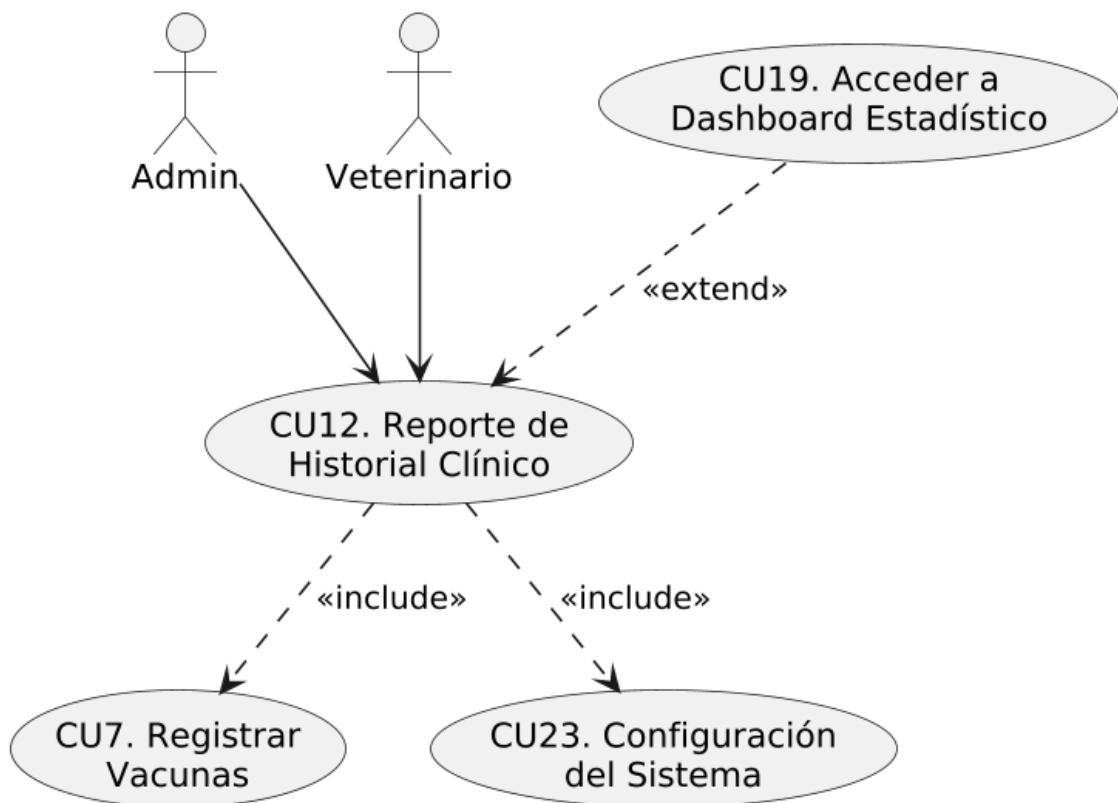
CU7: Registrar Vacunas



Nombre de CU	CU7. Registrar Vacunas
Propósito	Permite al veterinario registrar la aplicación de vacunas a la mascota/paciente, incluyendo lote, fecha, dosis y observaciones, generando comprobante/carnet.
Actores	Veterinario

Actor Principal	Veterinario
Precondición	Paciente y propietario registrados (CU3). Catálogo de vacunas activo. Stock disponible y vigente (CU9/CU18). Usuario autenticado.
Flujo de trabajo	<p>El veterinario accede al módulo “Vacunas”.</p> <p>El sistema solicita seleccionar paciente y esquema de vacunación.</p> <p>El veterinario ingresa datos de aplicación: vacuna, dosis, vía, lote, fecha/hora y observaciones.</p> <p>El sistema valida stock y vigencia del lote.</p> <p>El veterinario confirma el registro.</p> <p>El sistema descuenta stock, guarda el registro y actualiza el calendario de próximas dosis.</p> <p>El sistema ofrece imprimir/descargar carnet o comprobante.</p>
Postcondición	Vacuna registrada y vinculada al paciente; stock actualizado; próxima dosis programada si corresponde.
Excepción	Sin stock o lote vencido; paciente inexistente; datos incompletos; duplicidad de aplicación en ventana mínima; error de BD o permisos insuficientes.

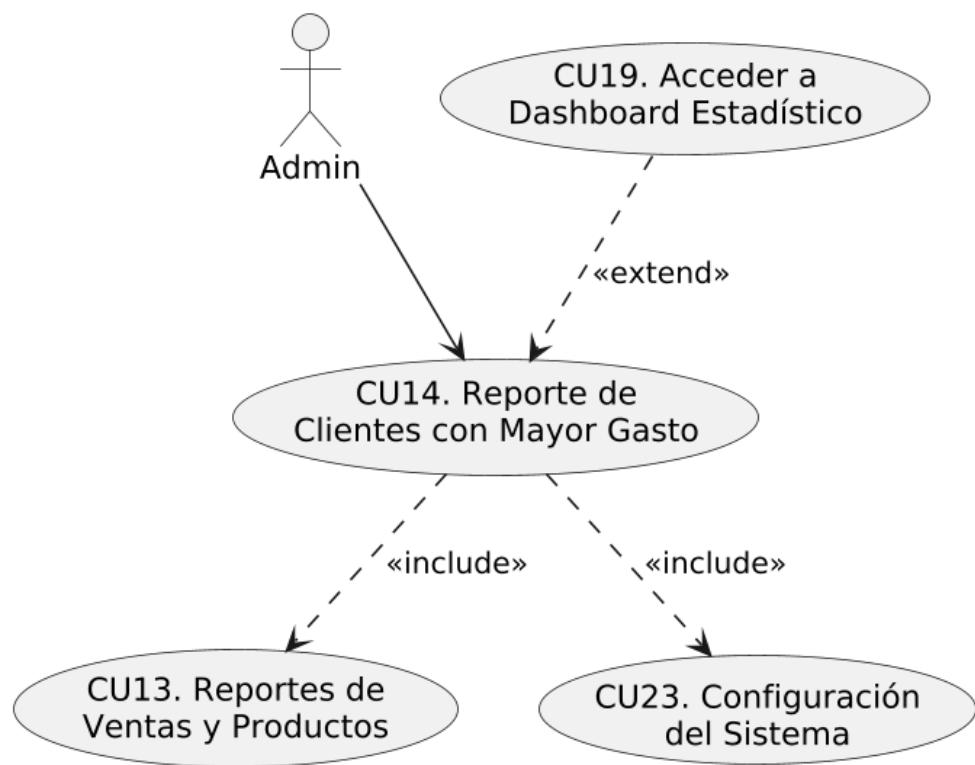
CU12. Reporte de Historial Clínico



Nombre de CU	CU12. Reporte de Historial Clínico
Propósito	Permite generar y visualizar el historial clínico del paciente (atenciones, diagnósticos, procedimientos y vacunas) con filtros y exportación.
Actores	Administrador del Sistema / Veterinario
Actor Principal	Administrador del Sistema
Precondición	Paciente registrado (CU3). Usuario autenticado con permisos. Existen atenciones y/o vacunas registradas (CU7)
Flujo de trabajo	El actor accede al módulo “Historial Clínico”. El sistema solicita selección de paciente y filtros (fechas, tipo de atención). El sistema consolida datos clínicos y de vacunas. El sistema muestra el reporte en pantalla (detalle o línea de tiempo).

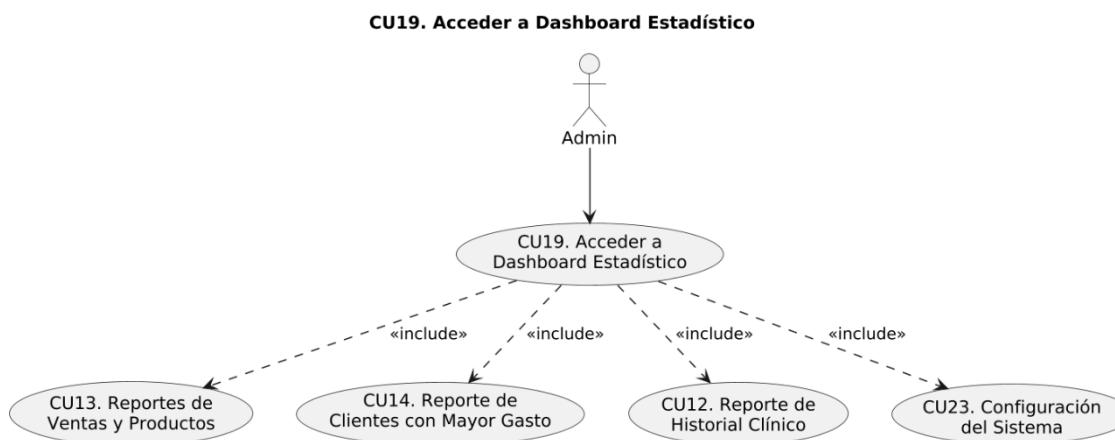
	El actor puede exportar o imprimir el reporte. Opcional: se accede al mismo desde el dashboard (CU19).
Postcondición	Reporte visualizado y/o exportado; se registra auditoría de la consulta.
Excepción	Paciente inexistente o sin datos; filtros inválidos; permisos insuficientes; error de BD o conectividad.

CU14. Reporte de Clientes con Mayor Gasto



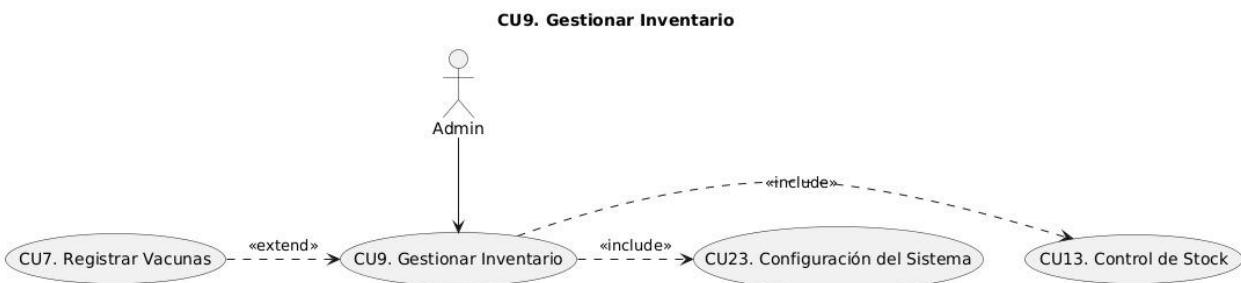
Nombre de CU	CU14. Reporte de Clientes con Mayor Gasto
Propósito	Permite identificar a los clientes con mayor monto de gasto en un periodo, con filtros y opciones de exportación.

Actores	Administrador del Sistema
Actor Principal	Administrador del Sistema
Precondición	Clientes registrados (CU3). Ventas/facturas disponibles (derivadas de CU11/CU13). Usuario autenticado.
Flujo de trabajo	<p>El administrador accede a “Reportes”.</p> <p>El sistema solicita rango de fechas y filtros (sucursal, categoría, top N).</p> <p>El sistema consolida ventas/pagos y calcula totales por cliente.</p> <p>El sistema ordena por monto y muestra tabla/gráficos.</p> <p>El administrador puede exportar o imprimir el reporte.</p>
Postcondición	Reporte mostrado y/o exportado; auditoría de consulta registrada.
Excepción	No hay datos en el periodo; filtros inválidos; permisos insuficientes; error de BD o conectividad.



Nombre de CU	CU19. Acceder a Dashboard Estadístico
Propósito	Permite visualizar KPIs y gráficos consolidados (ventas, productos, clientes top, actividad clínica), con filtros y navegación a reportes.
Actores	Administrador del Sistema

Actor Principal	Administrador del Sistema
Precondición	Usuario autenticado con permisos. Existen datos provenientes de CU13, CU14 y CU12. Configuración disponible (CU23).
Flujo de trabajo	<p>El administrador accede al “Dashboard Estadístico”.</p> <p>El sistema solicita o aplica filtros (fechas, sucursal, categoría, veterinario).</p> <p>El sistema agrega métricas y renderiza KPIs/gráficos.</p> <p>El administrador puede usar drill-down hacia reportes (CU13, CU14, CU12).</p> <p>El administrador puede ajustar preferencias/umbrales y guardar vistas.</p>
Postcondición	KPIs mostrados; preferencias de filtros guardadas; acceso a reportes detallados.
Excepción	Sin datos para el periodo; filtros inválidos; permisos insuficientes; error de BD o de conexión.



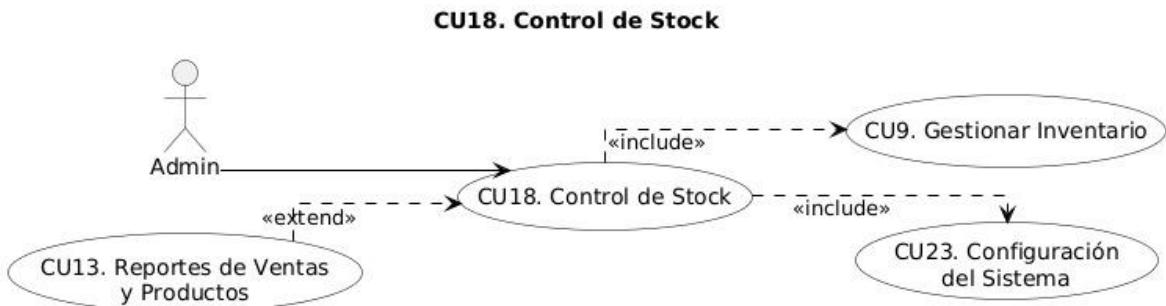
Nombre de CU	CU9. Gestionar Inventory
Propósito	Permite al administrador registrar ingresos, salidas y ajustes de stock; consultar existencias por producto/almacén y generar alertas por mínimos o vencimientos.
Actores	Administrador del Sistema
Actor Principal	Administrador del Sistema

Precondición	Catálogo de productos y almacenes registrado. Usuario autenticado con permisos. Políticas de stock mínimo configuradas (opcional).
Flujo de trabajo	<p>El administrador accede al módulo “Inventario”.</p> <p>El sistema muestra opciones: Ingresar, Retirar, Ajustar y Consultar.</p> <p>El administrador selecciona la acción y completa datos (producto, almacén, cantidad, motivo; lote/fechas si aplica).</p> <p>El sistema valida existencia, permisos y reglas (no stock negativo; lote vigente si corresponde).</p> <p>El sistema registra el movimiento y recalcula el stock del producto en el almacén.</p> <p>El sistema guarda auditoría y muestra confirmación con el stock actualizado.</p> <p>Si el stock queda en o por debajo del mínimo/vencimiento próximo, el sistema genera alerta.</p>
Postcondición	Movimiento registrado; stock actualizado; auditoría guardada; alertas generadas si corresponde.
Excepción	Stock insuficiente para retiro; ajuste dejaría stock < 0; lote inexistente/vencido; datos incompletos; permisos insuficientes; error de base de datos o conectividad.

CU13. Reportes de Ventas y Productos



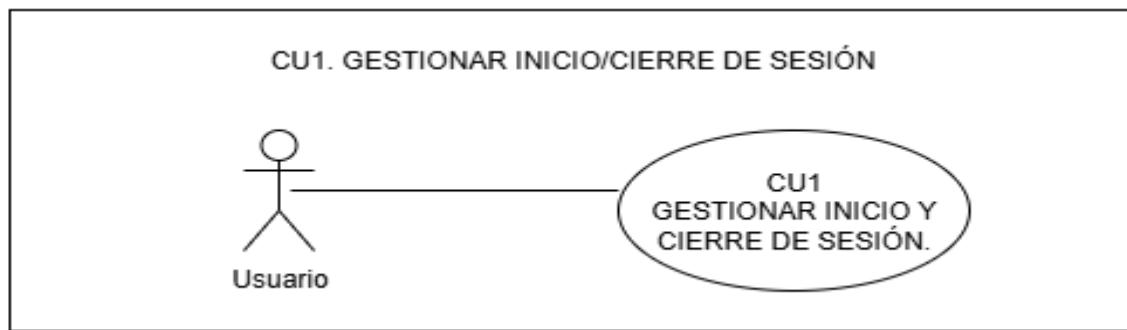
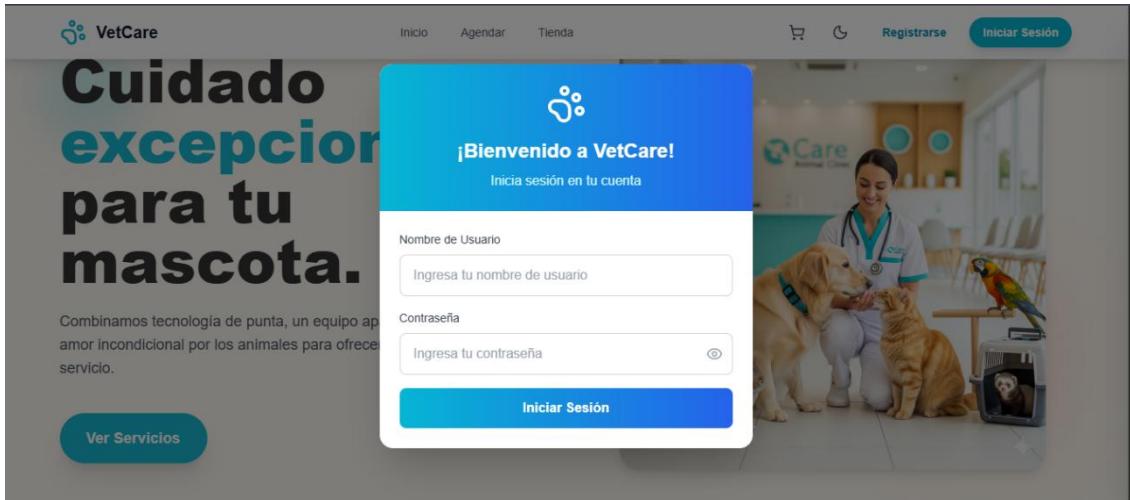
Nombre de CU	CU13. Reportes de Ventas y Productos
Propósito	Permite al administrador generar reportes de ventas por periodo, producto/servicio, método de pago y sucursal, con totales, KPIs y exportación.
Actores	Administrador del Sistema
Actor Principal	Administrador del Sistema
Precondición	Ventas/facturas registradas (CU11). Usuario autenticado con permisos.
Flujo de trabajo	<p>El administrador accede al módulo “Reportes”.</p> <p>El sistema solicita rango de fechas y filtros (sucursal, categoría, producto, método de pago).</p> <p>El sistema consolida datos de facturación y calcula métricas (ingresos, unidades, ticket promedio, top N).</p> <p>El sistema muestra tablas y/o gráficos y permite drill-down al detalle de facturas.</p> <p>El administrador puede exportar o imprimir el reporte.</p>
Postcondición	Reporte visualizado y/o exportado; queda registro de auditoría de la consulta.
Excepción	Sin datos en el periodo; filtros inválidos; error de BD o de conexión; permisos insuficientes.



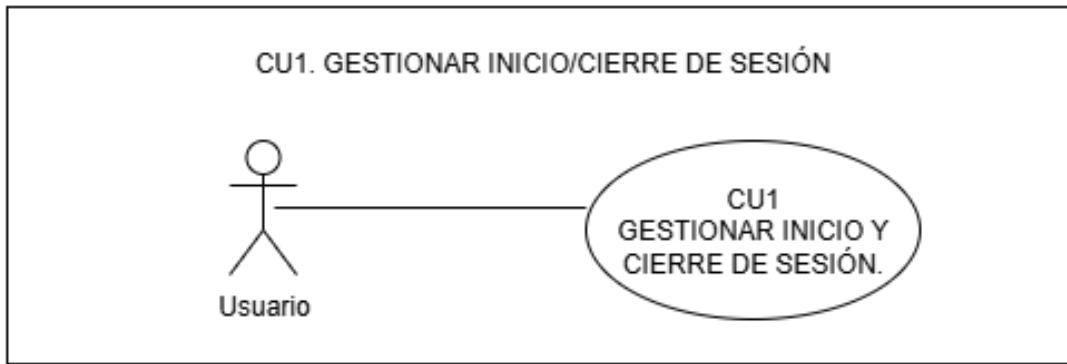
Nombre de CU	CU18. Control de Stock
Propósito	Permite al administrador monitorear existencias por producto/almacén, mínimos y vencimientos, generando alertas y sugerencias de reposición.
Actores	Administrador del Sistema
Actor Principal	Administrador del Sistema
Precondición	Inventario inicial y movimientos registrados (CU9). Políticas de stock mínimo/alertas configuradas (CU23). Usuario autenticado.
Flujo de trabajo	<p>El administrador accede a “Control de Stock”.</p> <p>El sistema lista existencias por producto/almacén con indicadores (mínimo, crítico, vencimiento).</p> <p>El administrador aplica filtros (categoría, estado, próximos a vencer).</p> <p>El sistema muestra alertas y sugiere cantidades de reposición según consumo histórico.</p> <p>El administrador marca alertas como revisadas y/o genera solicitud de compra/traslado.</p> <p>El sistema permite exportar la vista o enviar notificaciones.</p>
Postcondición	Estado de stock revisado; alertas atendidas o planificadas; registros de seguimiento actualizados.

Excepción	Políticas de mínimo no configuradas; datos inconsistentes (stock negativo); error de BD o permisos insuficientes.

2.4.- Prototipar la Interfaz de Usuario:

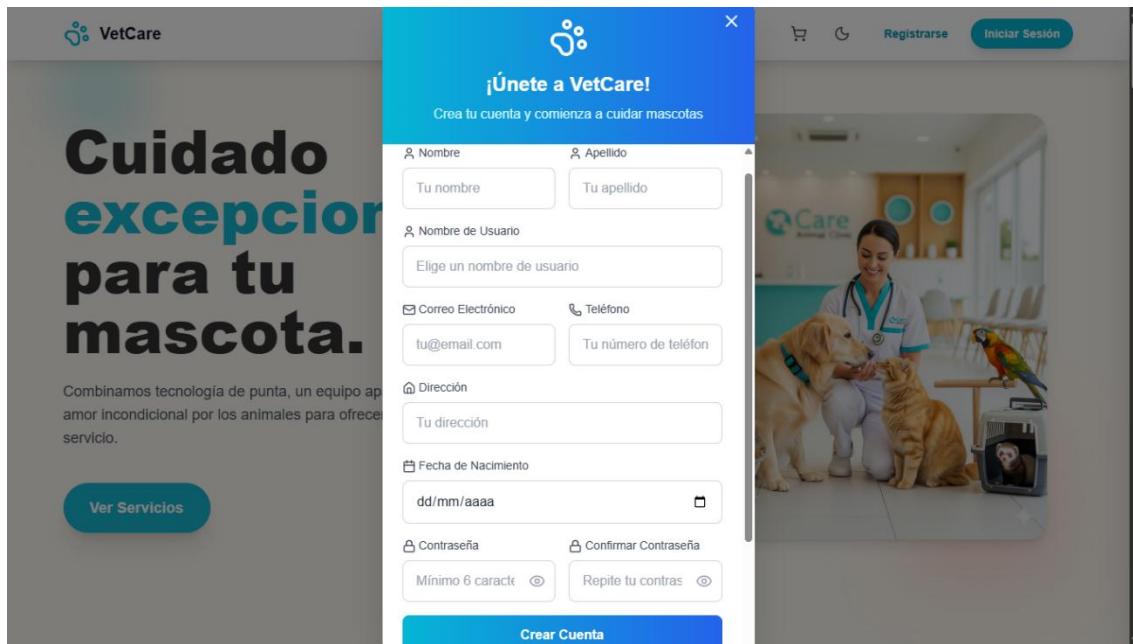


CU1:Inicio/Cierre Sesión. - Este caso de uso se encarga de que los usuarios (como el personal, veterinarios o administradores) puedan entrar y salir de la aplicación de forma segura. Esto soluciona el problema de no tener un control de acceso para el sistema. Es una función clave y crítica para el proyecto.



CU2:Gestionar Roles y Permisos.-

Este caso de uso le permite al Administrador del Sistema controlar lo que cada tipo de usuario puede ver y hacer en la aplicación. Es vital para asegurar que cada perfil (como el veterinario o el personal administrativo) tenga acceso a las funciones que necesita y no a otras. Su prioridad es crítica y su riesgo es crítico.



CU3: Registrar Clientes.- Este caso de uso es para que el Personal de Recepción pueda registrar de manera organizada a nuevos clientes en el sistema. Esto elimina el proceso manual y desorganizado que se usaba antes. Es una función crítica para el proyecto y forma parte del primer ciclo de desarrollo.

Registrar Nuevo Cliente

Datos del Propietario

Nombre Apellido
 Teléfono Correo electrónico (opcional)
 Dirección
 Fecha de nacimiento Username

Datos de la Mascota

Nombre de la mascota Selecciona especie
 Fecha de nacimiento masc Selecciona sexo
 Escribe para buscar raza

Bienvenido
 Panel de administración
 Total M **0**
 Ingresos **\$0**

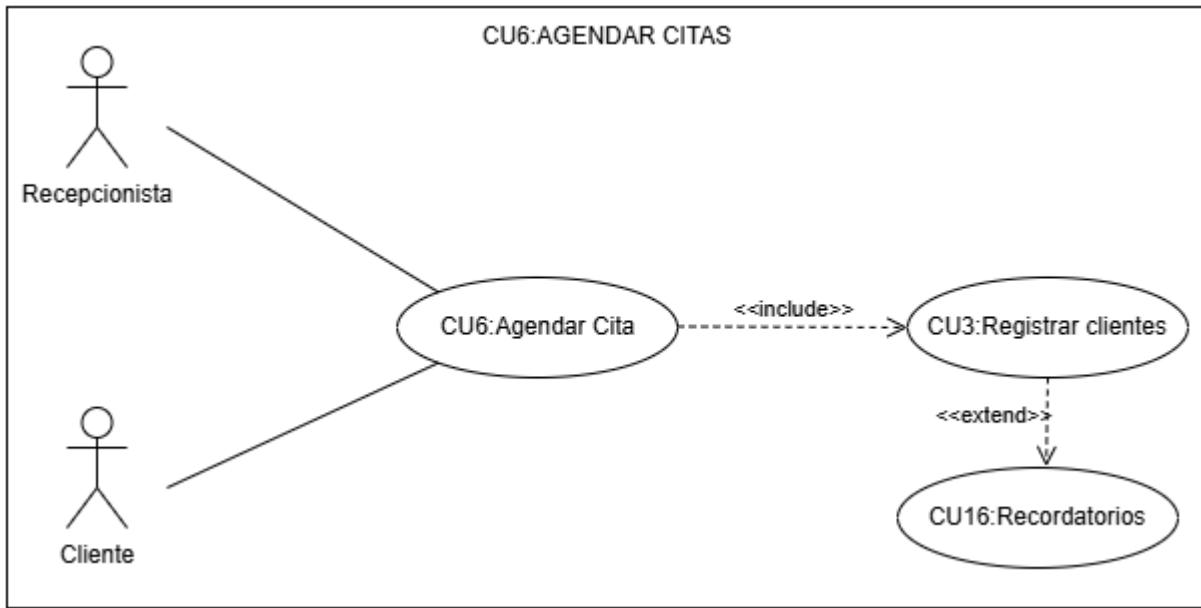
Nuevo Cliente

Administración
 Dashboard
 Ventas y Stock >
 Gestión de Usuarios >
 Reportes >

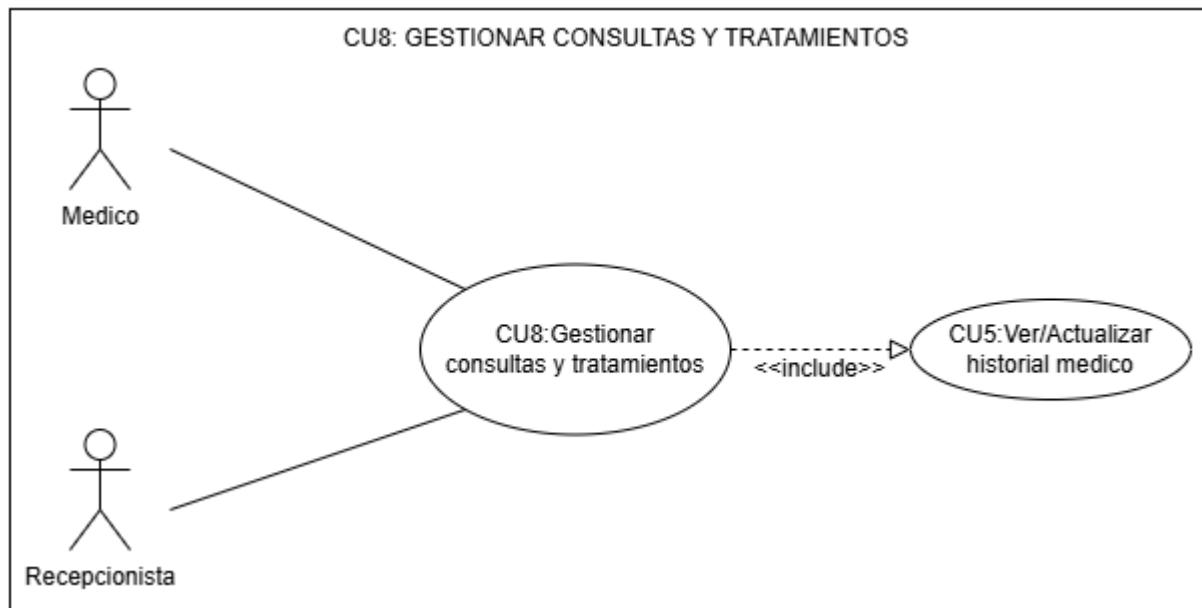
Citas por Mes



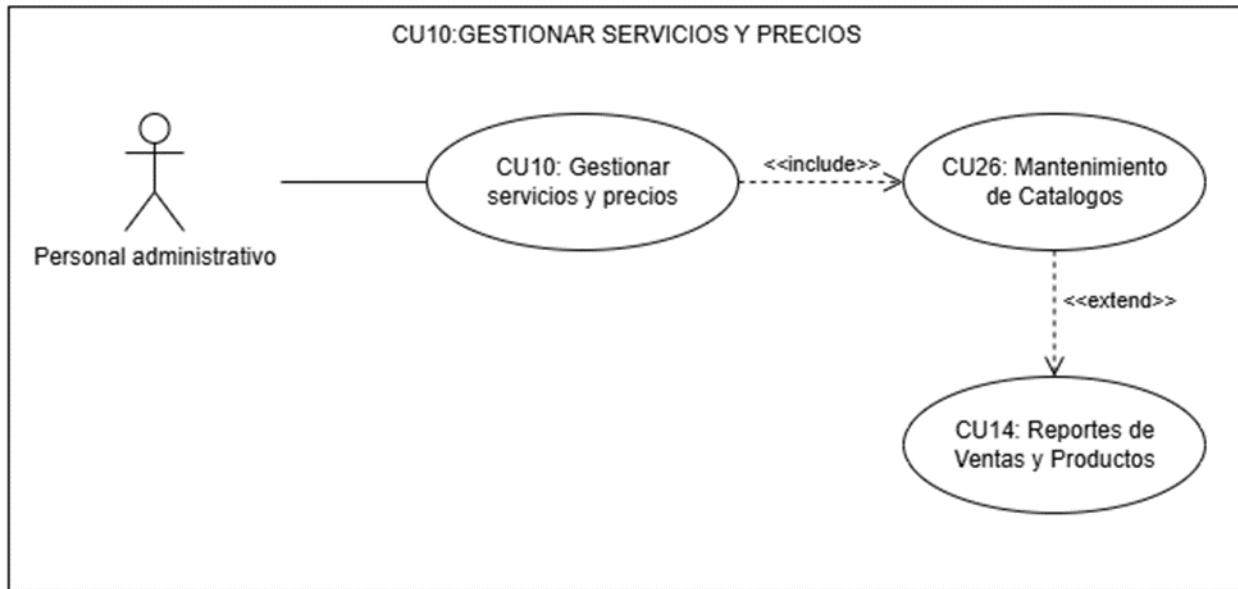
CU4: Registrar Mascotas.- Este caso de uso es para que el Personal de Recepción pueda registrar de manera organizada a nuevas Mascotas por cada propietario en el sistema. Esto elimina el proceso manual y desorganizado que se usaba antes. Es una función crítica para el proyecto y forma parte del primer ciclo de desarrollo.



CU6: Agendar Cita. Este caso de uso permite al usuario agendar una cita para su mascota con un veterinario disponible en el sistema. A través de este proceso, tanto los propietarios como los veterinarios pueden programar y gestionar las citas de manera organizada, eliminando el proceso manual previo y mejorando la eficiencia en la clínica. Esta función crítica facilita la atención oportuna de las mascotas y garantiza una experiencia de usuario fluida, siendo parte del primer ciclo de desarrollo del sistema.

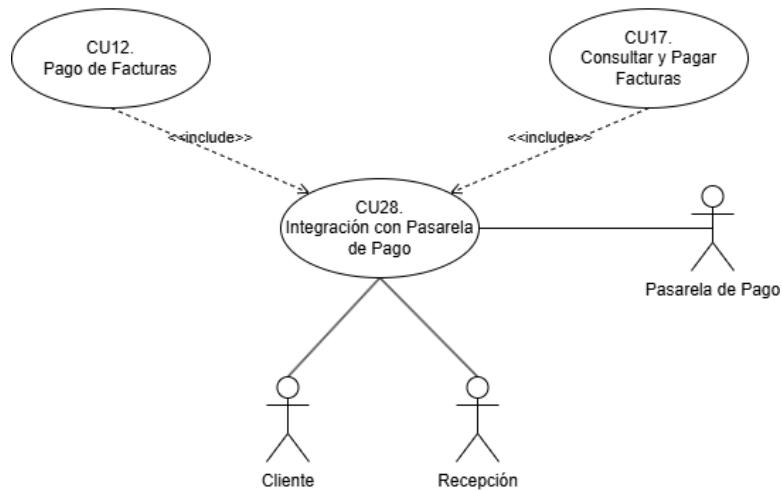


CU8: Gestionar Consultas y Tratamientos. Este caso de uso permite al personal autorizado gestionar y actualizar el diagnóstico de las consultas médicas de las mascotas en el sistema. A través de este proceso, se asegura que los datos médicos sean actualizados de manera eficiente, eliminando la necesidad de procesos manuales y mejorando la precisión en el registro de información clínica. Además, el sistema mantiene un registro de auditoría para cada acción realizada, asegurando transparencia y trazabilidad. Esta función es esencial para la gestión de la información médica de las mascotas, siendo parte del primer ciclo de desarrollo del sistema.



CU10: Gestión de servicios y precios- Este caso de uso permite al personal autorizado gestionar los servicios ofrecidos por la clínica veterinaria, así como establecer y actualizar los precios correspondientes a cada uno de ellos. Mediante esta función, se busca asegurar una gestión adecuada de los servicios disponibles, garantizando que los precios sean actualizados de manera eficiente y consistente en el sistema. Esto elimina el manejo manual de precios y servicios, mejorando la organización y facilitando el control sobre las tarifas. Esta funcionalidad es crucial para la correcta administración económica de la clínica y forma parte del ciclo de desarrollo del sistema.

CU28. Integración con Pasarela de Pago

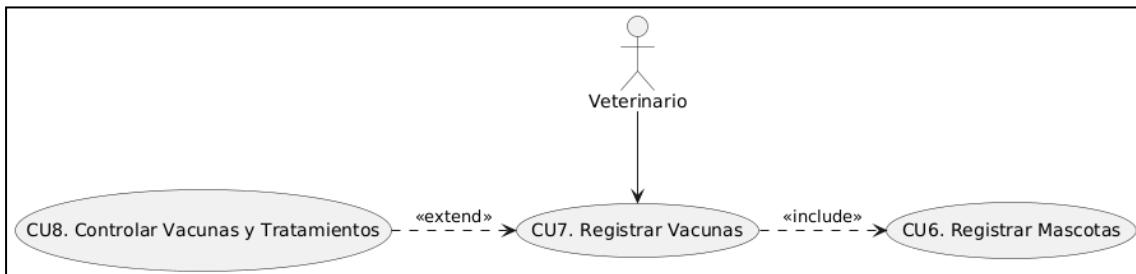


CU28: Integración con Pasarela de Pago- Este caso de uso permite la integración del sistema con una pasarela de pago, facilitando a los usuarios el proceso de pago de servicios en línea de manera segura y eficiente. A través de esta funcionalidad, los propietarios de mascotas pueden realizar pagos de citas,

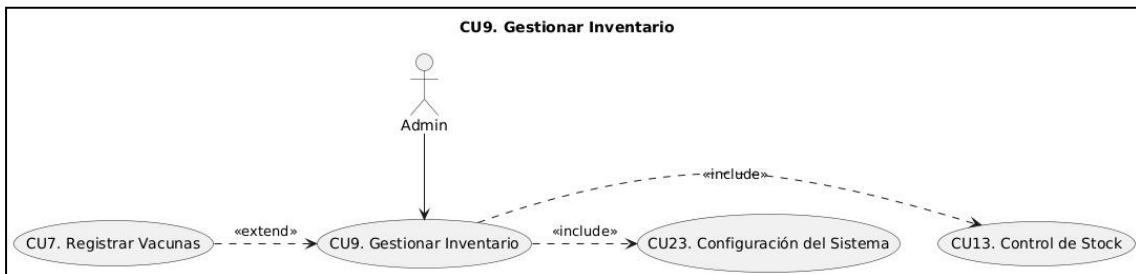
tratamientos y otros servicios directamente desde la plataforma, sin necesidad de procesos manuales. La integración con la pasarela de pago asegura una experiencia fluida para el usuario y una gestión adecuada de las transacciones, brindando seguridad tanto al propietario como a la clínica. Esta característica es fundamental para optimizar la gestión de cobros y mejorar la experiencia del usuario, siendo una parte clave en el ciclo de desarrollo del sistema.

CICLO #3

CU7: Registrar Vacunas.- Este caso de uso permite al personal veterinario registrar y actualizar la información de las vacunas aplicadas a cada mascota dentro del sistema. A través de este proceso, se registran los datos esenciales como tipo de vacuna, dosis, fecha de aplicación, próxima dosis y lote, garantizando un seguimiento adecuado del historial inmunológico.

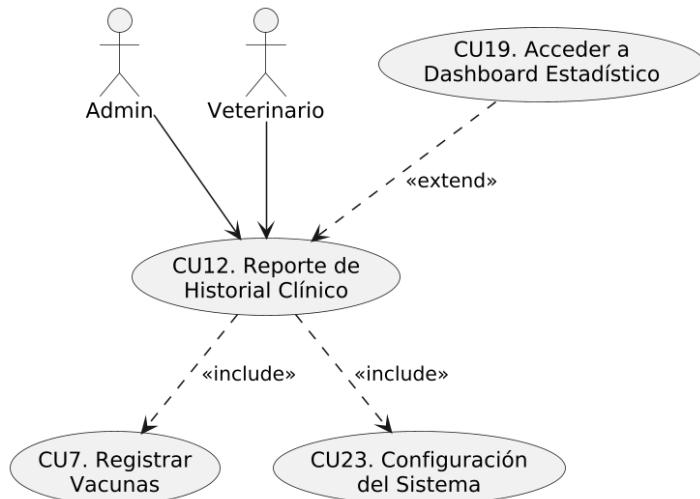


CU9: Gestionar Inventario.- Permite al administrador del sistema controlar y registrar el movimiento de productos, insumos y materiales dentro del inventario de la clínica veterinaria. A través de esta función, se pueden realizar operaciones de ingreso, actualización y baja de existencias. El sistema automatiza el control de entradas y salidas mediante reportes, optimizando la gestión de recursos.



CU12: Reporte de Historial Clínico.- Este caso de uso permite al personal autorizado generar y visualizar reportes completos del historial clínico de cada mascota registrada en el sistema. A través de esta funcionalidad, se consolidan los datos de consultas, tratamientos, vacunas y observaciones médicas en un solo documento, facilitando el seguimiento del estado de salud del paciente.

CU12. Reporte de Historial Clínico

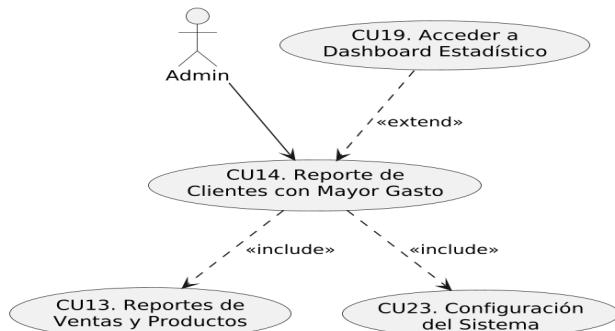


CU13: Reportes de Ventas y Productos: Permite al administrador generar y consultar reportes detallados sobre las ventas realizadas y el comportamiento de los productos en la clínica veterinaria. A través de esta funcionalidad, el sistema consolida la información de facturación, inventario y movimientos de productos, presentando indicadores clave de desempeño comercial.

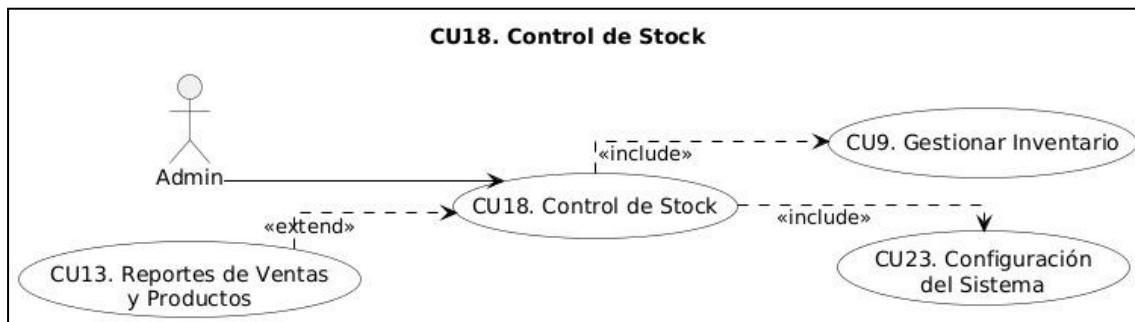


CU14: Reporte de Clientes con Mayor Gasto.- Este caso de uso permite al administrador generar reportes que identifican a los clientes con mayor nivel de gasto dentro de la clínica veterinaria. A través de esta funcionalidad, el sistema analiza los registros de facturación y consumo de servicios para determinar los clientes más rentables, permitiendo segmentar la base de datos y establecer estrategias de fidelización.

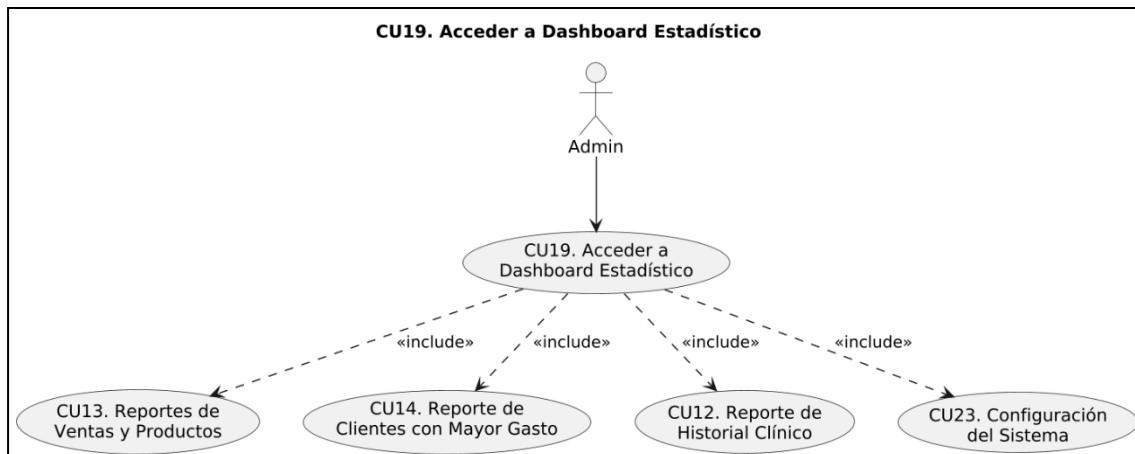
CU14. Reporte de Clientes con Mayor Gasto



CU18: Control de Stock.- Este caso de uso permite al administrador supervisar y mantener actualizado el nivel de existencias de los productos e insumos disponibles en la clínica veterinaria. A través de esta funcionalidad, el sistema muestra el estado actual del inventario, genera alertas cuando los niveles de stock alcanzan cantidades mínimas y permite realizar ajustes o correcciones en los registros.

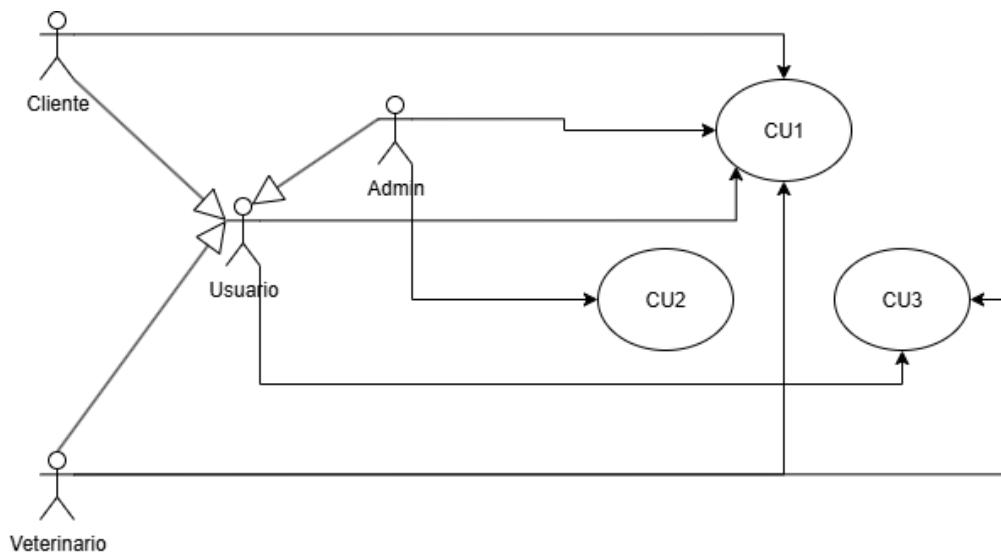


CU19: Acceder a Dashboard Estadístico.- Este caso de uso permite al administrador visualizar en tiempo real indicadores y métricas clave sobre el desempeño general de la clínica veterinaria. A través del dashboard, el sistema presenta información consolidada de ventas, consultas, tratamientos, control de stock y desempeño del personal, mediante gráficos y tablas dinámicas. Esta funcionalidad facilita el análisis comparativo y la toma de decisiones estratégicas basadas en datos precisos y actualizados.

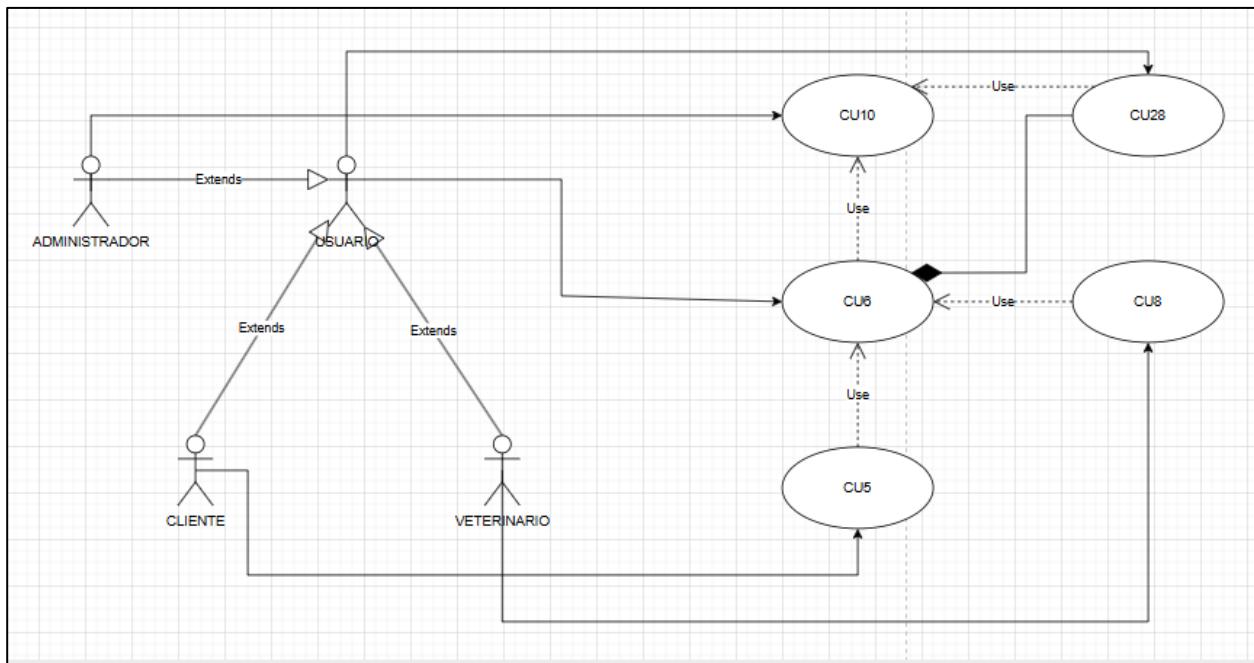


2.5.- Estructurar Modelos de Casos de Uso:

CICLO 1.-



CICLO 2.-



CU5.- Ver y Actualizar Historial Clínico

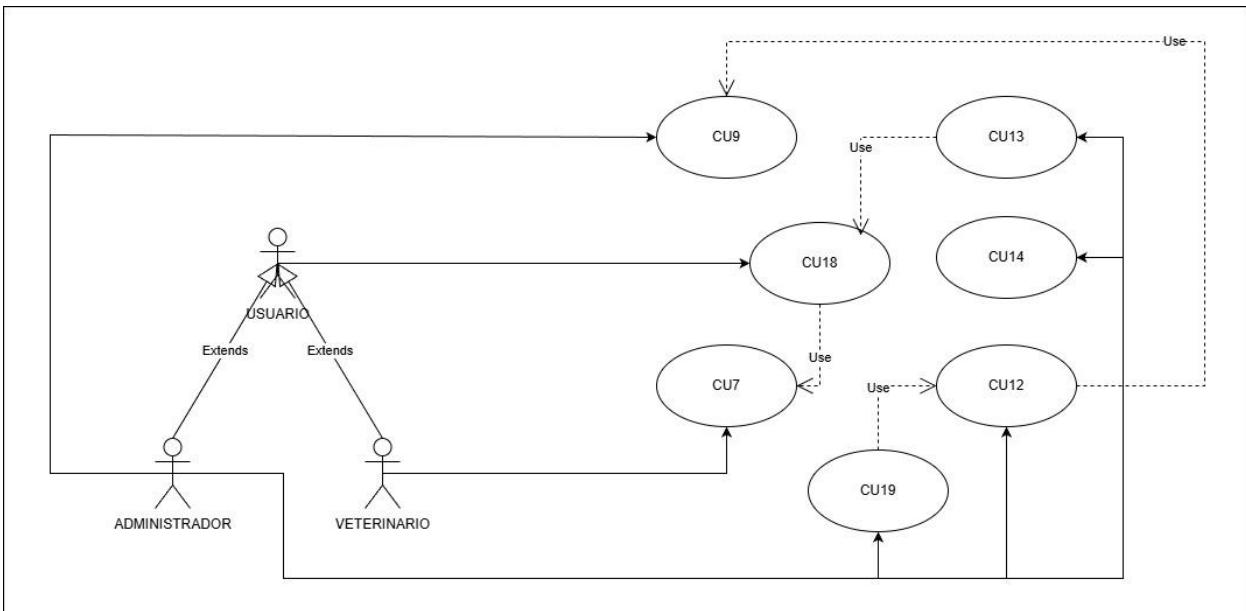
CU6.- Agendar Citas

CU8.- Gestionar Consultas y Tratamientos

CU10.- Gestionar Servicios y Precios

CU28.- Integración con Pasarela de Pago

CICLO 3.-



CU7.- Registrar Vacunas

CU9.- Gestionar inventario

CU12.- Reporte Historial Clínico

CU13.- Reporte de ventas y productos

CU14.- Reporte clientes con mayor gasto

CU18.- Control Stock

CU19.- Acceder a Dasboard Estadístico

CAPITULO 3 FLUJO DE TRABAJO: ANALISIS.

3.1.- Análisis de Arquitectura:

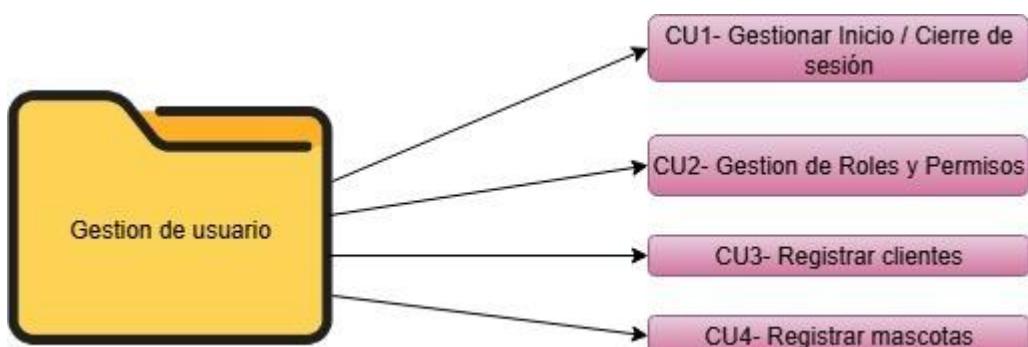
3.1.1.- Identificar Paquetes:

PAQUETE	DESCRIPCION
Gestión de Usuarios	Incluye los procesos de autenticación (inicio/cierre de sesión), administración de roles, permisos, y el registro de clientes con sus mascotas . Es la base del sistema, ya que garantiza seguridad, control de accesos y la correcta identificación de los actores que interactúan con la aplicación.
Gestión Clínica	Agrupa los casos de uso relacionados con la atención médica veterinaria: agendamiento de citas, registro y actualización de historiales clínicos, administración de consultas, tratamientos y vacunas. Representa el núcleo funcional del sistema, gestionando los servicios brindados a los pacientes y sus dueños.
Inventario	Comprende las funciones de control de medicamentos, insumos y servicios ofrecidos. Permite registrar entradas y salidas , gestionar proveedores , controlar niveles de stock y fechas de vencimiento, además de planificar reabastecimientos . Su objetivo es asegurar la disponibilidad de recursos para el funcionamiento de la clínica.
Facturación y Administración	Incluye las funcionalidades para la emisión de facturas , registro de pagos , control de caja y auditoría de transacciones. Permite llevar un control financiero confiable y transparente, asegurando la correcta gestión de ingresos y gastos de la clínica.

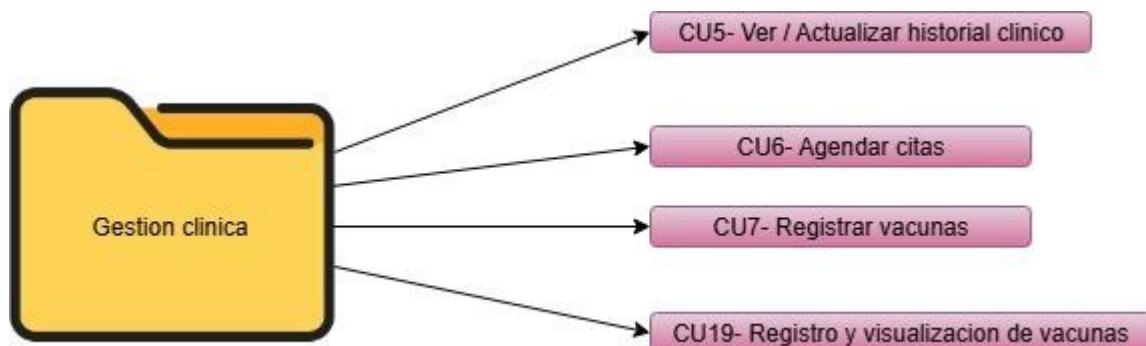
 <p>Comunicación</p>	<p>Abarca los procesos de interacción tanto interna como externa. Facilita recordatorios automáticos a los clientes sobre citas, pagos o vacunas, así como la coordinación interna mediante calendarios al personal. Su propósito es mejorar la relación clínica-cliente y mejorar la eficiencia operativa.</p>
 <p>Reportes</p>	<p>Reúne los casos de uso destinados al análisis y la generación de información estratégica. Incluye reportes de ventas, historial clínico, estadísticas de veterinarios, identificación de clientes más rentables y dashboards de indicadores clave. Este paquete aporta herramientas para la toma de decisiones gerenciales y la mejora continua.</p>
 <p>Catálogos/ Configuración</p>	<p>Se encarga de la gestión y mantenimiento de datos maestros, como especies, razas, vacunas y servicios. También abarca la configuración del sistema en aspectos como impuestos, horarios, métodos de pago e integración con pasarelas de pago. Su objetivo es brindar flexibilidad y adaptabilidad según las necesidades de la clínica.</p>

3.1.2.- Relacionar Paquetes y Casos de Uso:

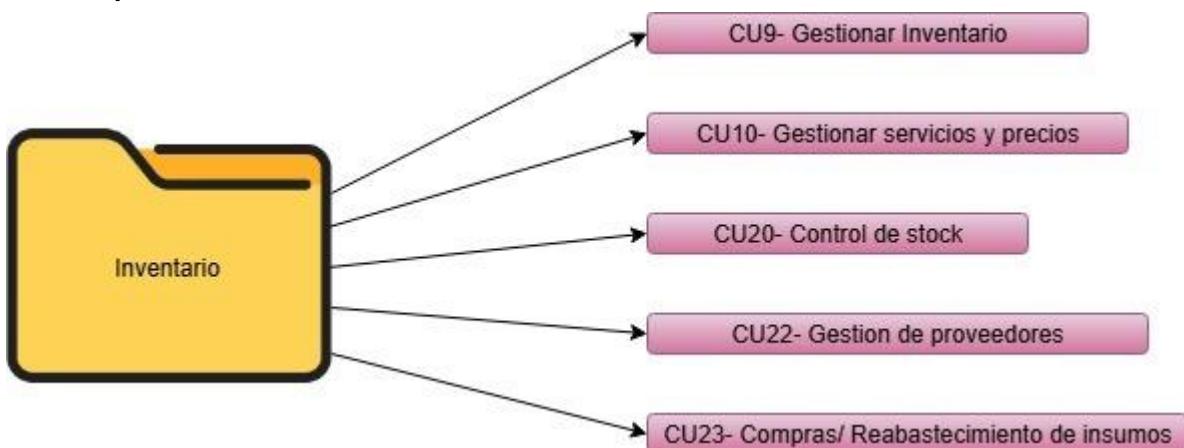
Paquete 1.- Gestión de Usuarios:



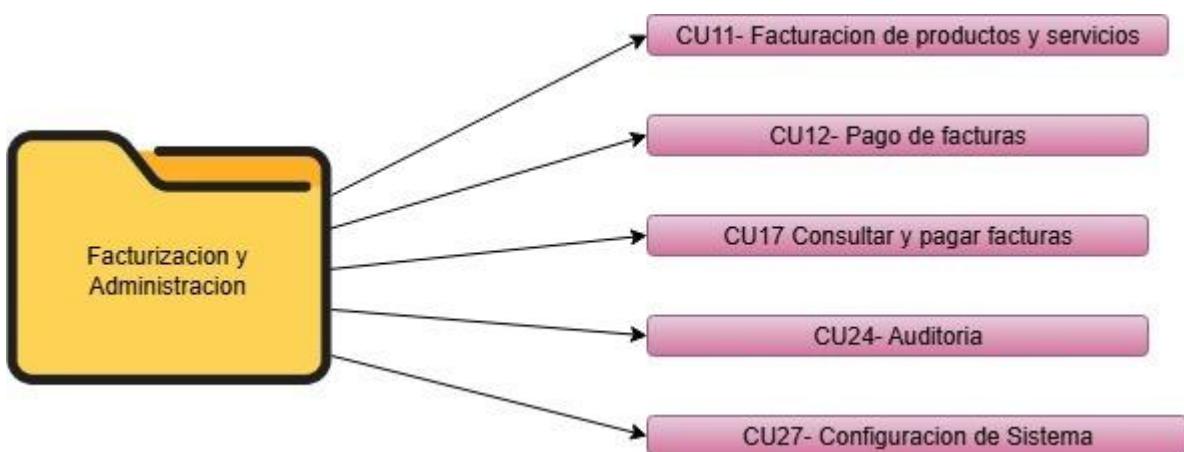
Paquete 2.- Gestión Clínica:



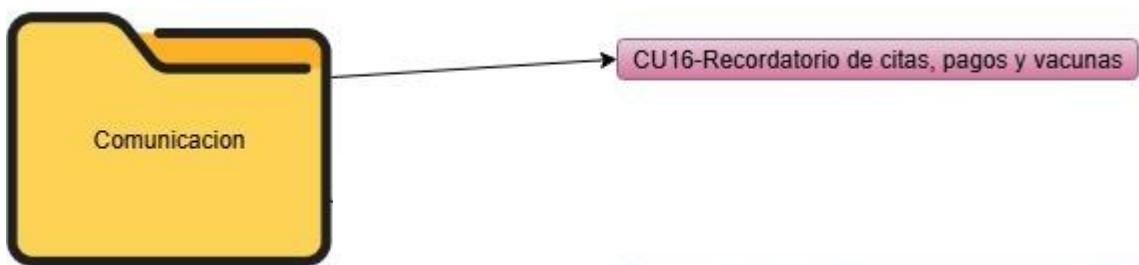
Paquete 3.- Inventario:



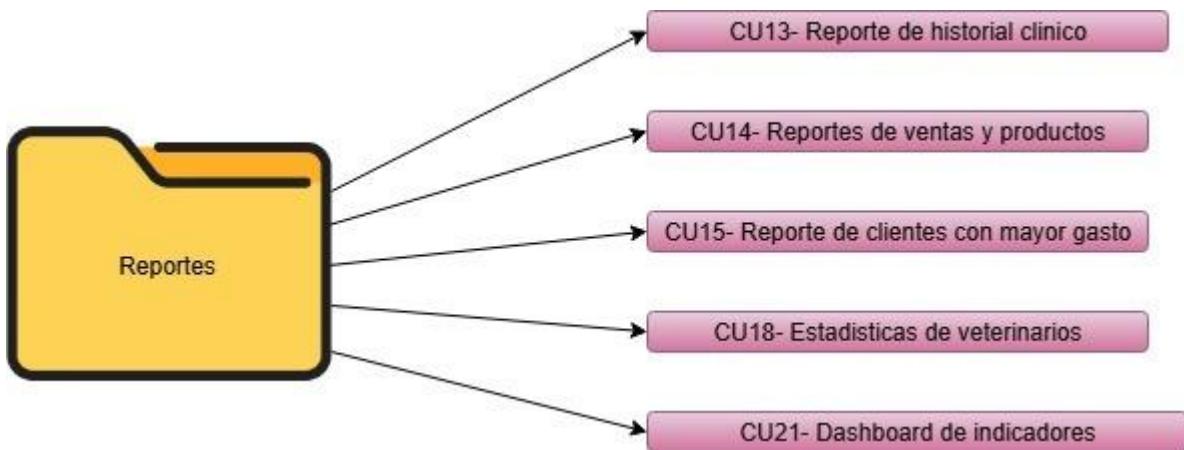
Paquete 4.- Facturación y Administración:



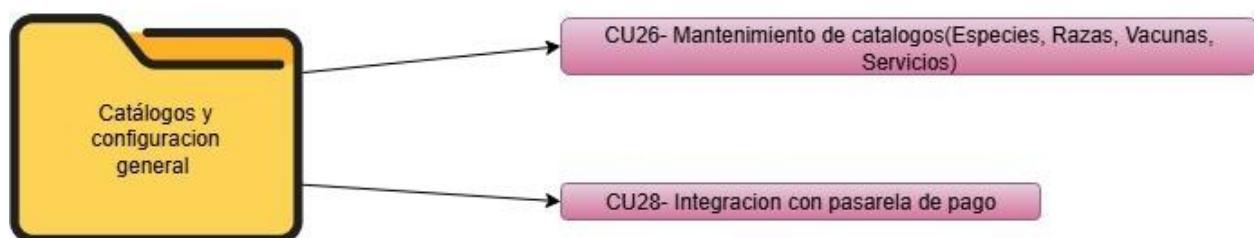
Paquete 5.- Comunicación:



Paquete 6.- Reportes:

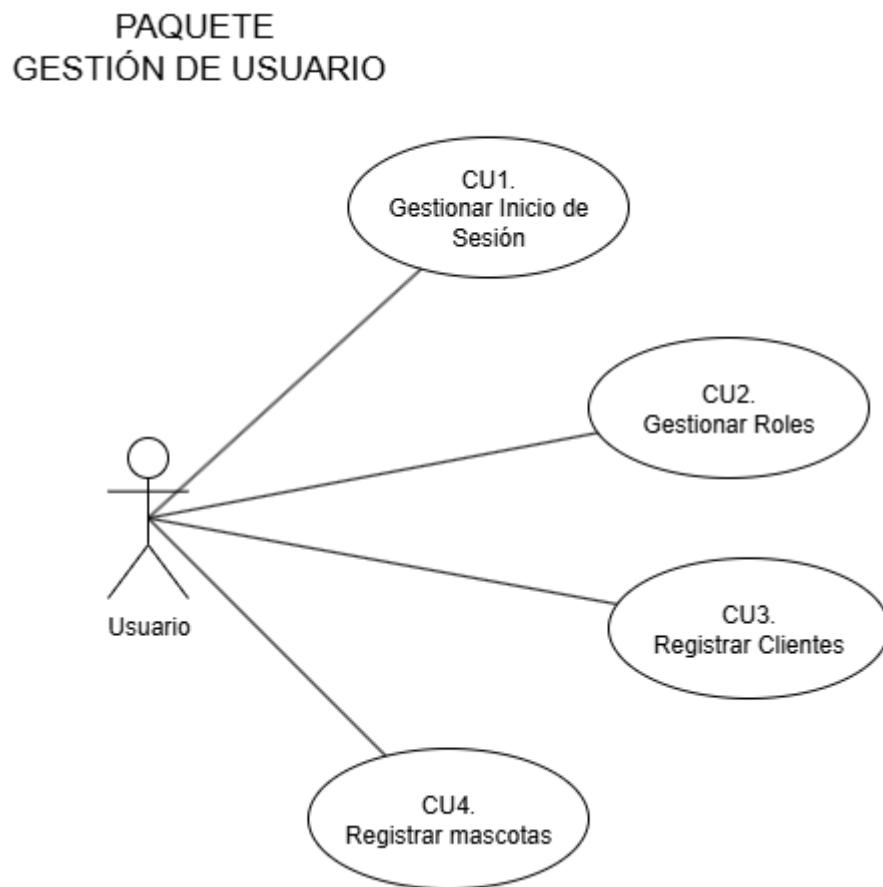


Paquete 7.- Catálogos/Configuración:

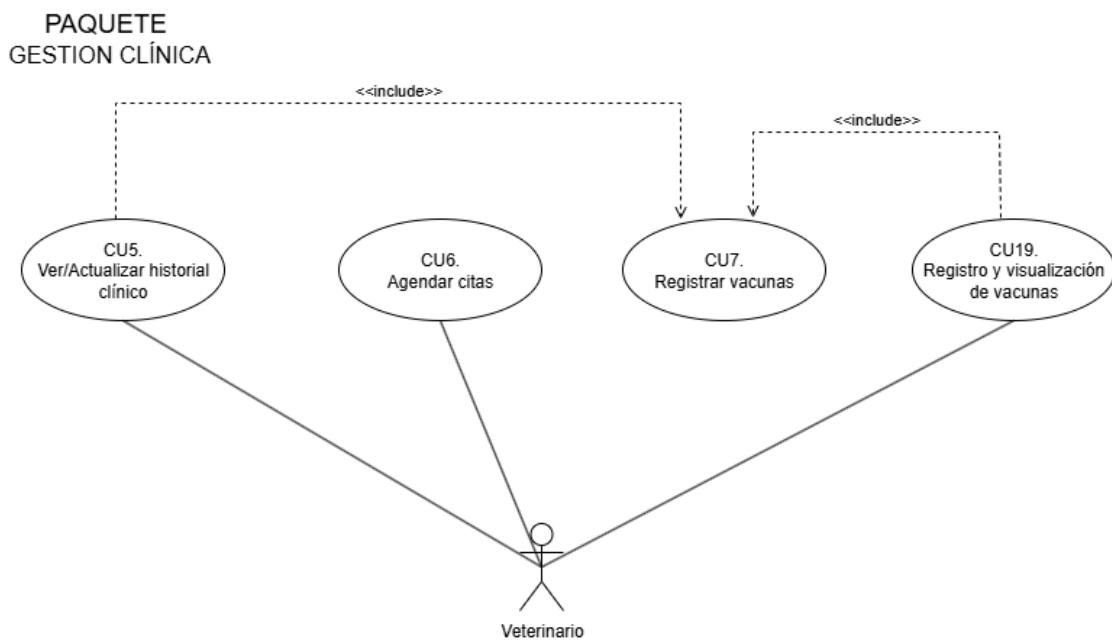


3.1.3.- Vista de Paquetes

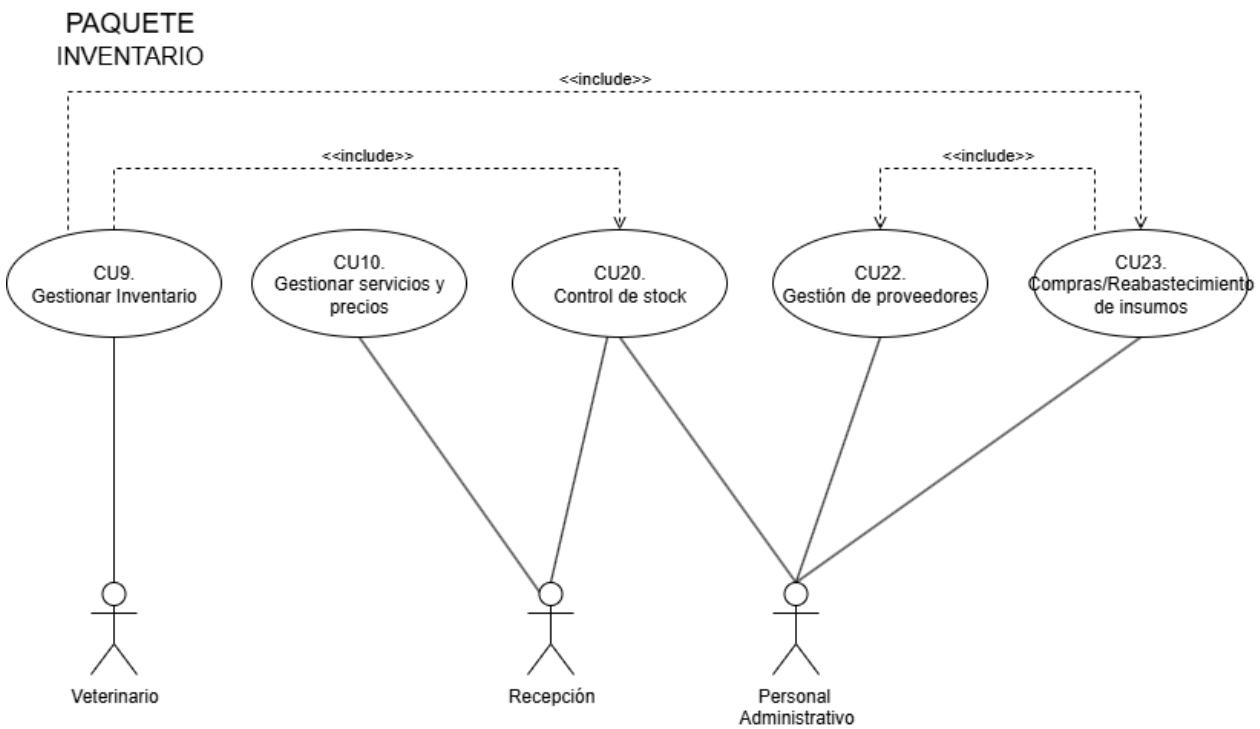
P1:



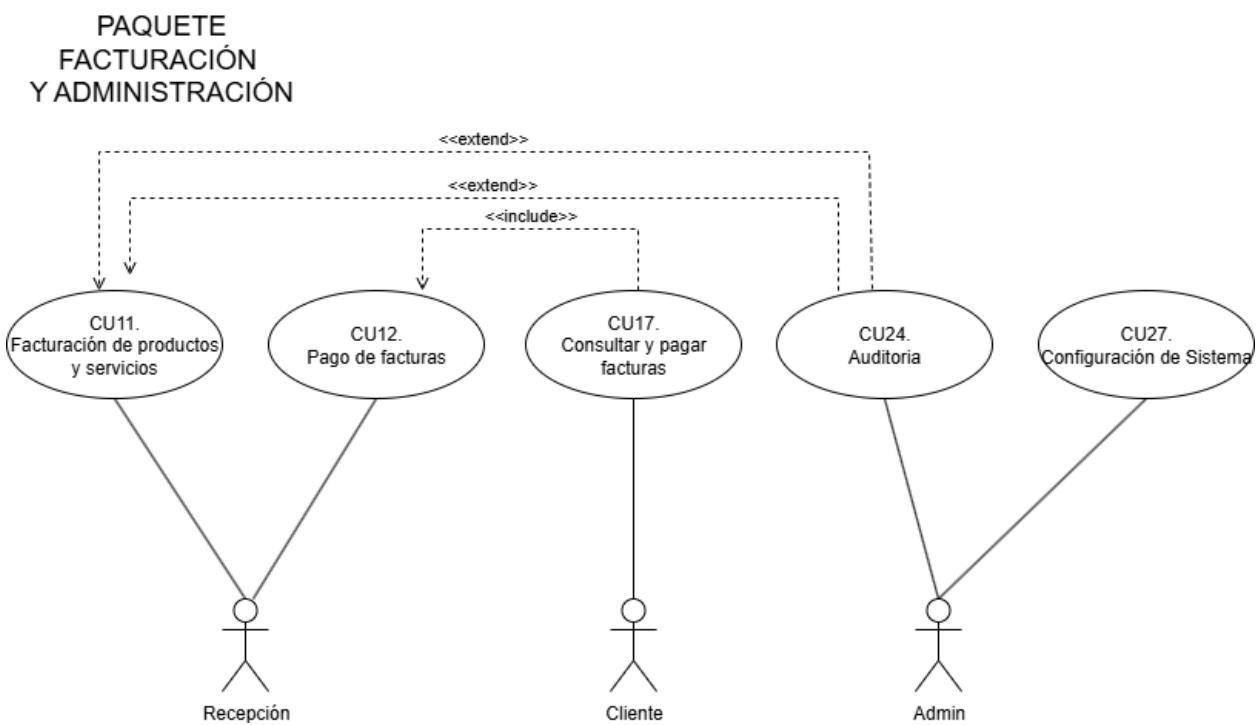
P2:



P3:

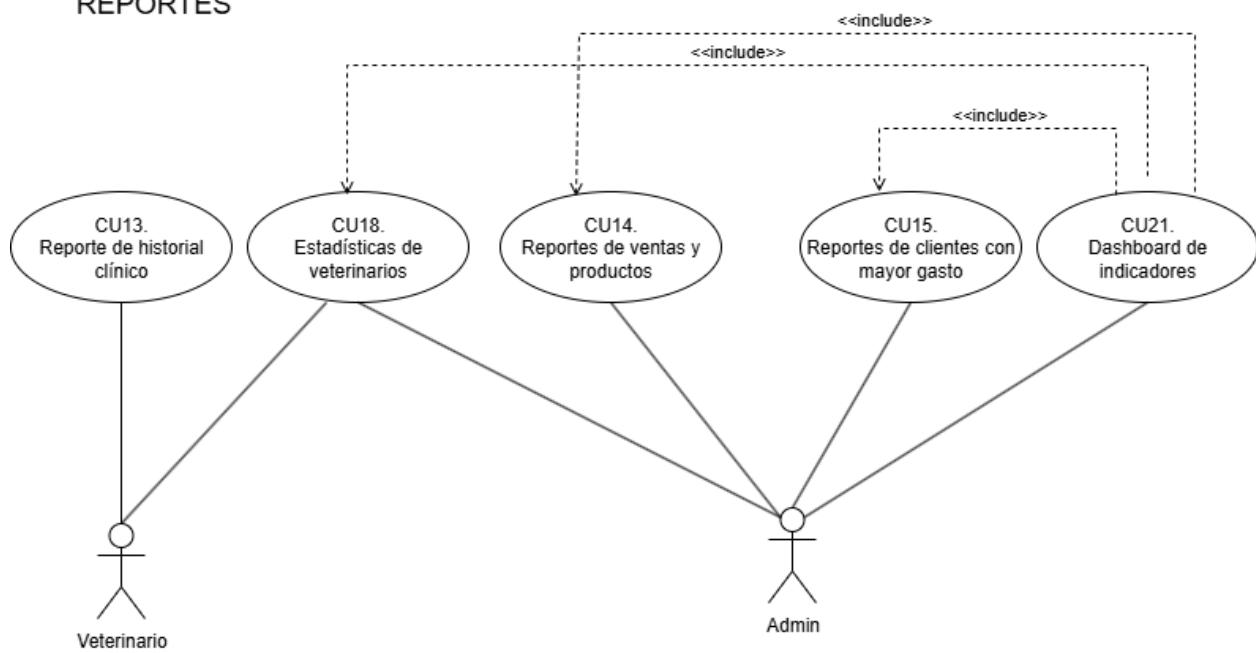


P4:



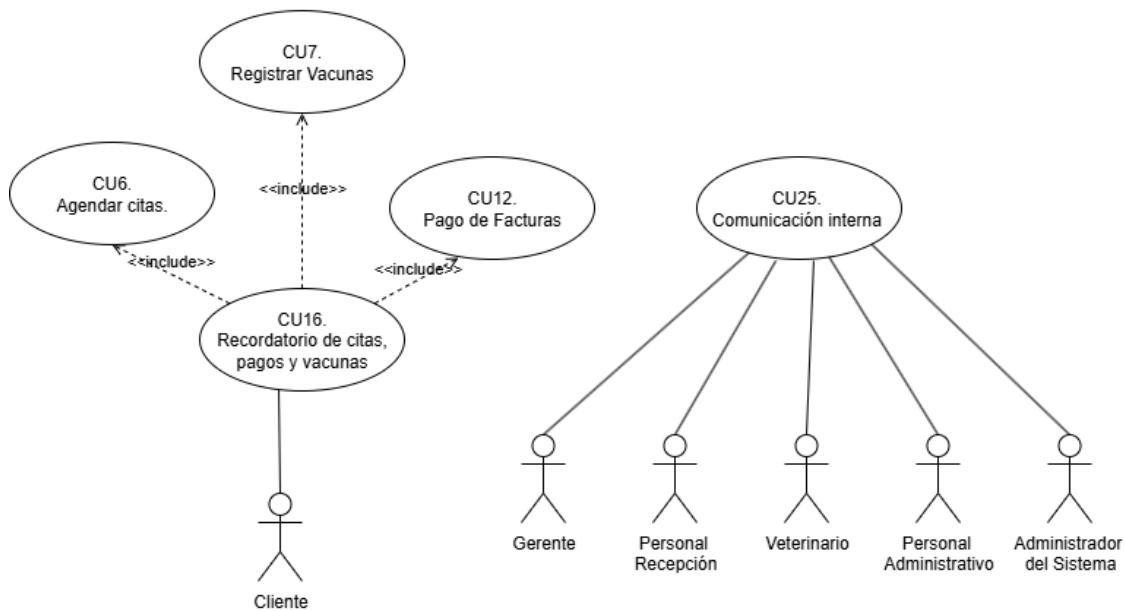
P5:

PAQUETE REPORTES



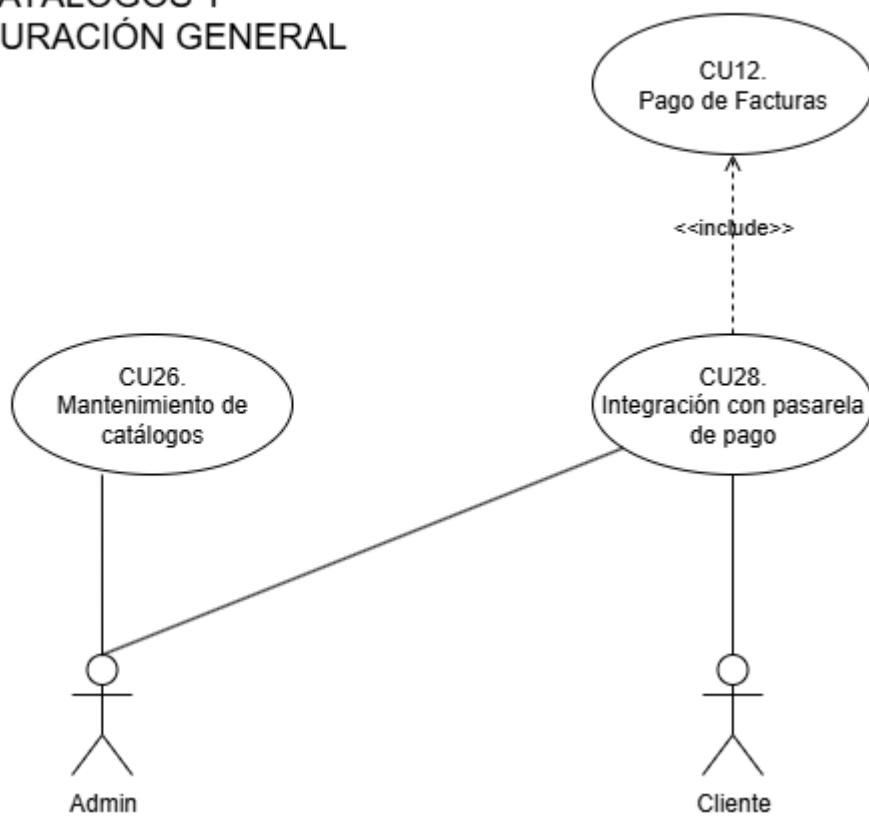
P6:

PAQUETE COMUNICACIÓN



P7:

PAQUETE
CATÁLOGOS Y
CONFIGURACIÓN GENERAL

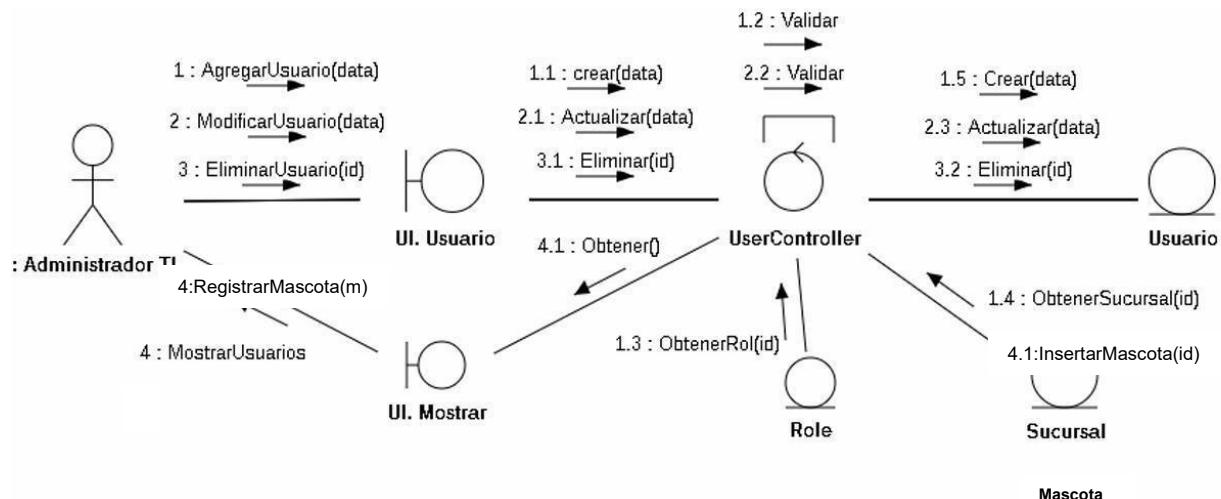


3.2.- Analizar Casos de Uso

3.2.1.- Diagrama de Comunicación

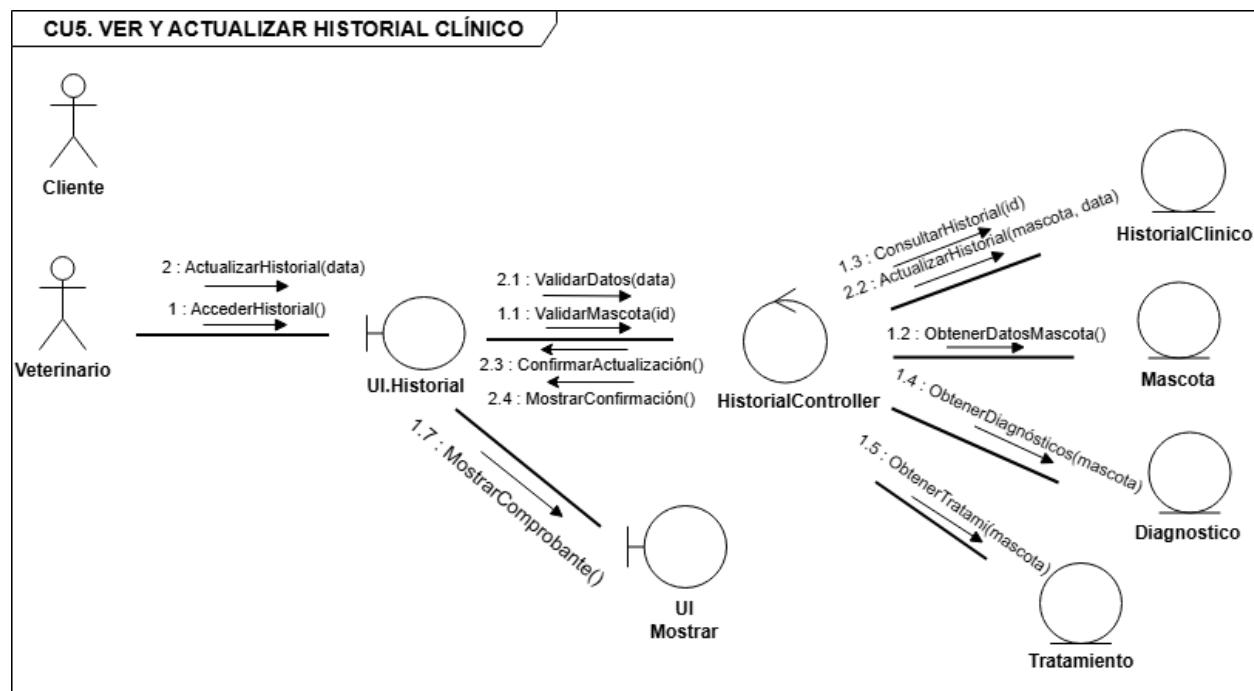
■ CICLO#1

- CU1.- Inicio/Cierre Sesión
- CU2.- Gestionar Roles y Permisos
- CU3.- Registrar Clientes
- CU4.- Registrar Mascotas

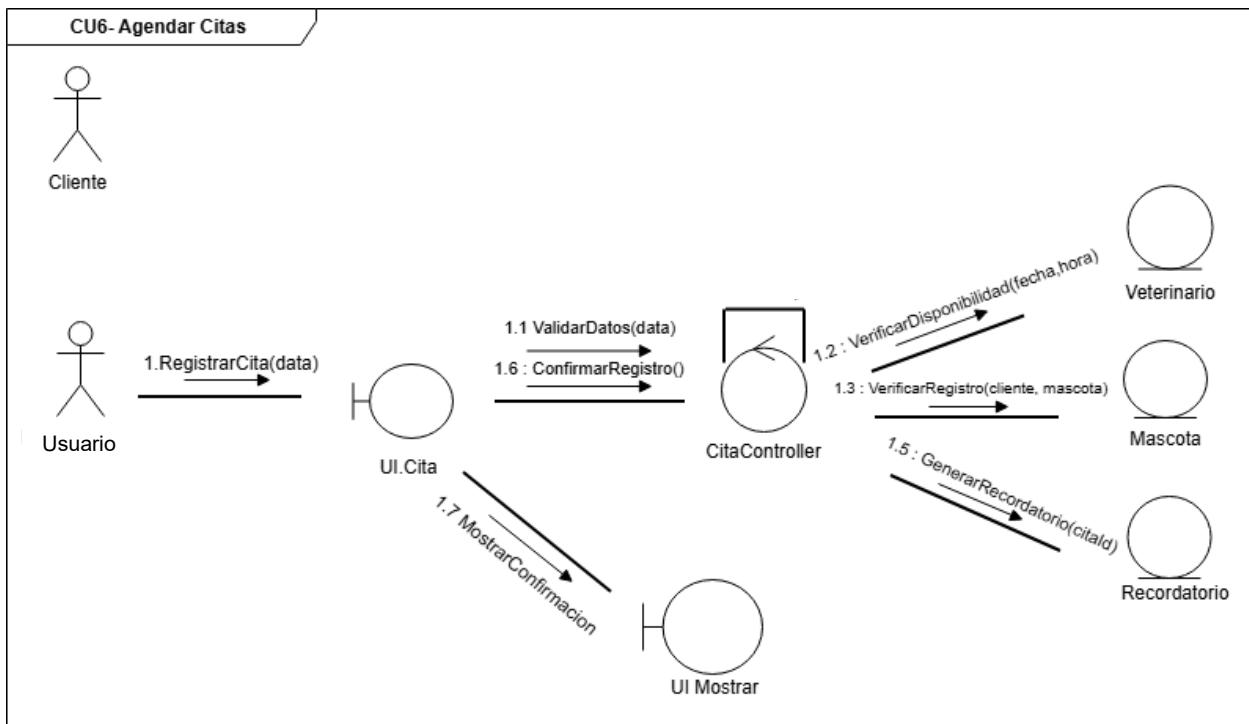


■ CICLO#2

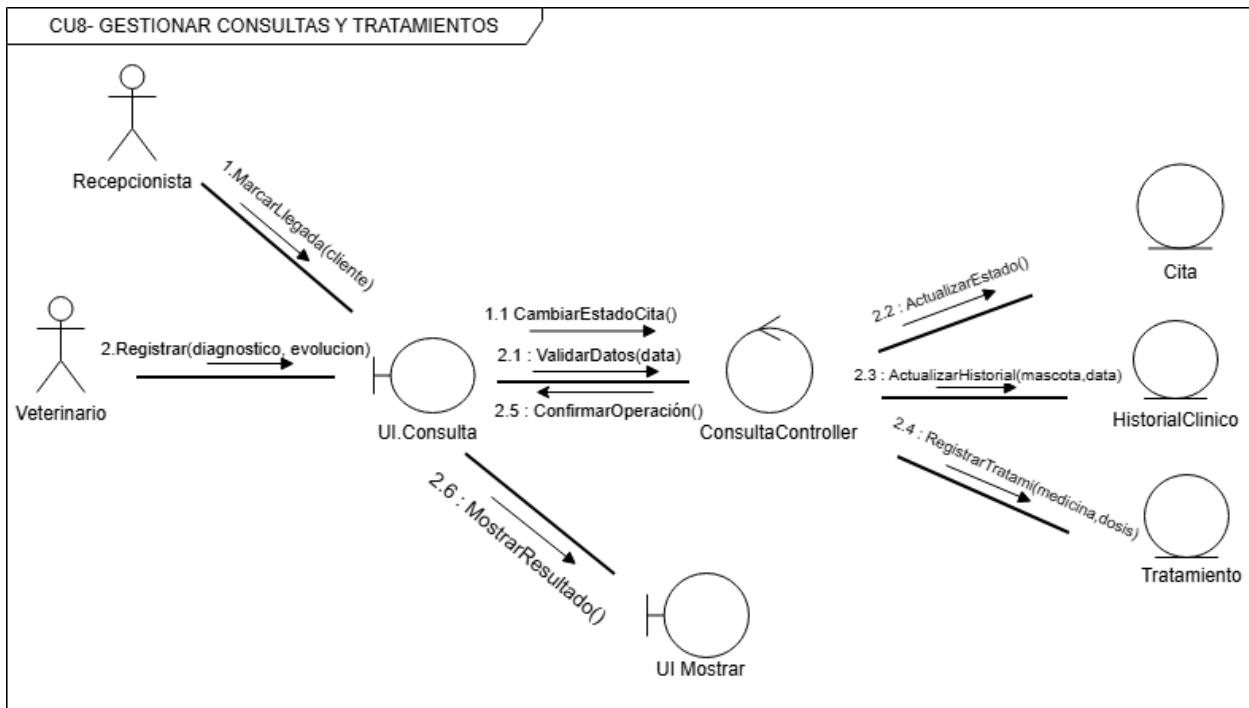
- CU5.- Ver y Actualizar Historial Clínico



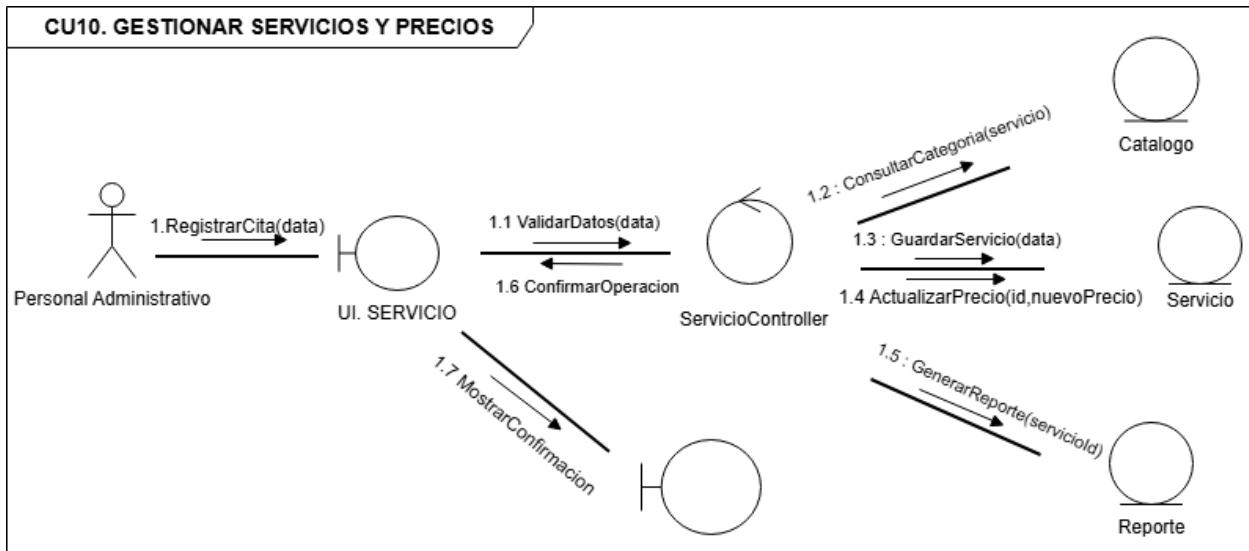
○ CU6.- Agendar Citas



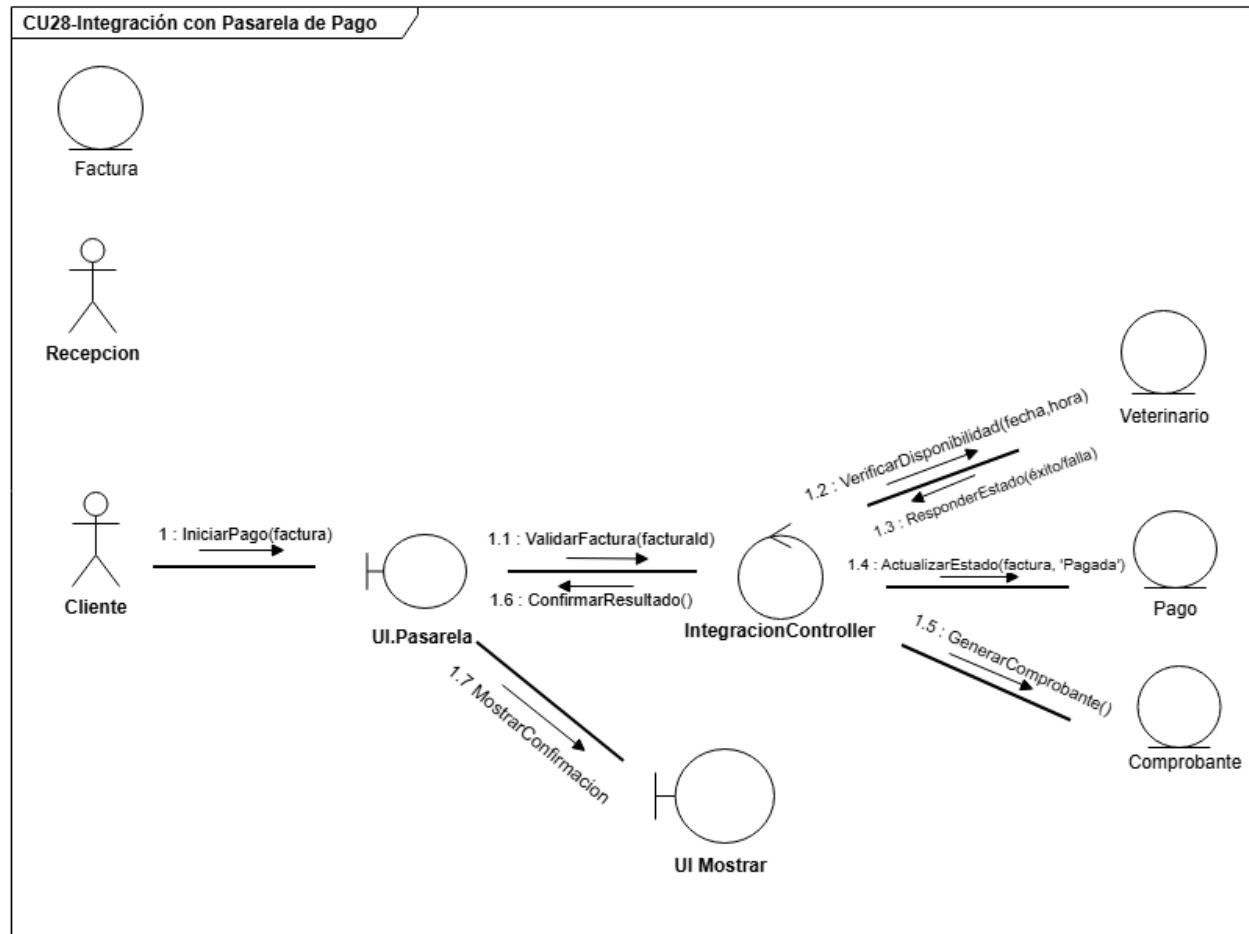
○ CU8.- Gestionar Consultas y Tratamientos



○ CU10.- Gestionar Servicios y Precios



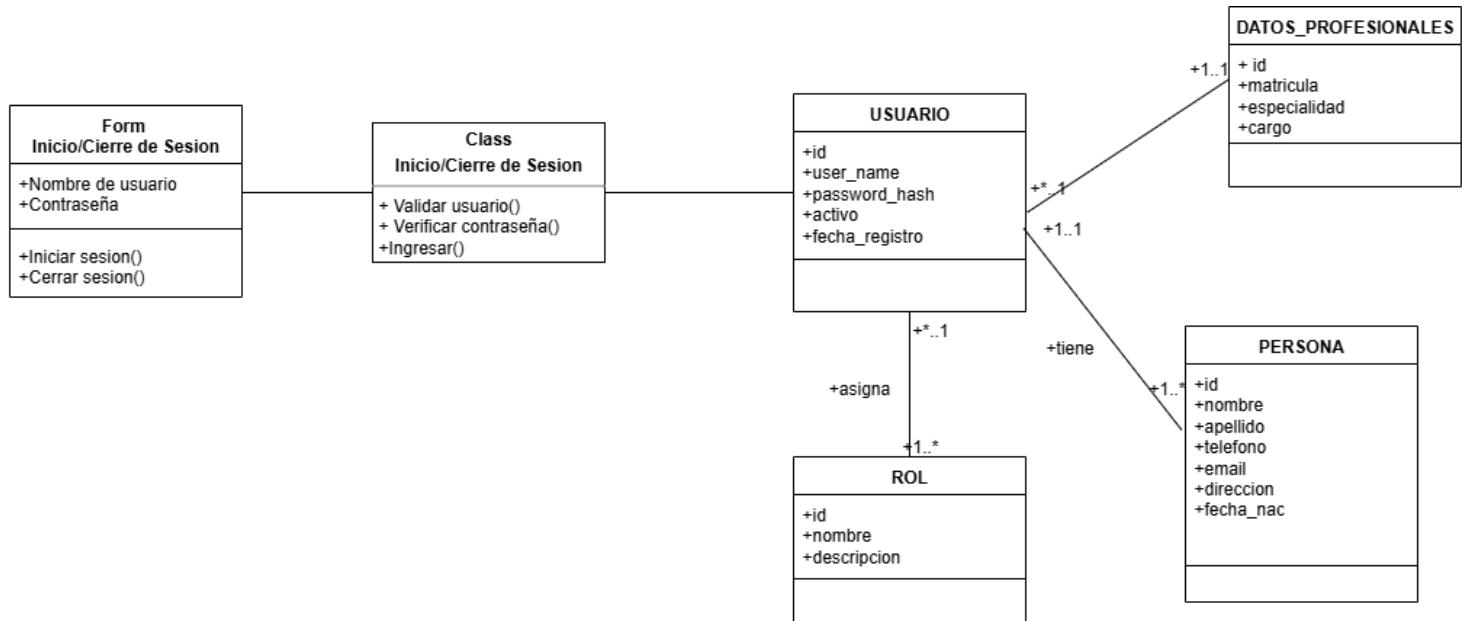
○ CU28.- Integración con Pasarela de Pago



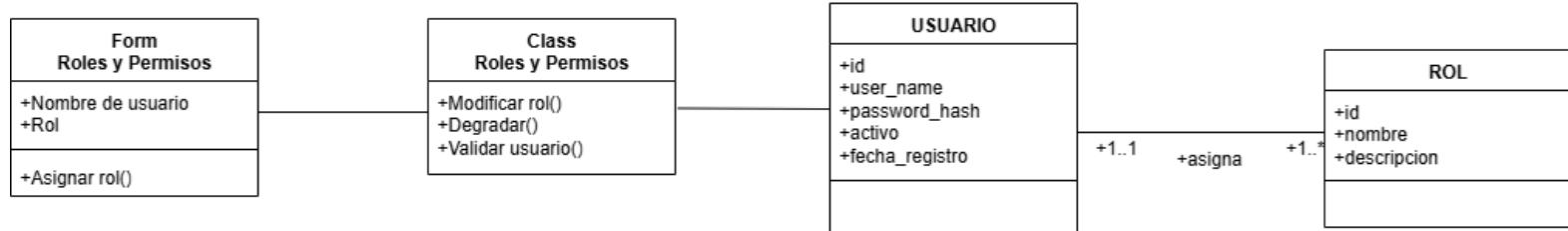
3.3.- Analizar una Clase

3.3.1.- Ciclo 1

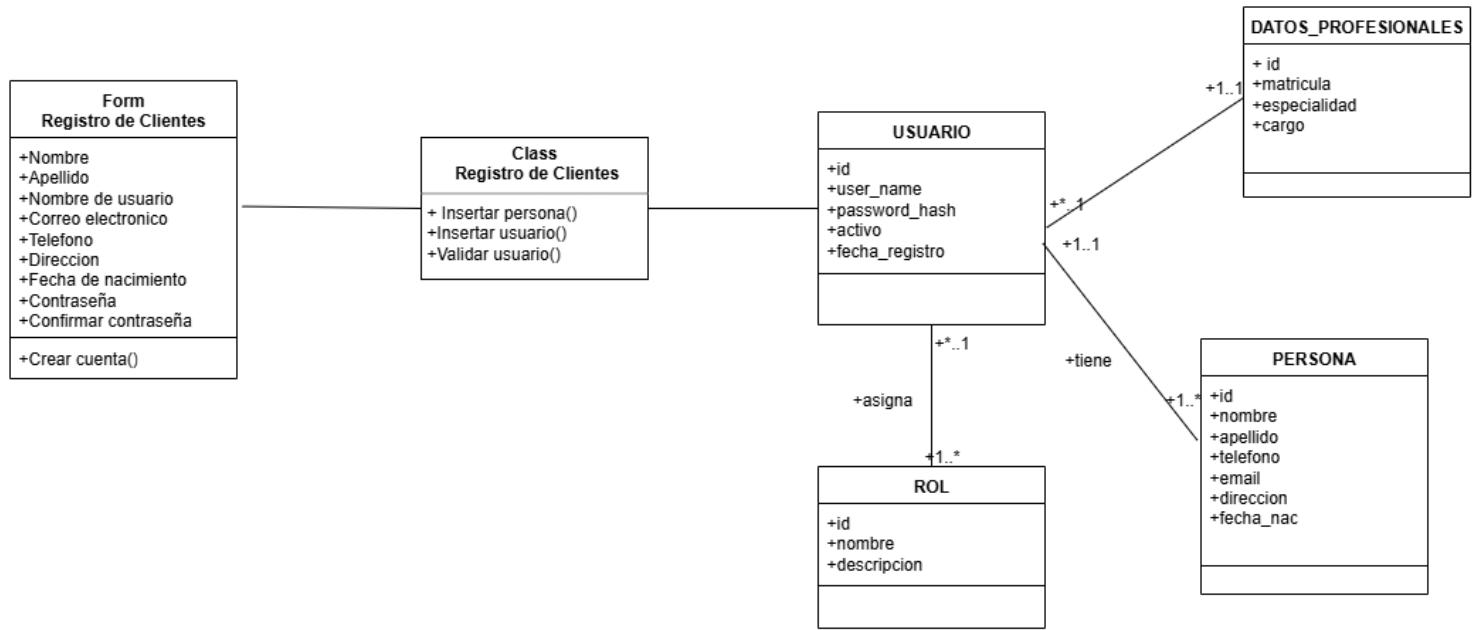
CU1: Inicio/Cierre sesión



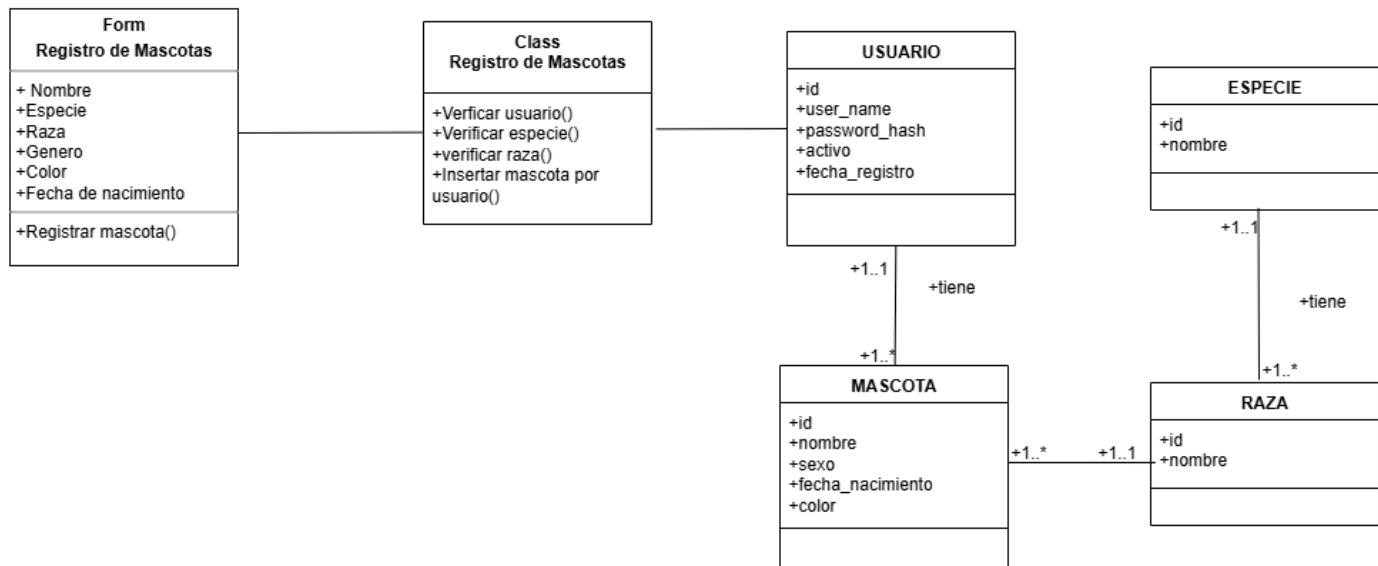
CU2: Gestionar Roles y permiso



CU3: Registro de Clientes



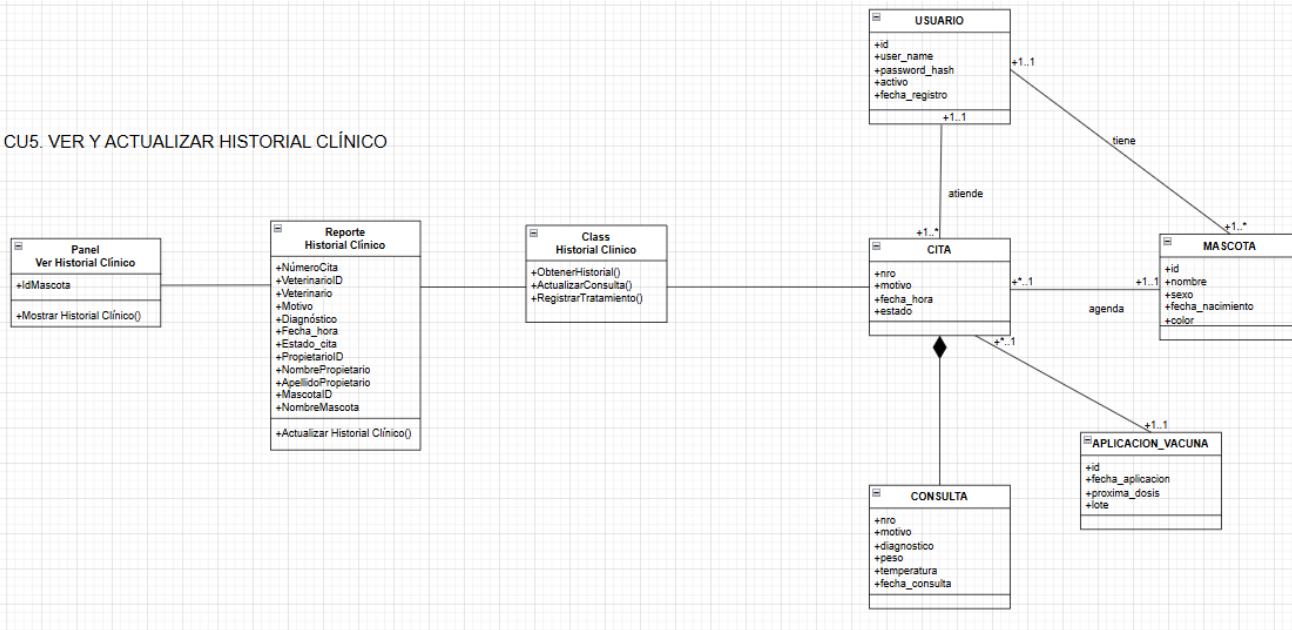
CU4: Registro de Mascotas



3.3.2.- Ciclo 2

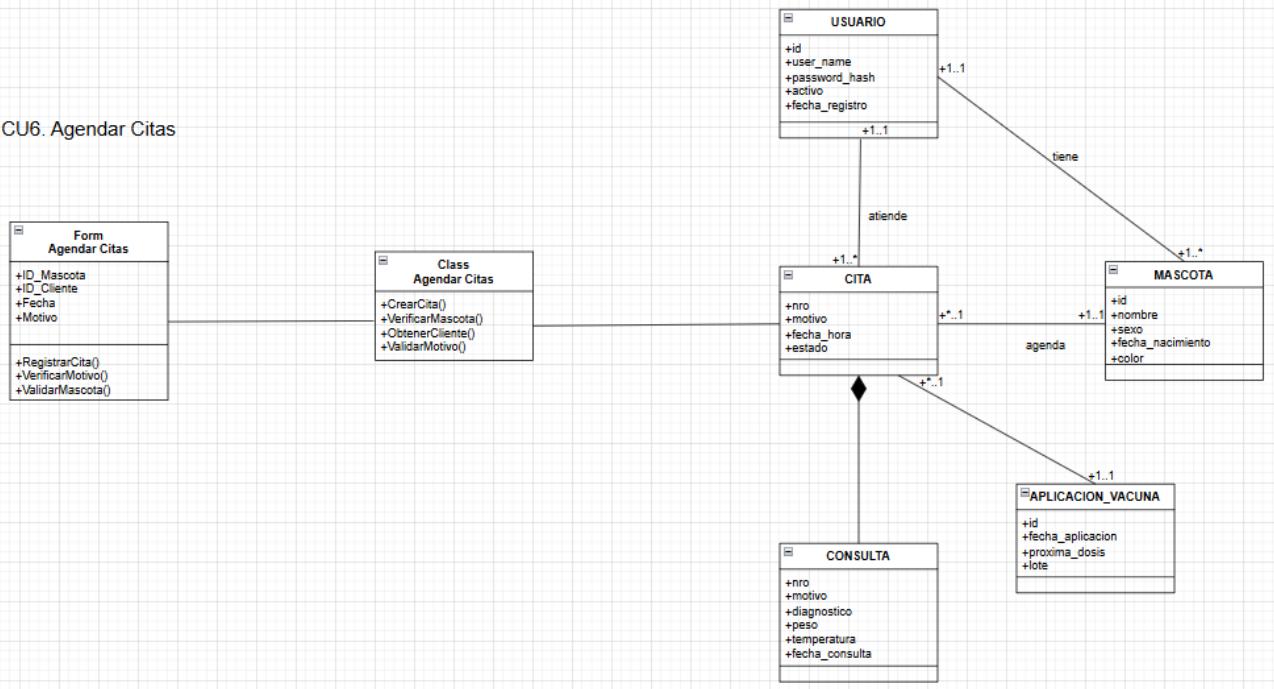
CU5.- Ver y Actualizar Historial Clínico

CU5. VER Y ACTUALIZAR HISTORIAL CLÍNICO

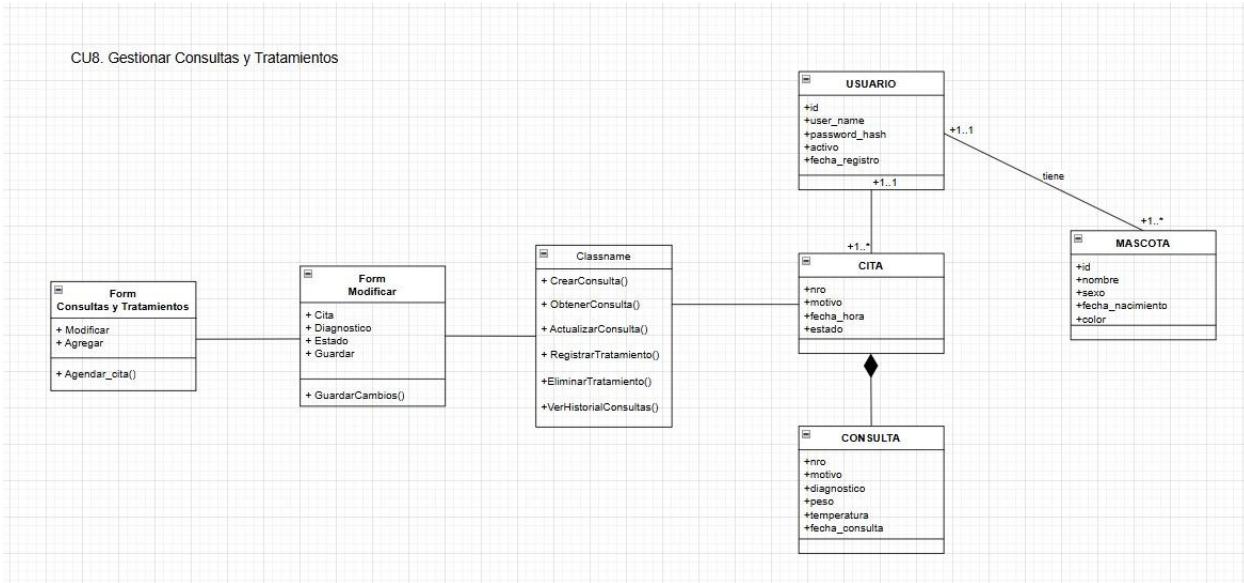


CU6.- Agendar Cita

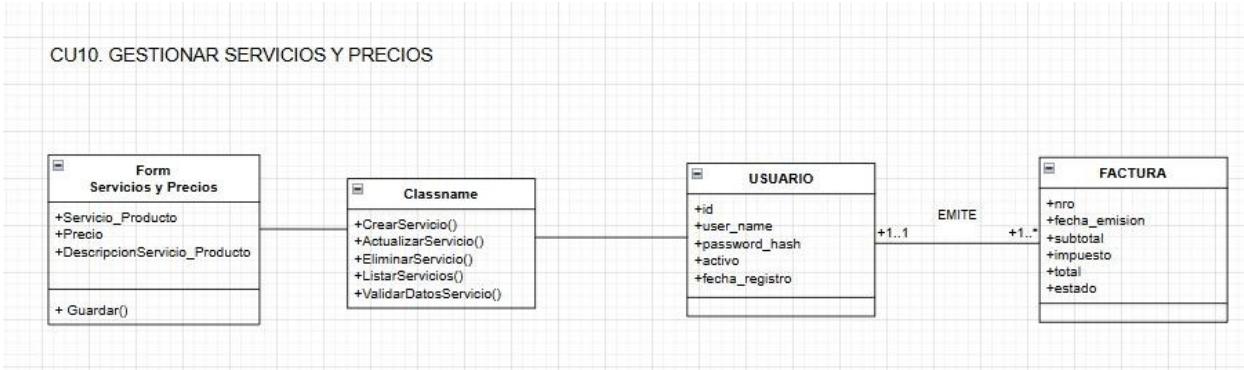
CU6. Agendar Citas



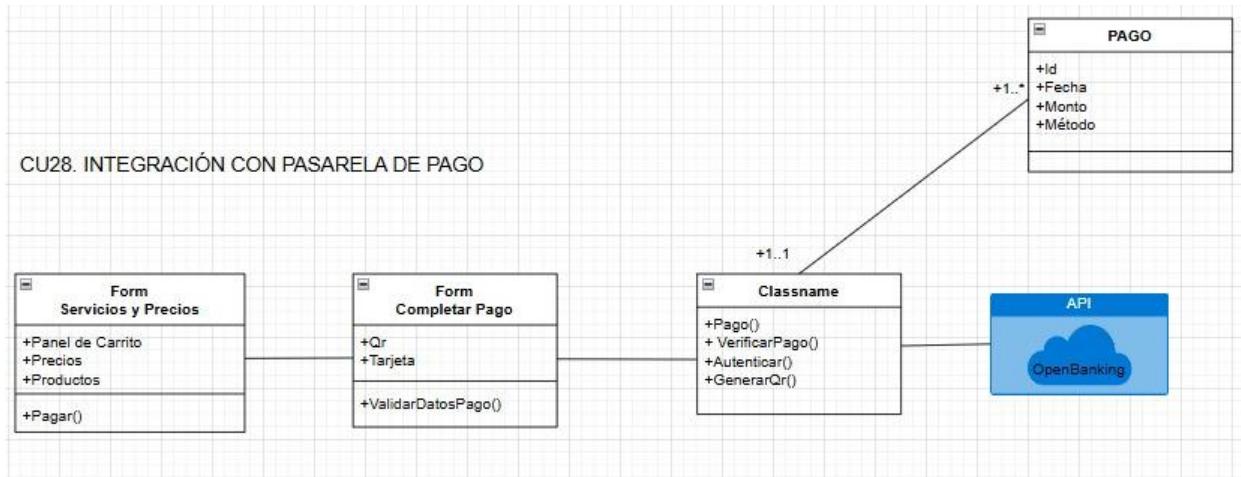
CU8.- Gestionar Consultas y Tratamientos



CU10.- Gestionar Servicios y Precios



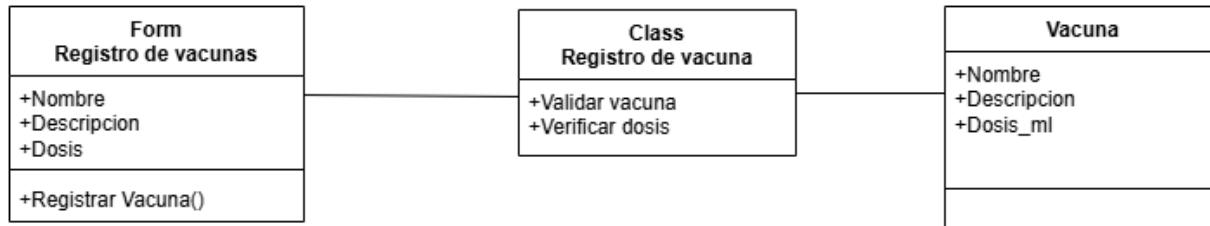
CU28.- Integración con Pasarela de Pago



3.3.3.- Ciclo 3

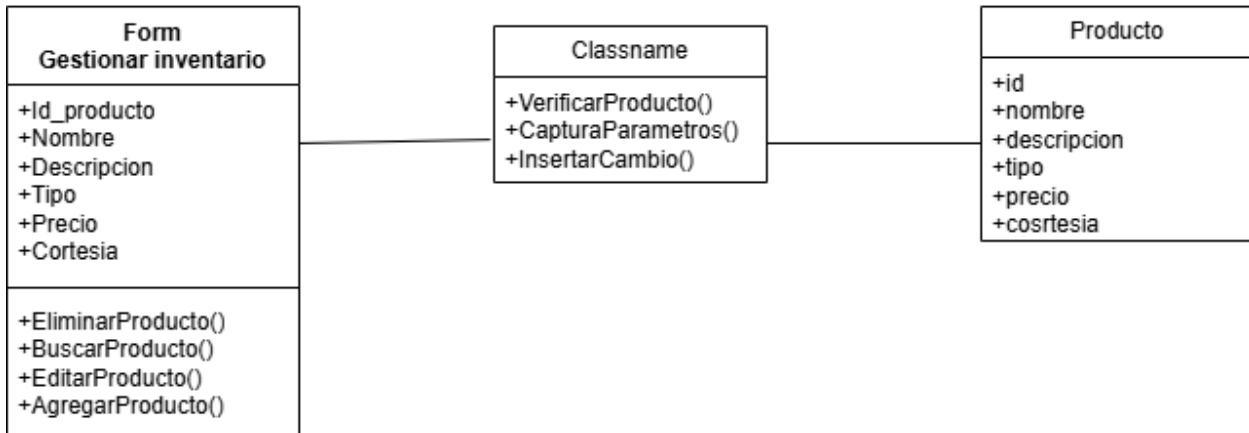
CU7: Registro de Vacuna

CU7: Registro de Vacuna



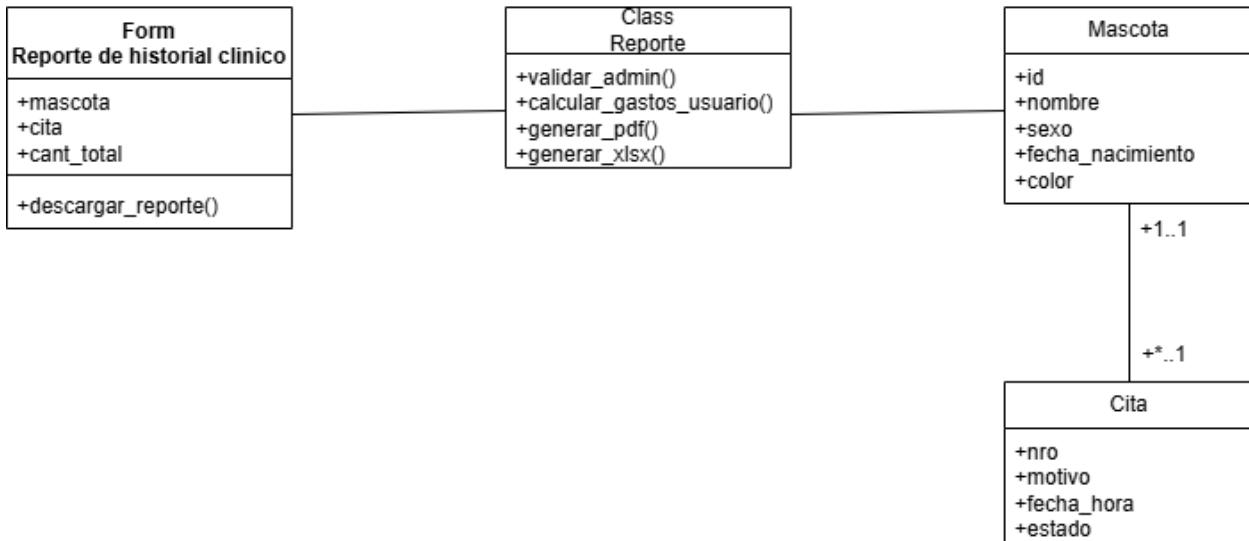
CU9: Gestionar inventario

CU9: Gestionar inventario

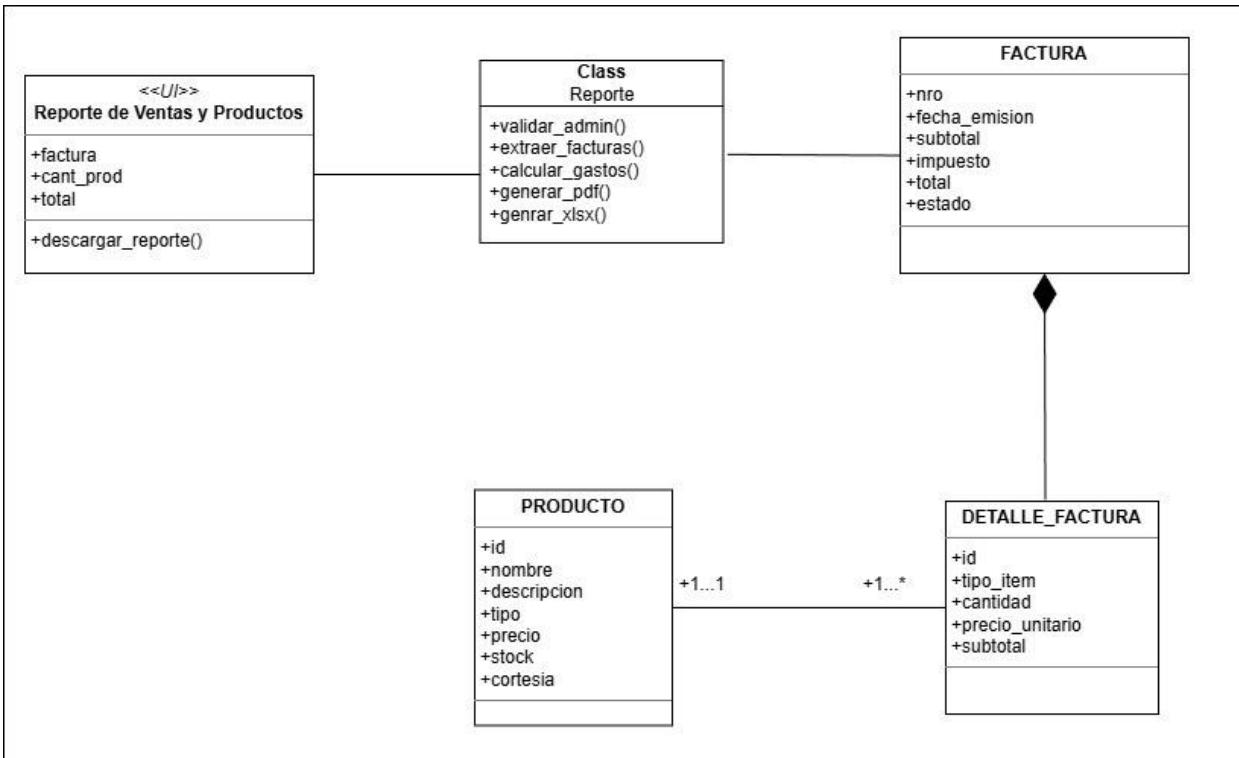


CU12: Reporte de Historial Clinico

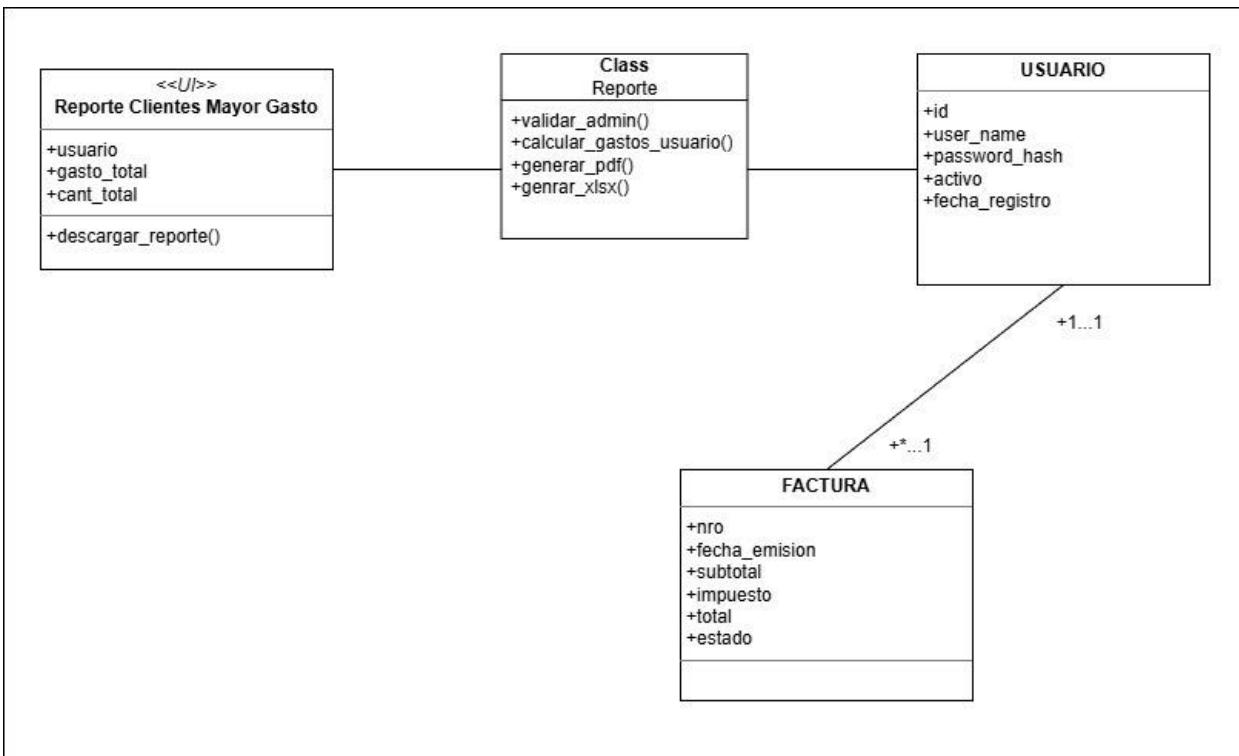
CU12: Reporte de Historial clinico



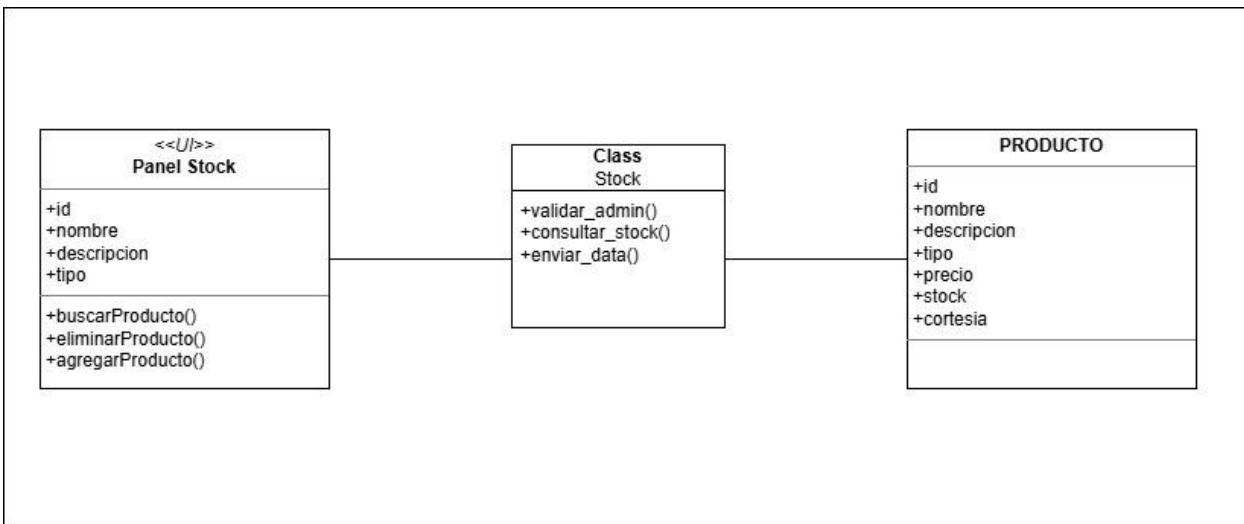
CU13: Reporte de Ventas y Productos



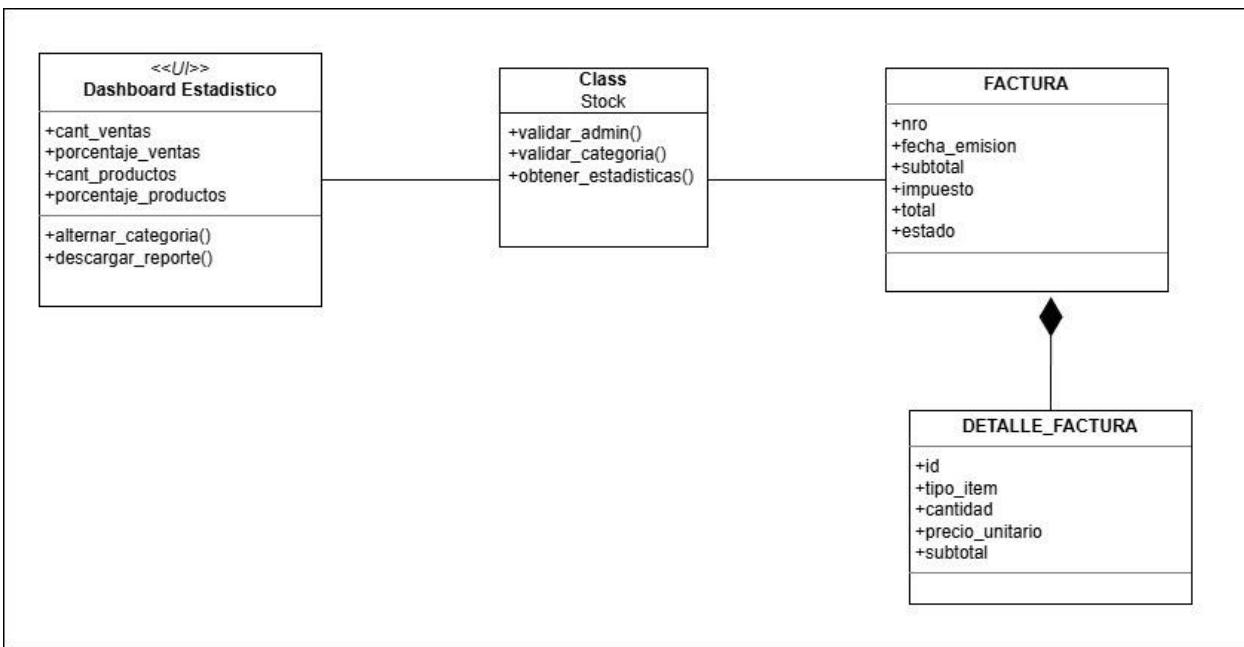
CU14: Reporte de Clientes con Mayor Gasto



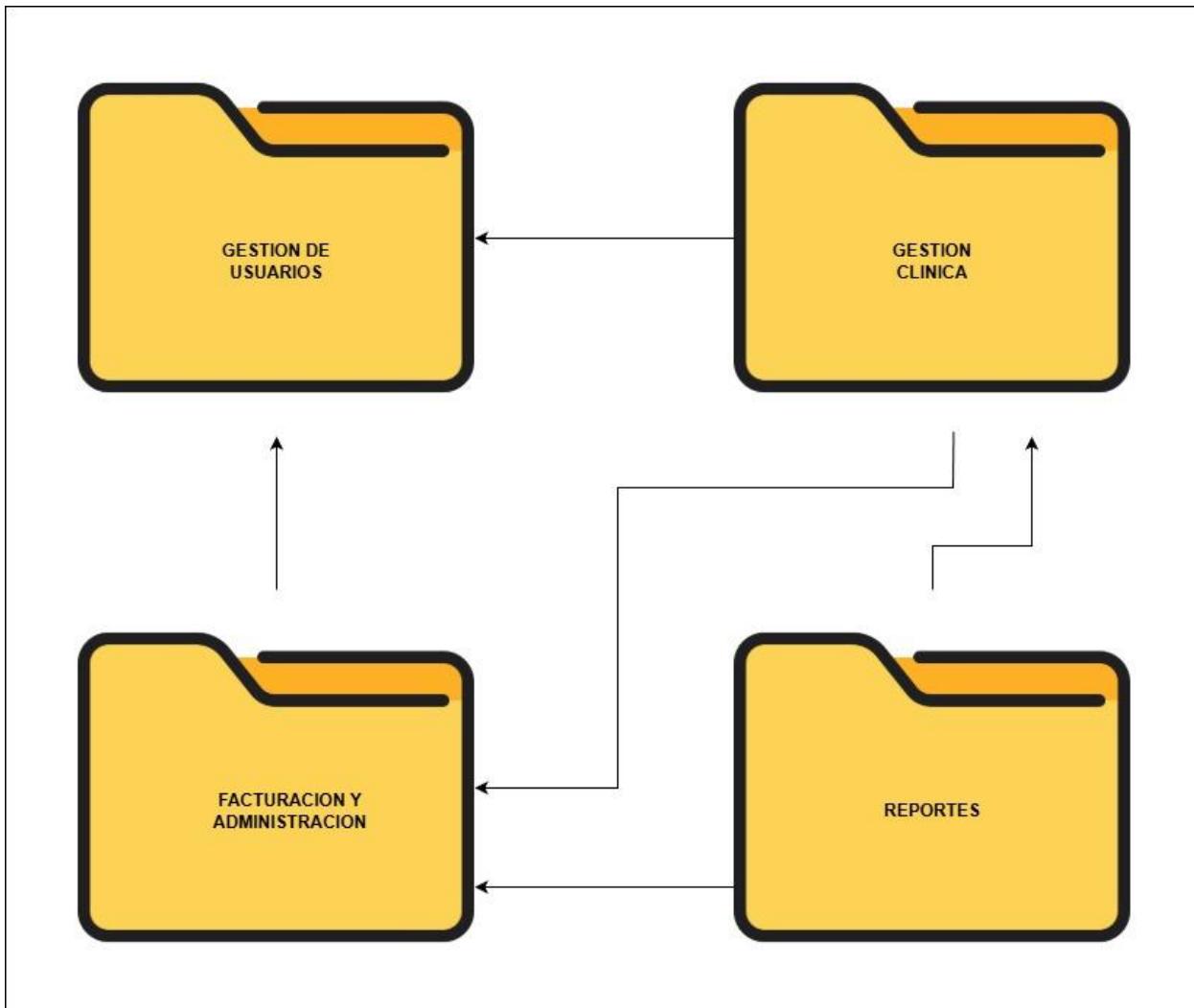
CU18: Control de Stock



CU19: Acceder al Dashboard Estadístico



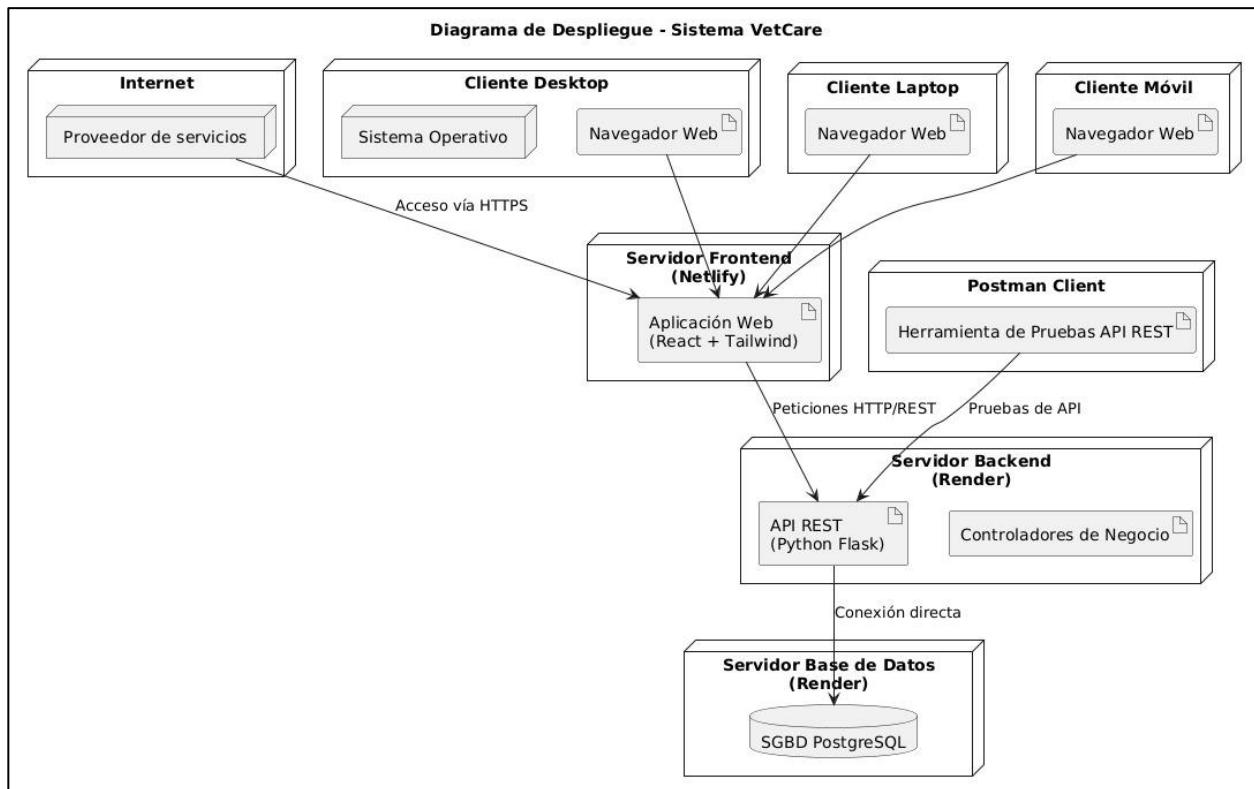
3.4.- Analizar un Paquete



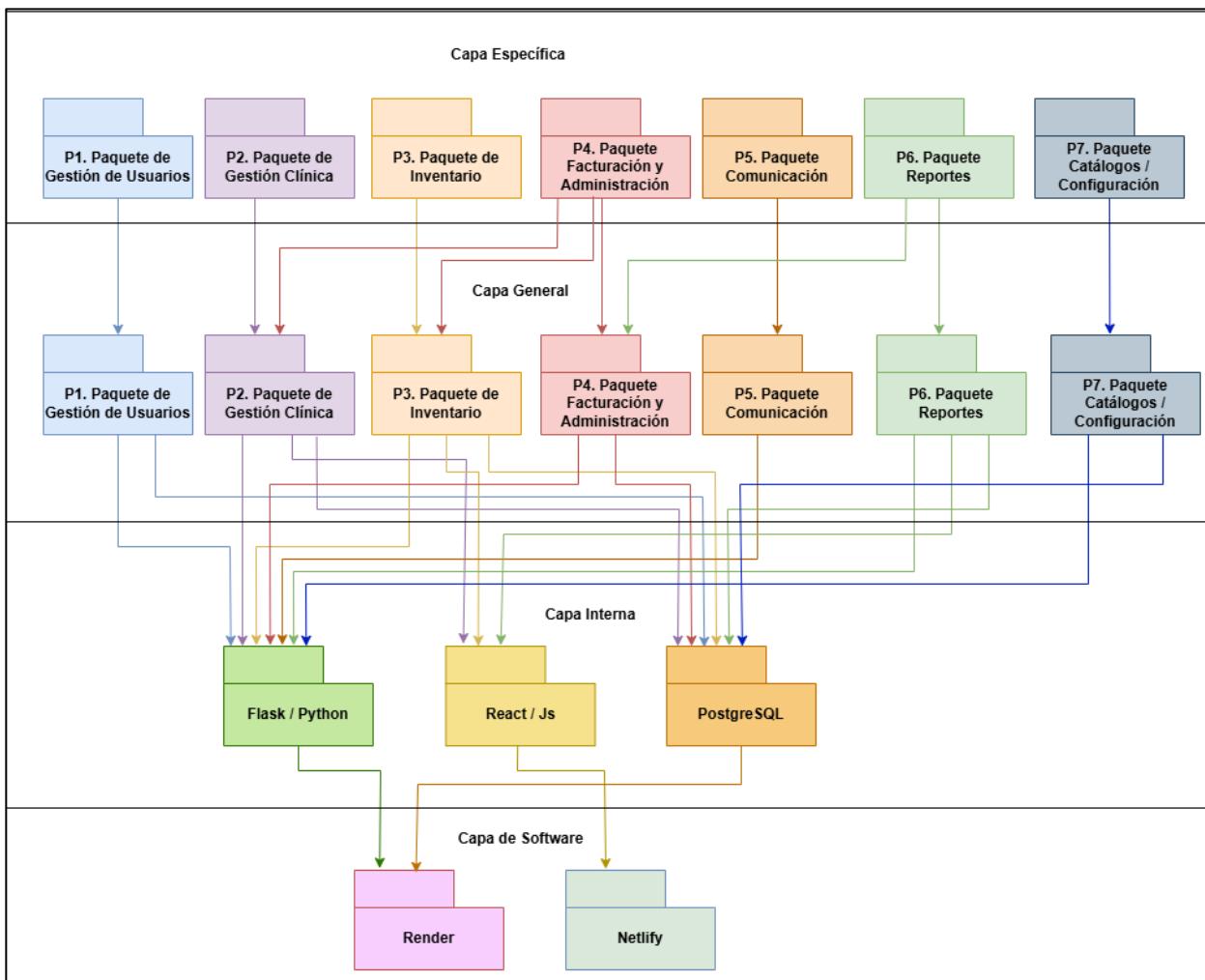
CAPITULO 4.- FLUJO DE TRABAJO: DISEÑO

4.1.- Diseño de Arquitectura

Arquitectura Física (Diagrama de Despliegue)



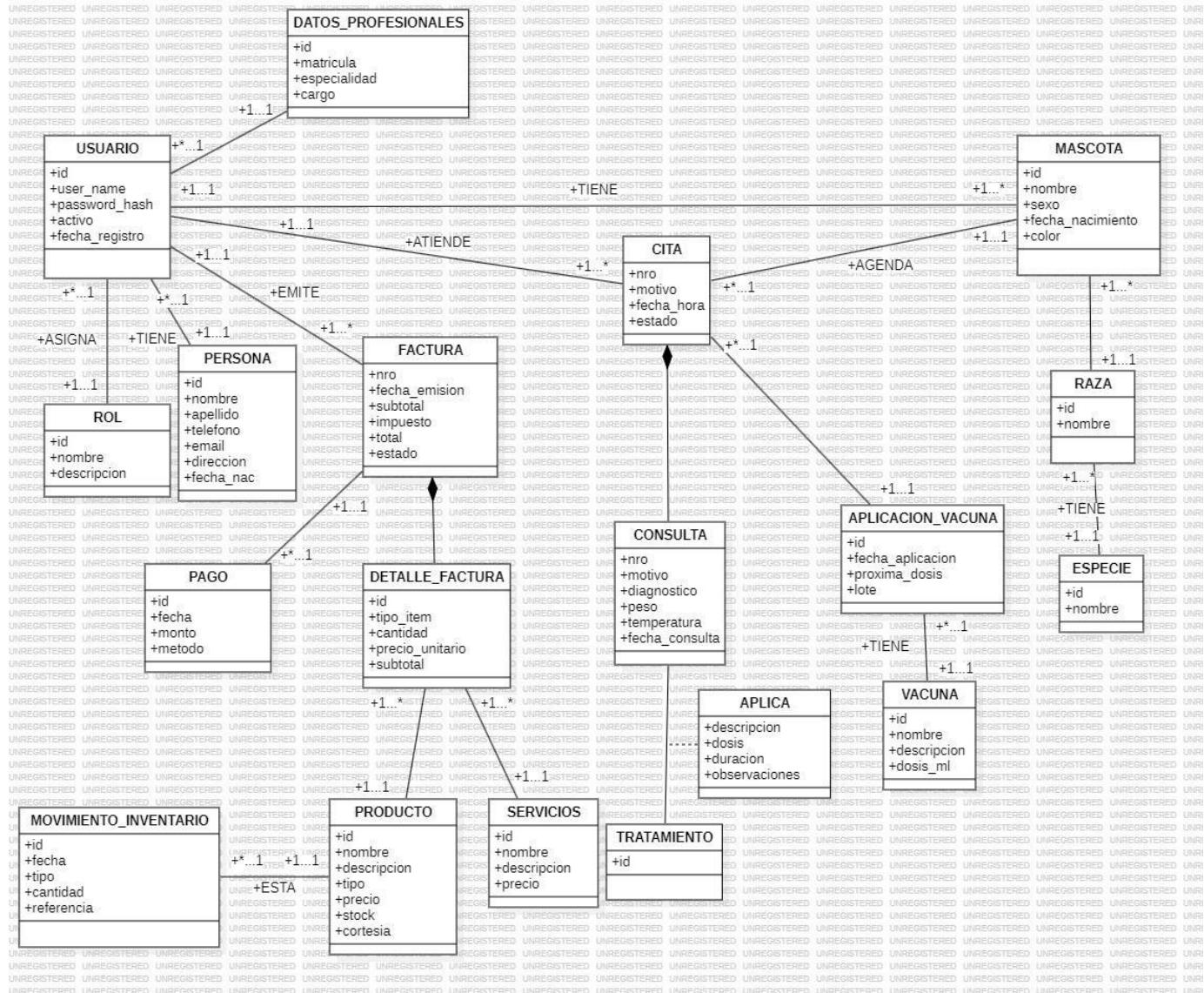
Arquitectura Lógica (Diagrama de paquete)



4.2.- Diseño de Datos

4.2.1.- Diseño de Datos Lógico

Diagrama de Clase



Mapeo

ROL	1							
pk								
id	nombre	descripcion						
PERSONA	1							
pk								
id	nombre	apellido	telefono	email	direccion	fecha_nac		
USUARIO	2							
pk					<i>fk</i>	<i>fk</i>	<i>fk</i>	
id	user_name	password	rol	activo	fecha	<i>id_persona</i>	<i>id_rol</i>	<i>id_datos_prof</i>
DATOS_PROFESIONALES	1							
pk								
id	matricula	especialidad	cargo					
FACTURA								
pk						<i>fk</i>		
nro	fecha_emision	subtotal	impuesto	total	estado	<i>id_usuario</i>		
PAGO								
pk				<i>fk</i>				
id	fecha	monto	metodo	<i>nro_factura</i>				
DETALLE_FACTURA						<i>fk</i>	<i>fk</i>	
pk								
fk						<i>id_producto</i>	<i>id_servicios</i>	
id	<i>nro_factura</i>	tipo_item	cantidad	precio_unitario	subtotal			
PRODUCTO								
pk								
id	nombre	descripcion	tipo	precio	stock	cortesia		
SERVICIOS								
pk								
id	nombre	descripcion	precio					
MOVIMIENTO_INVENTARIO						<i>fk</i>		
pk								
id	fecha	tipo	cantidad	referencia		<i>id_producto</i>		

ESPECIE	1					
pk						
id	nombre					
RAZA	2					
pk	<i>fk</i>					
id	<i>especie_id</i>	nombre				
MASCOTA	3					
pk			<i>fk</i>	<i>fk</i>		
id	nombre	sexo	fecha_nacimiento	color	<i>id_raza</i>	<i>id_usuario</i>

VACUNA	1					
pk						
id	nombre	descripcion	dosis_ml			
TRATAMIENTO		1				
pk						
id						
CITA	4					
pk			<i>fk</i>	<i>fk</i>	<i>fk</i>	
nro	motivo	fecha_hora	estado	<i>id_usuario</i>	<i>id_mascota</i>	<i>id_aplicacion_vacuna</i>
APLICACIÓN_VACUNA		2				
pk			<i>fk</i>			
id	fecha_aplicacion	proxima_dosis	lote	<i>id_vacuna</i>		
CONSULTA	5					
pk	<i>fk(PK)</i>					
nro	<i>nro_cita</i>	motivo	diagnostico	peso	temperatura	fecha_consulta
APLICA	6					
<i>fk(PK)</i>	<i>fk</i>					
<i>nro_consulta</i>	<i>id_tratamiento</i>	descripcion	dosis	duracion	observaciones	

Normalización

El sistema de información ya se encuentra en 1era, 2da, 3ra y 4ta forma normal.

4.2.2.- Diseño De Datos Físico

Tabla De Volumen

PERSONA

Atributo	Tipo de Dato	Descripción	Tamaño	Nulo	Llave
id	SERIAL	Identificador único	4 bytes	No	Primaria
nombre	VARCHAR(40)	Nombre	40 caracteres	No	
apellido	VARCHAR(40)	Apellido	40 caracteres	No	
telefono	VARCHAR(30)	Teléfono	30 caracteres	No	
email	TEXT	Email	Variable	Sí	
direccion	TEXT	Dirección	Variable	No	
fecha_nacimiento	DATE	Fecha nacimiento	4 bytes	No	

DATO_PROFESIONAL

Atributo	Tipo de Dato	Descripción	Tamaño	Nulo	Llave
id	SERIAL	Identificador único	4 bytes	No	Primaria
matricula	INTEGER	Número matrícula	4 bytes	No	
especialidad	VARCHAR(40)	Especialidad	40 caracteres	No	
cargo	VARCHAR(40)	Cargo	40 caracteres	No	

ROL

Atributo	Tipo de Dato	Descripción	Tamaño	Nulo	Llave
id	SERIAL	Identificador único	4 bytes	No	Primaria
nombre	VARCHAR(25)	Nombre rol	25 caracteres	No	
descripcion	VARCHAR(50)	Descripción	50 caracteres	Sí	

USUARIO

Atributo	Tipo de Dato	Descripción	Tamaño	Nulo	Llave
id	SERIAL	Identificador único	4 bytes	No	Primaria
persona_id	INTEGER	ID persona	4 bytes	No	Foránea (PERSONA)
user_name	VARCHAR(30)	Nombre usuario	30 caracteres	No	Única
password_hash	VARCHAR(255)	Hash contraseña	255 caracteres	No	
rol_id	INTEGER	ID rol	4 bytes	No	Foránea (ROL)
activo	BOOLEAN	Estado activo	1 byte	No	
dato_profesional_id	INTEGER	ID datos profesionales	4 bytes	No	Foránea (DATO_PROFESIONAL)

ESPECIE

Atributo	Tipo de Dato	Descripción	Tamaño	Nulo	Llave
id	SERIAL	Identificador único	4 bytes	No	Primaria
nombre	VARCHAR(40)	Nombre especie	40 caracteres	No	

RAZA

Atributo	Tipo de Dato	Descripción	Tamaño	Nulo	Llave
id	SERIAL	Identificador único	4 bytes	No	Primaria
especie_id	INTEGER	ID especie	4 bytes	No	Foránea (ESPECIE)
nombre	VARCHAR(40)	Nombre raza	40 caracteres	No	

MASCOTA

Atributo	Tipo de Dato	Descripción	Tamaño	Nulo	Llave
id	SERIAL	Identificador único	4 bytes	No	Primaria
usuario_id	INTEGER	ID dueño	4 bytes	No	Foránea (USUARIO)
raza_id	INTEGER	ID raza	4 bytes	No	Foránea (RAZA)
nombre	VARCHAR(40)	Nombre mascota	40 caracteres	No	
sexo	VARCHAR(40)	Sexo	40 caracteres	No	
fecha_nacimiento	DATE	Fecha nacimiento	4 bytes	No	

VACUNA

Atributo	Tipo de Dato	Descripción	Tamaño	Nulo	Llave
id	SERIAL	Identificador único	4 bytes	No	Primaria
nombre	VARCHAR(40)	Nombre vacuna	40 caracteres	No	
descripcion	VARCHAR(80)	Descripción	80 caracteres	No	
dosis_ml	INTEGER	Dosis en ml	4 bytes	Sí	

APLICACION_VACUNA

Atributo	Tipo de Dato	Descripción	Tamaño	Nulo	Llave
id	SERIAL	Identificador único	4 bytes	No	Primaria
vacuna_id	INTEGER	ID vacuna	4 bytes	No	Foránea (VACUNA)
fecha_aplicación	DATE	Fecha aplicación	4 bytes	No	
proxima_dosis	DATE	Próxima dosis	4 bytes	Sí	
lote	VARCHAR(60)	Número lote	60 caracteres	Sí	

CITA

Atributo	Tipo de Dato	Descripción	Tamaño	Nulo	Llave
nro	SERIAL	Número cita	4 bytes	No	Primaria
usuario_id	INTEGER	ID usuario	4 bytes	No	Foránea (USUARIO)
ap_vac_id	INTEGER	ID aplicación vacuna	4 bytes	No	Foránea (APLICACION_VACUNA)
motivo	VARCHAR(60)	Motivo cita	60 caracteres	No	
fecha_hora	TIMESTAMP	Fecha y hora	8 bytes	No	
estado	estado_cita	Estado cita	Variable	No	

CONSULTA

Atributo	Tipo de Dato	Descripción	Tamaño	Nulo	Llave
nro	SERIAL	Número consulta	4 bytes	No	Primaria
nro_cita	INTEGER	Número cita	4 bytes	No	Foránea (CITA)
motivo	VARCHAR(60)	Motivo consulta	60 caracteres	No	
diagnostico	VARCHAR(60)	Diagnóstico	60 caracteres	Sí	
peso	FLOAT	Peso mascota	8 bytes	No	
temperatura	FLOAT	Temperatura	8 bytes	No	
fecha_consulta	DATE	Fecha consulta	4 bytes	No	

TRATAMIENTO

Atributo	Tipo de Dato	Descripción	Tamaño	Nulo	Llave
id	SERIAL	Identificador único	4 bytes	No	Primaria

APLICA (Tabla intermedia)

Atributo	Tipo de Dato	Descripción	Tamaño	Nulo	Llave
nro_consulta	INTEGER	Número consulta	4 bytes	No	Primaria, Foránea (CONSULTA)
tratamiento_id	INTEGER	ID tratamiento	4 bytes	No	Primaria, Foránea (TRATAMIENTO)
descripcion	VARCHAR(60)	Descripción	60 caracteres	Sí	
dosis	FLOAT	Dosis	8 bytes	No	
duracion	TIME	Duración	8 bytes	No	
observaciones	VARCHAR(60)	Observaciones	60 caracteres	Sí	

SERVICIO

Atributo	Tipo de Dato	Descripción	Tamaño	Nulo	Llave
id	SERIAL	Identificador único	4 bytes	No	Primaria
nombre	VARCHAR(40)	Nombre servicio	40 caracteres	No	
descripcion	VARCHAR(60)	Descripción	60 caracteres	Sí	
precio	FLOAT	Precio	8 bytes	No	

PRODUCTO

Atributo	Tipo de Dato	Descripción	Tamaño	Nulo	Llave
id	SERIAL	Identificador único	4 bytes	No	Primaria
nombre	VARCHAR(40)	Nombre producto	40 caracteres	No	
descripcion	VARCHAR(60)	Descripción	60 caracteres	Sí	
tipo	VARCHAR(40)	Tipo producto	40 caracteres	No	
precio	FLOAT	Precio	8 bytes	No	
stock	INTEGER	Stock	4 bytes	No	
cortesia	BOOLEAN	Es cortesía	1 byte	No	

MOVIMIENTO_INVENTARIO

Atributo	Tipo de Dato	Descripción	Tamaño	Nulo	Llave
id	SERIAL	Identificador único	4 bytes	No	Primaria
producto_id	INTEGER	ID producto	4 bytes	No	Foránea (PRODUCTO)
fecha	DATE	Fecha movimiento	4 bytes	No	
tipo	VARCHAR(40)	Tipo movimiento	40 caracteres	No	
cantidad	INTEGER	Cantidad	4 bytes	No	
referencia	VARCHAR(60)	Referencia	60 caracteres	Sí	

FACTURA

Atributo	Tipo de Dato	Descripción	Tamaño	Nulo	Llave
nro	SERIAL	Número factura	4 bytes	No	Primaria
usuario_id	INTEGER	ID usuario	4 bytes	No	Foránea (USUARIO)
fecha_emision	DATE	Fecha emisión	4 bytes	No	
subtotal	FLOAT	Subtotal	8 bytes	No	
impuesto	FLOAT	Impuesto	8 bytes	No	
total	FLOAT	Total	8 bytes	No	
estado	VARCHAR(40)	Estado	40 caracteres	No	

DETALLE_FACTURA

Atributo	Tipo de Dato	Descripción	Tamaño	Nulo	Llave
id	SERIAL	Identificador único	4 bytes	No	Primaria
factura_nro	INTEGER	Número factura	4 bytes	No	Primaria, Foránea (FACTURA)
producto_id	INTEGER	ID producto	4 bytes	Sí	Foránea (PRODUCTO)
servicio_id	INTEGER	ID servicio	4 bytes	Sí	Foránea (SERVICIO)
tipo_item	VARCHAR(40)	Tipo ítem	40 caracteres	No	
cantidad	INTEGER	Cantidad	4 bytes	No	
precio_unitario	FLOAT	Precio unitario	8 bytes	No	
subtotal	FLOAT	Subtotal	8 bytes	No	

PAGO

Atributo	Tipo de Dato	Descripción	Tamaño	Nulo	Llave
id	SERIAL	Identificador único	4 bytes	No	Primaria
factura_nro	INTEGER	Número factura	4 bytes	No	Foránea (FACTURA)
fecha	DATE	Fecha pago	4 bytes	No	
monto	FLOAT	Monto	8 bytes	No	
metodo	VARCHAR(40)	Método pago	40 caracteres	No	

Script

-----PERSONAS/USUARIOS-----

```
CREATE TABLE vetcare.persona
(
    id SERIAL NOT NULL PRIMARY KEY,
    nombre VARCHAR(40) NOT NULL,
    apellido VARCHAR(40) NOT NULL,
    telefono VARCHAR(30) NOT NULL,
    email TEXT,
    direccion TEXT NOT NULL,
    fecha_nacimiento DATE NOT NULL
);

CREATE TABLE vetcare.dato_profesional
(
    id SERIAL NOT NULL PRIMARY KEY,
    matricula INTEGER NOT NULL,
    especialidad VARCHAR(40) NOT NULL,
    cargo VARCHAR (40) NOT NULL
);

CREATE TABLE vetcare.usuario
(
    id SERIAL PRIMARY KEY,
    persona_id INTEGER NOT NULL UNIQUE,
    user_name VARCHAR(30) NOT NULL UNIQUE,
    password_hash VARCHAR(255) NOT NULL,
    rol_id INTEGER NOT NULL,
    activo BOOLEAN NOT NULL,
    dato_profesional_id INTEGER NOT NULL,
    FOREIGN KEY(persona_id) REFERENCES vetcare.persona(id)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
    FOREIGN KEY(rol_id) REFERENCES vetcare.rol(id)
```

```
    ON DELETE CASCADE
    ON UPDATE CASCADE,
    FOREIGN KEY (dato_profesional_id) REFERENCES
vetcare.dato_profesional(id)
    ON DELETE CASCADE
    ON UPDATE CASCADE
);
```

```
CREATE TABLE vetcare.rol
(
id SERIAL PRIMARY KEY NOT NULL,
nombre VARCHAR(25) NOT NULL,
descripcion VARCHAR(50) NULL
);
```

-----ANIMALES/MASCOTAS-----

```
create table vetcare.especie
(
id serial not null primary key,
nombre varchar(40) not null
);
```

```
create table vetcare.raza
(
id serial not null primary key,
especie_id integer not null,
nombre varchar(40) not null,
foreign key(especie_id) references vetcare.especie(id)
on delete cascade on update cascade
);
```

```
CREATE TABLE vetcare.mascota
(
id SERIAL PRIMARY KEY NOT NULL,
usuario_id INTEGER NOT NULL,
raza_id INTEGER NOT NULL,
nombre VARCHAR(40) NOT NULL,
sexo VARCHAR(40) NOT NULL,
fecha_nacimiento DATE NOT NULL,
FOREIGN KEY(usuario_id) REFERENCES vetcare.usuario(id)
ON DELETE CASCADE
ON UPDATE CASCADE,
FOREIGN KEY(raza_id) REFERENCES vetcare.raza(id)
ON DELETE CASCADE
ON UPDATE CASCADE
);
```

-----VACUNAS/CITAS-----

```
CREATE TABLE vetcare.vacuna
(
id SERIAL NOT NULL PRIMARY KEY,
nombre VARCHAR(40) NOT NULL,
descripcion VARCHAR(80) NOT NULL,
dosis_ml INTEGER
);

CREATE TABLE vetcare.aplicacion_vacuna
(
id SERIAL NOT NULL PRIMARY KEY,
vacuna_id INTEGER NOT NULL,
fecha_aplicacion DATE NOT NULL,
proxima_dosis DATE,
lote VARCHAR(60),
FOREIGN KEY (vacuna_id) REFERENCES vetcare.vacuna(id)
ON DELETE CASCADE
ON UPDATE CASCADE
);

--ENUM ESTADO_CITA
CREATE TYPE estado_cita AS
ENUM('Pendiente','Confirmada','Cancelada','Atendida');

CREATE TABLE vetcare.cita
(
nro SERIAL NOT NULL PRIMARY KEY,
usuario_id INTEGER NOT NULL,
ap_vac_id INTEGER NOT NULL,
motivo VARCHAR(60) NOT NULL,
fecha_hora TIMESTAMP NOT NULL,
estado estado_cita NOT NULL,
FOREIGN KEY(ap_vac_id) REFERENCES vetcare.aplicacion_vacuna(id)
ON DELETE CASCADE
ON UPDATE CASCADE,
FOREIGN KEY(usuario_id) REFERENCES vetcare.usuario(id)
ON DELETE CASCADE
ON UPDATE CASCADE
);

SELECT *
FROM vetcare.cita;
```

-----TRATAMIENTOS Y CONSULTAS-----

```
CREATE TABLE vetcare.tratamiento
(
id SERIAL NOT NULL PRIMARY KEY
);

CREATE TABLE vetcare.consulta
(
nro SERIAL NOT NULL PRIMARY KEY,
nro_cita INTEGER NOT NULL,
motivo VARCHAR(60) NOT NULL,
diagnostico VARCHAR(60),
peso FLOAT NOT NULL,
temperatura FLOAT NOT NULL,
fecha_consulta DATE NOT NULL
);

CREATE TABLE vetcare.aplica
(
nro_consulta INTEGER NOT NULL,
tratamiento_id INTEGER NOT NULL,
descripcion VARCHAR(60),
dosis FLOAT NOT NULL,
duracion TIME NOT NULL,
observaciones VARCHAR(60),
PRIMARY KEY (nro_consulta, tratamiento_id),
FOREIGN KEY (nro_consulta) REFERENCES vetcare.consulta(nro)
ON DELETE CASCADE
ON UPDATE CASCADE,
FOREIGN KEY (tratamiento_id) REFERENCES vetcare.tratamiento(id)
ON DELETE CASCADE
ON UPDATE CASCADE
);
```

-----CREACION INVETARIOS/FACTURAS/PAGOS-----

```
CREATE TABLE vetcare.servicio
(
id SERIAL NOT NULL PRIMARY KEY,
nombre VARCHAR(40) NOT NULL,
descripcion VARCHAR(60),
precio FLOAT NOT NULL
);

CREATE TABLE vetcare.producto
(
id SERIAL NOT NULL PRIMARY KEY,
```

```
nombre VARCHAR(40) NOT NULL,
descripcion VARCHAR(60),
tipo VARCHAR(40) NOT NULL,
precio FLOAT NOT NULL,
stock INTEGER NOT NULL,
cortesia BOOLEAN NOT NULL
);

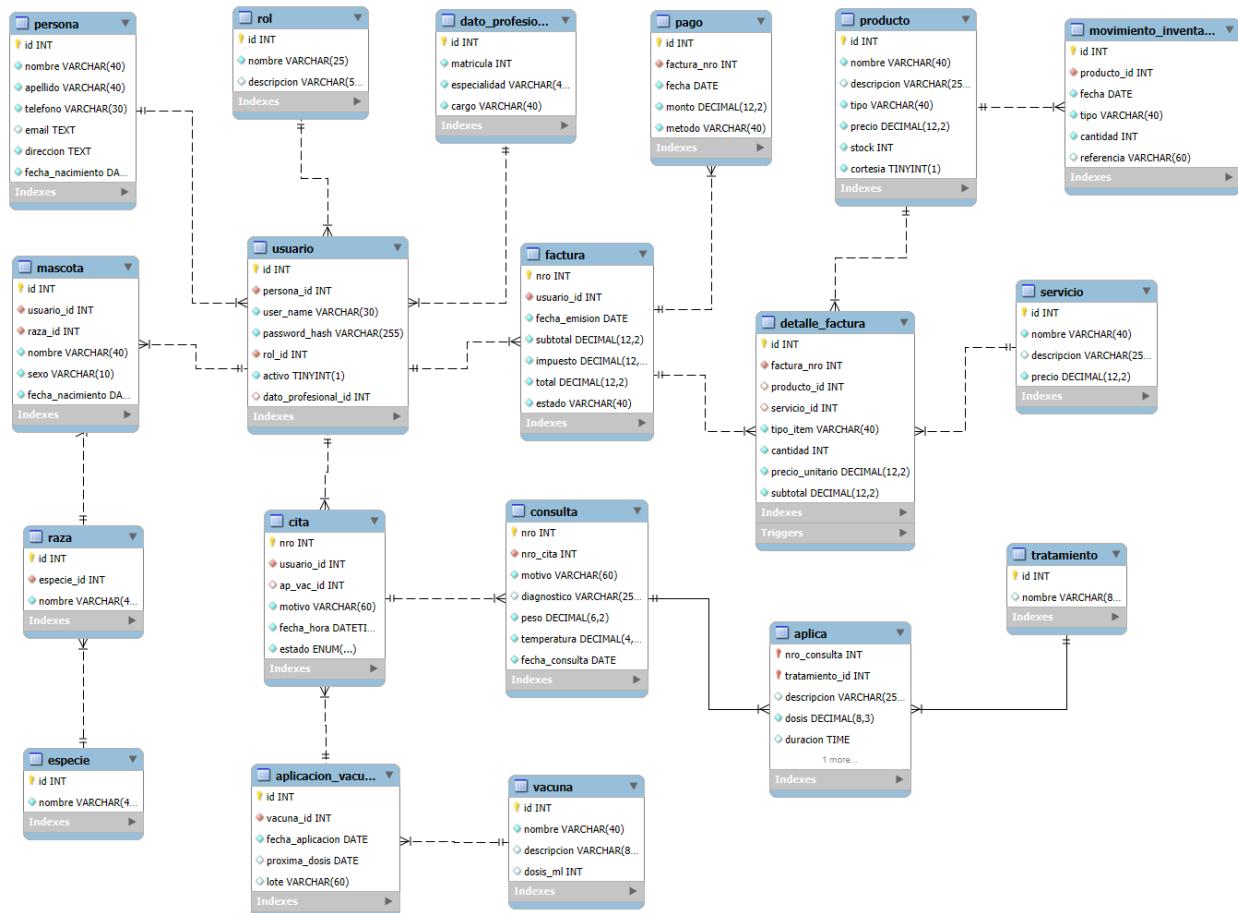
CREATE TABLE vetcare.movimiento_inventario
(
id SERIAL NOT NULL PRIMARY KEY,
producto_id INTEGER NOT NULL,
fecha DATE NOT NULL,
tipo VARCHAR(40) NOT NULL,
cantidad INTEGER NOT NULL,
referencia VARCHAR(60),
FOREIGN KEY(producto_id) REFERENCES vetcare.producto(id)
ON DELETE CASCADE
ON UPDATE CASCADE
);

CREATE TABLE vetcare.factura
(
nro SERIAL NOT NULL PRIMARY KEY,
usuario_id INTEGER NOT NULL,
fecha_emision DATE NOT NULL,
subtotal FLOAT NOT NULL,
impuesto FLOAT NOT NULL,
total FLOAT NOT NULL,
estado VARCHAR(40) NOT NULL,
FOREIGN KEY(usuario_id) REFERENCES vetcare.usuario(id)
ON DELETE CASCADE
ON UPDATE CASCADE
);
```

```
CREATE TABLE vetcare.detalle_factura
(
    id SERIAL NOT NULL,
    factura_nro INTEGER NOT NULL,
    producto_id INTEGER,
    servicio_id INTEGER,
    tipo_item VARCHAR(40) NOT NULL,
    cantidad INTEGER NOT NULL,
    precio_unitario FLOAT NOT NULL,
    subtotal FLOAT NOT NULL,
    PRIMARY KEY (id,factura_nro),
    FOREIGN KEY(factura_nro) REFERENCES vetcare.factura(nro)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
    FOREIGN KEY(producto_id) REFERENCES vetcare.producto(id)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
    FOREIGN KEY(servicio_id) REFERENCES vetcare.servicio(id)
    ON DELETE CASCADE
    ON UPDATE CASCADE
);
```

```
CREATE TABLE vetcare.pago
(
    id SERIAL NOT NULL PRIMARY KEY,
    factura_nro INTEGER NOT NULL,
    fecha DATE NOT NULL,
    monto FLOAT NOT NULL,
    metodo VARCHAR(40) NOT NULL,
    FOREIGN KEY(factura_nro) REFERENCES vetcare.factura(nro)
    ON DELETE CASCADE
    ON UPDATE CASCADE
);
```

Diagrama Relacional



Actualización de Tuplas (Población de Datos)

```

INSERT INTO vetcare.rol (id, nombre, descripcion) VALUES
(1, 'Administrador', 'Acceso completo al sistema'),
(2, 'Veterinario', 'Encargado de consultas y tratamientos'),
(3, 'Cliente', 'Dueño de mascotas');

-- Veterinarios (IDs 1-5)
INSERT INTO vetcare.persona (id, nombre, apellido, telefono, email, direccion,
fecha_nacimiento) VALUES
(1,'Carlos','Pérez','70000001','cperez@vet.com','Av. Libertad 123','1980-04-12'),
(2,'María','Gómez','70000002','mgomez@vet.com','Calle Central 45','1985-11-22'),
(3,'Luis','Torrez','70000003','ltorrez@vet.com','Av. Busch 233','1990-02-10'),
(4,'Patricia','Suárez','70000004','psuarez@vet.com','Barrio Norte 10','1987-06-15'),
(5,'Jorge','Vargas','70000005','jvargas@vet.com','Av. América 45','1982-09-09');

```

```

-- Clientes (IDs 6-15)
INSERT INTO vetcare.persona (id, nombre, apellido, telefono, email, direccion,
fecha_nacimiento) VALUES
(6, 'José', 'López', '71000001', 'jlopez@gmail.com', 'Barrio Sur 101', '1975-02-10'),
(7, 'Ana', 'Fernández', '71000002', 'afernandez@gmail.com', 'Av. San Martín 234', '1988-06-15'),
(8, 'Luis', 'Ramírez', '71000003', 'lramirez@gmail.com', 'Calle 10 #55', '1995-08-09'),
(9, 'Carmen', 'Morales', '71000004', 'cmorales@gmail.com', 'Zona Central 50', '1980-12-30'),
(10, 'Hugo', 'Ríos', '71000005', 'hrios@gmail.com', 'Av. Busch 120', '1992-04-01'),
(11, 'Mónica', 'Salazar', '71000006', 'msalazar@gmail.com', 'Zona Norte 75', '1990-07-07'),
(12, 'Daniel', 'Quispe', '71000007', 'dquispe@gmail.com', 'Av. Blanco Galindo 233', '1987-09-25'),
(13, 'Claudia', 'Pinto', '71000008', 'cpinto@gmail.com', 'Zona Sur 12', '1994-10-13'),
(14, 'Oscar', 'Mamani', '71000009', 'omamani@gmail.com', 'Av. Aroma 111', '1979-03-20'),
(15, 'Gabriela', 'Aguilar', '71000010', 'gaguilar@gmail.com', 'Zona Este 200', '1986-08-18');

INSERT INTO vetcare.dato_profesional (id, matricula, especialidad, cargo) VALUES
(1, 1001, 'Medicina General', 'Veterinario'),
(2, 1002, 'Cirugía', 'Veterinario'),
(3, 1003, 'Dermatología', 'Veterinario'),
(4, 1004, 'Cardiología', 'Veterinario'),
(5, 1005, 'Neurología', 'Veterinario');

-- Veterinarios (IDs 1-5)
INSERT INTO vetcare.usuario (id, persona_id, user_name, password_hash, rol_id,
activo, dato_profesional_id) VALUES
(1, 1, 'cperez', 'hash1', 2, TRUE, 1),
(2, 2, 'mgomez', 'hash2', 2, TRUE, 2),
(3, 3, 'ltorrez', 'hash3', 2, TRUE, 3),
(4, 4, 'psuarez', 'hash4', 2, TRUE, 4),
(5, 5, 'jvargas', 'hash5', 2, TRUE, 5);

-- Clientes (IDs 6-15)
INSERT INTO vetcare.usuario (id, persona_id, user_name, password_hash, rol_id,
activo, dato_profesional_id) VALUES
(6, 6, 'jlopez', 'hash6', 3, TRUE, NULL),
(7, 7, 'afernandez', 'hash7', 3, TRUE, NULL),
(8, 8, 'lramirez', 'hash8', 3, TRUE, NULL),
(9, 9, 'cmorales', 'hash9', 3, TRUE, NULL),
(10, 10, 'hrios', 'hash10', 3, TRUE, NULL),
(11, 11, 'msalazar', 'hash11', 3, TRUE, NULL),
(12, 12, 'dquispe', 'hash12', 3, TRUE, NULL),
(13, 13, 'cpinto', 'hash13', 3, TRUE, NULL),
(14, 14, 'omamani', 'hash14', 3, TRUE, NULL),
(15, 15, 'gaguilar', 'hash15', 3, TRUE, NULL);

-- Especies
INSERT INTO vetcare.especie (id, nombre) VALUES
(1, 'Perro'),
(2, 'Gato');

```

```

-- Razas
INSERT INTO vetcare.raza (id, especie_id, nombre) VALUES
(1,1,'Labrador'),
(2,1,'Bulldog'),
(3,1,'Pastor Alemán'),
(4,2,'Siames'),
(5,2,'Persa'),
(6,2,'Maine Coon');

INSERT INTO vetcare.mascota (id, usuario_id, raza_id, nombre, sexo, fecha_nacimiento)
VALUES
(1,6,1,'Max','Macho','2019-01-10'),
(2,7,2,'Rocky','Macho','2018-05-22'),
(3,8,3,'Luna','Hembra','2020-03-14'),
(4,9,4,'Misu','Hembra','2017-08-09'),
(5,10,5,'Simba','Macho','2016-11-30'),
(6,11,6,'Nala','Hembra','2019-09-12'),
(7,12,1,'Bruno','Macho','2020-12-05'),
(8,13,2,'Coco','Macho','2021-01-15'),
(9,14,3,'Bella','Hembra','2018-06-20'),
(10,15,4,'Kitty','Hembra','2022-02-28'),
(11,6,1,'Toby','Macho','2020-07-11'),
(12,7,2,'Jack','Macho','2019-04-18'),
(13,8,3,'Molly','Hembra','2021-03-30'),
(14,9,4,'Lilly','Hembra','2017-12-05'),
(15,10,5,'Charlie','Macho','2018-09-09'),
(16,11,6,'Lucy','Hembra','2020-05-22'),
(17,12,1,'Duke','Macho','2021-08-12'),
(18,13,2,'Bailey','Macho','2022-01-19'),
(19,14,3,'Sasha','Hembra','2021-11-11'),
(20,15,4,'Lola','Hembra','2020-10-10');

-- Vacunas
INSERT INTO vetcare.vacuna (id,nombre,descripcion,dosis_ml) VALUES
(1,'Rabia','Vacuna contra rabia',2),
(2,'Parvovirus','Vacuna parvovirus canino',3),
(3,'Leucemia','Vacuna leucemia felina',2),
(4,'Moquillo','Vacuna moquillo canino',3);

-- Aplicaciones
INSERT INTO vetcare.aplicacion_vacuna
(id,vacuna_id,fecha_aplicacion,proxima_dosis,lote) VALUES
(1,1,'2023-01-10','2024-01-10','A123'),
(2,2,'2023-02-15','2023-08-15','B456'),
(3,3,'2023-03-12','2024-03-12','C789'),
(4,4,'2023-04-20','2023-10-20','D101'),
(5,1,'2023-05-05','2024-05-05','E202'),
(6,2,'2023-06-07','2023-12-07','F303'),
(7,3,'2023-07-11','2024-07-11','G404'),
(8,4,'2023-08-15','2024-08-15','H505'),
(9,1,'2023-09-20','2024-09-20','I606'),
(10,2,'2023-10-25','2024-10-25','J707');

```

```

INSERT INTO vetcare.cita (nro, usuario_id, ap_vac_id, motivo, fecha_hora, estado, mascota_id) VALUES
(1,1,1,'Chequeo general','2023-09-01 09:00','Pendiente',1),
(2,2,2,'Vacuna anual','2023-09-01 10:00','Confirmada',2),
(3,3,3,'Problema piel','2023-09-02 11:00','Pendiente',4),
(4,4,4,'Chequeo cardiaco','2023-09-02 13:00','Atendida',5),
(5,5,5,'Vacuna de refuerzo','2023-09-03 09:30','Confirmada',3),
(6,1,6,'Chequeo general','2023-09-03 10:30','Pendiente',6),
(7,2,7,'Problema digestivo','2023-09-04 11:30','Atendida',7),
(8,3,8,'Vacuna anual','2023-09-04 12:00','Confirmada',8),
(9,4,9,'Chequeo general','2023-09-05 09:00','Pendiente',9),
(10,5,10,'Vacuna felina','2023-09-05 10:30','Confirmada',10),
(11,1,1,'Chequeo general','2023-09-06 09:00','Pendiente',11),
(12,2,2,'Vacuna anual','2023-09-06 10:30','Confirmada',12),
(13,3,3,'Problema piel','2023-09-07 11:00','Atendida',13),
(14,4,4,'Chequeo cardiaco','2023-09-07 13:00','Pendiente',14),
(15,5,5,'Vacuna de refuerzo','2023-09-08 09:30','Confirmada',15);

INSERT INTO vetcare.tratamiento (id) VALUES
(1),(2),(3),(4),(5);

INSERT INTO vetcare.consulta (nro, nro_cita, motivo, diagnostico, peso, temperatura, fecha_consulta) VALUES
(1,4,'Chequeo cardiaco','Corazón saludable',5.2,38.5,'2023-09-02'),
(2,5,'Vacuna de refuerzo','Vacuna aplicada correctamente',4.5,38.0,'2023-09-03'),
(3,7,'Problema digestivo','Gastritis leve',6.1,38.7,'2023-09-04'),
(4,8,'Vacuna anual','Vacuna aplicada correctamente',3.8,38.2,'2023-09-04'),
(5,13,'Problema piel','Dermatitis',4.2,38.3,'2023-09-07');

INSERT INTO vetcare.aplica (nro_consulta, tratamiento_id, descripcion, dosis, duracion, observaciones) VALUES
(1,1,'Chequeo cardiaco completo',1,'00:30:00','Sin observaciones'),
(2,2,'Vacuna refuerzo',2,'00:15:00','Aplicar segunda dosis en 6 meses'),
(3,3,'Tratamiento digestivo',1,'00:20:00','Seguir dieta blanda'),
(4,4,'Vacuna anual',2,'00:15:00','Revisar estado general'),
(5,5,'Tratamiento dermatológico',1,'00:25:00','Aplicar crema 7 días');

INSERT INTO vetcare.servicio (id, nombre, descripcion, precio) VALUES
(1,'Consulta general','Chequeo completo del paciente',50),
(2,'Vacunación','Aplicación de vacunas',30),
(3,'Cirugía menor','Pequeñas intervenciones',150),
(4,'Baño y peluquería','Limpieza y corte de pelo',25),
(5,'Consulta cardiológica','Evaluación corazón',80);

INSERT INTO vetcare.producto (id, nombre, descripcion, tipo, precio, stock, cortesia) VALUES
(1,'Alimento balanceado','Comida para perros adultos','Alimento',40,100,0),
(2,'Arena para gatos','Arena higiénica','Accesorio',15,50,0),
(3,'Shampoo para perros','Shampoo medicado','Higiene',12,30,0),
(4,'Collar antipulgas','Collar repelente','Accesorio',10,40,0),
(5,'Vitaminas felinas','Suplemento vitamínico','Salud',20,25,0);

```

```

INSERT INTO vetcare.movimiento_inventario (id, producto_id, fecha, tipo, cantidad, referencia) VALUES
(1,1,'2023-08-01','Ingreso',50,'Compra proveedor'),
(2,2,'2023-08-02','Ingreso',30,'Compra proveedor'),
(3,3,'2023-08-03','Ingreso',20,'Compra proveedor'),
(4,4,'2023-08-04','Ingreso',40,'Compra proveedor'),
(5,5,'2023-08-05','Ingreso',25,'Compra proveedor'),
(6,1,'2023-09-01','Salida',5,'Venta a cliente'),
(7,2,'2023-09-02','Salida',2,'Venta a cliente'),
(8,3,'2023-09-03','Salida',1,'Uso en baño'),
(9,4,'2023-09-04','Salida',3,'Venta a cliente'),
(10,5,'2023-09-05','Salida',2,'Venta a cliente');

INSERT INTO vetcare.factura (nro, usuario_id, fecha_emision, subtotal, impuesto, total, estado) VALUES
(1,6,'2023-09-01',50,5,55,'Pagada'),
(2,7,'2023-09-01',30,3,33,'Pagada'),
(3,8,'2023-09-04',80,8,88,'Pendiente'),
(4,9,'2023-09-05',25,2.5,27.5,'Pagada'),
(5,10,'2023-09-06',150,15,165,'Pendiente');

INSERT INTO vetcare.detalle_factura (id, factura_nro, producto_id, servicio_id, tipo_item, cantidad, precio_unitario, subtotal) VALUES
(1,1,NULL,1,'Servicio',1,50,50),
(2,2,NULL,2,'Servicio',1,30,30),
(3,3,NULL,5,'Servicio',1,80,80),
(4,4,3,NULL,'Producto',1,25,25),
(5,5,NULL,3,'Servicio',1,150,150);

INSERT INTO vetcare.pago (id, factura_nro, fecha, monto, metodo) VALUES
(1,1,'2023-09-01',55,'Efectivo'),
(2,2,'2023-09-01',33,'Tarjeta'),
(3,4,'2023-09-05',27.5,'Efectivo');

```

Consultas

```
--CONSULTA 1.- OBTENER DATOS NOMBRE Y ROL AL INCIAR SESION
SELECT u.id,u.user_name,p.nombre,p.apellido ,r.nombre as rol
FROM vetcare.usuario u
JOIN vetcare.persona p ON p.id=u.persona_id
JOIN vetcare.rol r ON r.id=u.rol_id
```

1	cperez	Carlos	Pérez	veterinario
2	mgomez	María	Gómez	veterinario
3	ltorrez	Luis	Torrez	veterinario

```
--CONSULTA 2.- LISTAR A TODOS LOS USUARIOS CON EL ROL: "Cliente"
SELECT u.id,u.user_name,r.nombre
FROM vetcare.usuario u
JOIN vetcare.rol r ON r.id=u.rol_id
WHERE r.nombre='Veterinario';
```

6	jlopez	cliente
7	afernandez	cliente
8	lramirez	cliente

```
--CONSULTA 3.- LISTAR TODAS LAS MASCOTAS Y SU PROPIETARIO
SELECT p.nombre as dueño,p.telefono,m.nombre AS nombre_mascota,e.nombre AS
nombre_especie,r.nombre AS nombre_raza
FROM vetcare.usuario u
JOIN vetcare.mascota m ON m.usuario_id=u.id
JOIN vetcare.persona p ON p.id =u.persona_id
JOIN vetcare.raza r ON m.raza_id =r.id
JOIN vetcare.especie e ON e.id=r.especie_id
JOIN vetcare.cita c ON c.mascota_id=m.id ;
```

José	71000001	Max	Perro	Labrador
Ana	71000002	Rocky	Perro	Bulldog
Carmen	71000004	Misu	Gato	Siames

```
--CONSULTA 4.- CANTIDAD DE MASCOTAS POR ESPECIE
SELECT e.nombre,COUNT(m.id)
FROM vetcare.especie e
JOIN vetcare.raza r ON r.especie_id = e.id
JOIN vetcare.mascota m ON m.raza_id=r.id
GROUP BY e.nombre;
```

Perro	17
Gato	10

```
--CONSULTA 5.- USUARIOS SIN MASCOTA
SELECT u.id,u.user_name
FROM vetcare.usuario u
WHERE u.id NOT IN
(
    SELECT usuario_id
    FROM vetcare.mascota m
);
```

1	cperez
2	mgomez
3	ltorrez

```
--CONSULTA 6.- PROXIMAS DOSIS DE VACUNAS POR MASCOTA
SELECT m.id,m.nombre,av.proxima_dosis
FROM vetcare.aplicacion_vacuna av
JOIN vetcare.cita c ON c.ap_vac_id =av.id
JOIN vetcare.mascota m ON m.id = c.mascota_id
JOIN vetcare.usuario u ON u.id=c.usuario_id
WHERE av.proxima_dosis >= CURRENT_DATE
GROUP BY m.id,m.nombre,av.proxima_dosis;
```

15	Charlie	2024-05-05
14	Lilly	2023-10-20
11	Toby	2024-01-10

```
--CONSULTA 7.- VACUNAS MAS APLICADAS
SELECT v.nombre AS vacuna, COUNT(*) AS aplicaciones
FROM vetcare.aplicacion_vacuna av
JOIN vetcare.vacuna v ON v.id = av.vacuna_id
GROUP BY v.nombre
ORDER BY aplicaciones DESC;
```

Rabia	3
Parvovirus	3
Moquillo	2

```
--CONSULTA 8.- CITAS PENDIENTES DE HOY
SELECT c.nro, c.fecha_hora, c.motivo, u.user_name AS cliente, m.nombre AS mascota
FROM vetcare.cita c
JOIN vetcare.usuario u ON u.id = c.usuario_id
JOIN vetcare.mascota m ON m.id = c.mascota_id
WHERE c.estado = 'Pendiente'
AND c.fecha_hora >= '2024-01-01'
ORDER BY c.fecha_hora;
```

1	2023-09-01	09:00:00.000	Chequeo general	cperez	Max
3	2023-09-02	11:00:00.000	Problema piel	ltorrez	Misu
6	2023-09-03	10:30:00.000	Chequeo general	cperez	Nala

```
--CONSULTA 9.- CONSULTAS CON DIAGNOSTICO Y MASCOTA
SELECT co.nro, co.fecha_consulta, co.diagnostico, m.nombre AS mascota, u.user_name AS cliente
FROM vetcare.consulta co
JOIN vetcare.cita c ON c.nro = co.nro_cita
JOIN vetcare.mascota m ON m.id = c.mascota_id
JOIN vetcare.usuario u ON u.id = m.usuario_id
ORDER BY co.fecha_consulta DESC;
```

5	2023-09-07	Dermatitis	Molly	lramirez
3	2023-09-04	Gastritis leve	Bruno	dquispe
4	2023-09-04	Vacuna aplicada correctamente	Coco	cpinto

```
--CONSULTA 10.- PESO PROMEDIO Y TEMPERATURA PROMEDIO POR MASCOTA
SELECT m.nombre AS mascota, AVG(co.peso), 2 AS peso_promedio, AVG(co.temperatura) AS temp_promedio
FROM vetcare.consulta co
JOIN vetcare.cita c ON co.nro_cita = c.nro
JOIN vetcare.mascota m ON m.id = c.mascota_id
GROUP BY m.nombre;
```

Luna	4.5	2	38.0
Bruno	6.1	2	38.7
Simba	5.2	2	38.5

```
--CONSULTA 11.- DUEÑOS CON MAS MASCOTAS REGISTRADAS
SELECT u.user_name, COUNT(m.id) as cant_mascotas
FROM vetcare.mascota m
JOIN vetcare.usuario u ON u.id = m.usuario_id
GROUP BY u.user_name
ORDER BY cant_mascotas DESC;
```

afernandez	2
msalazar	2
cpinto	2

```
--CONSULTA 12.- MASCOTAS QUE NUNCA TUVIERON UNA CITA
SELECT m.id, m.nombre, r.nombre
FROM vetcare.mascota m
JOIN vetcare.raza r ON r.id = m.raza_id
WHERE m.id NOT IN
(
    SELECT mascota_id
    FROM vetcare.cita
);
```

16	Lucy	Maine Coon
17	Duke	Labrador
18	Bailey	Bulldog

```
--CONSULTA 14.- LISTAR LAS FACTURAS PENDIENTES CON INFORMACIÓN DEL CLIENTE
SELECT f.nro, f.fecha_emision, f.total, u.user_name AS cliente
FROM vetcare.factura f
JOIN vetcare.usuario u ON u.id = f.usuario_id
WHERE f.estado = 'Pendiente'
ORDER BY f.fecha_emision ASC;
```

3	2023-09-04	88.0	lramirez
5	2023-09-06	165.0	hrios

```
--CONSULTA 15.- OBTENER LOS SERVICIOS MÁS VENDIDOS (POR NÚMERO DE VECES)
SELECT s.nombre, COUNT(df.servicio_id) AS cantidad_vendida
FROM vetcare.detalle_factura df
JOIN vetcare.servicio s ON s.id = df.servicio_id
GROUP BY s.nombre
ORDER BY cantidad_vendida DESC;
```

Consulta general	1
Cirugía menor	1
Consulta cardiológica	1

```
--CONSULTA 16.- OBTENER LOS PRODUCTOS MAS VENDIDOS
SELECT p.nombre, COUNT(df.producto_id) as cantidad_vendida
FROM vetcare.detalle_factura df
JOIN vetcare.producto p ON p.id = df.producto_id
GROUP BY p.nombre
ORDER BY cantidad_vendida DESC;
```

Shampoo para perros	1
---------------------	---

```
--CONSULTA 17.- OBTENER VALOR DEL INVENTARIO ACTUAL
SELECT SUM(precio*stock) as valor_inventario
FROM vetcare.producto p;
```

6010.0

```
--CONSULTA 18.- MOSTRAR QUE VETERINARIO ATENDIO A QUE PERRO
SELECT m.nombre as nombre_mascota, p2.nombre as nombre_dueño, p2.apellido as apellido_dueño, p.nombre as nombre_vet, p.apellido as apellido_vet
FROM vetcare.cita c
JOIN vetcare.usuario u ON u.id=c.usuario_id
JOIN vetcare.persona p ON p.id=u.persona_id
JOIN vetcare.mascota m ON m.id = c.mascota_id
JOIN vetcare.usuario u2 ON u2.id=m.usuario_id
JOIN vetcare.persona p2 ON p2.id=u2.persona_id
GROUP BY m.nombre, p2.nombre, p2.apellido, p.nombre, p.apellido;
```

Kitty	Gabriela	Aguilar	Jorge	Vargas
Jack	Ana	Fernández	María	Gómez
Max	José	López	Carlos	Pérez

```
--CONSULTA 19.- PROXIMAS CITAS DE UN VETERINARIO
SELECT c.nro, c.fecha_hora, m.nombre AS mascota, u.user_name AS cliente
FROM vetcare.cita c
JOIN vetcare.mascota m ON m.id = c.mascota_id
JOIN vetcare.usuario u ON u.id = c.usuario_id
WHERE c.usuario_id = 2 -- id del veterinario
    AND c.fecha_hora >= '2022-01-01'
ORDER BY c.fecha_hora ASC;
```

2	2023-09-01 10:00:00.000	Rocky	mgomez
7	2023-09-04 11:30:00.000	Bruno	mgomez
12	2023-09-06 10:30:00.000	Jack	mgomez

--CONSULTA 20.-HISTORIAL COMPLETO DE VACUNAS DE UNA MASCOTA

```
SELECT v.nombre AS vacuna, av.fecha_aplicacion, av.proxima_dosis, av.lote
FROM vetcare.aplicacion_vacuna av
JOIN vetcare.vacuna v ON v.id = av.vacuna_id
JOIN vetcare.cita c ON c.ap_vac_id = av.id
WHERE c.mascota_id = 5-- id de la mascota
ORDER BY av.fecha_aplicacion DESC;
```

Moquillo	2023-04-20	2023-10-20	D101
----------	------------	------------	------

--CONSULTA 21.-CLIENTES CON FACTURAS PENDIENTES Y TOTAL DEUDA

```
SELECT u.user_name AS cliente, COUNT(f.nro) AS facturas_pendientes, SUM(f.total) AS total_adeudado
FROM vetcare.factura f
JOIN vetcare.usuario u ON u.id = f.usuario_id
WHERE f.estado = 'Pendiente'
GROUP BY u.user_name
ORDER BY total_adeudado DESC;
```

hrios	1	165.0
lramirez1		88.0

--CONSULTA 22.- VENTAS TOTALES POR PRODUCTOS

```
SELECT p.nombre AS producto, SUM(df.cantidad) AS total_vendido, SUM(df.subtotal) AS ingresos
FROM vetcare.detalle_factura df
RIGHT JOIN vetcare.producto p ON p.id = df.producto_id
GROUP BY p.nombre
ORDER BY ingresos DESC;
```

Collar antipulgas		
Alimento balanceado		
Shampoo para perros	1	25.0

```
--CONSULTA 23.- VENTAS TOTALES POR SERVICIO
SELECT s.nombre AS producto, SUM(df.cantidad) AS total_vendido, SUM(df.subtotal) AS ingresos
FROM vetcare.detalle_factura df
RIGHT JOIN vetcare.servicio s ON s.id = df.producto_id
GROUP BY s.nombre
ORDER BY ingresos DESC;
```

Consulta cardiológica

Vacunación

Cirugía menor	1	25.0
---------------	---	------

```
-- CONSULTA 24.- MASCOTAS QUE NO HAN RECIBIDO NINGUNA VACUNA
SELECT m.id, m.nombre, r.nombre AS raza, e.nombre AS especie
FROM vetcare.mascota m
JOIN vetcare.raza r ON r.id = m.raza_id
JOIN vetcare.especie e ON e.id = r.especie_id
WHERE m.id NOT IN (
    SELECT c.mascota_id
    FROM vetcare.cita c
    WHERE c.ap_vac_id IS NOT NULL
);
```

16	Lucy	Maine Coon	Gato
17	Duke	Labrador	Perro
18	Bailey	Bulldog	Perro

```
-- CONSULTA 25.- CONSULTAS REALIZADAS POR CADA VETERINARIO
SELECT u.user_name AS veterinario, p.nombre AS nombre_vet, p.apellido AS apellido_vet,
COUNT(co.nro) AS consultas_realizadas
FROM vetcare.consulta co
JOIN vetcare.cita c ON c.nro = co.nro_cita
JOIN vetcare.usuario u ON u.id = c.usuario_id
JOIN vetcare.persona p ON p.id=u.persona_id
GROUP BY u.user_name, p.nombre, p.apellido
ORDER BY consultas_realizadas DESC;
```

ltorrez	Luis	Torrez	2
mgomez	María	Gómez	1
jvargas	Jorge	Vargas	1

```
--CONSULTA 26.- MASCOTAS CON CITAS Y PROXIMO CONTROL
SELECT m.id, m.nombre AS mascota, MAX(c.fecha_hora) AS ultima_cita, MIN(c.fecha_hora) AS proxima_cita
FILTER (WHERE c.fecha_hora > CURRENT_DATE)
FROM vetcare.mascota m
LEFT JOIN vetcare.cita c ON c.mascota_id = m.id
GROUP BY m.id, m.nombre
ORDER BY m.nombre;
```

18	Bailey	
9	Bella	2023-09-05 09:00:00.000
7	Bruno	2023-09-04 11:30:00.000

```
--CONSULTA 27.- CLIENTES CON MAS GASTOS ACUMULADOS
SELECT u.user_name AS cliente, SUM(f.total) AS total_gastado
FROM vetcare.factura f
JOIN vetcare.usuario u ON u.id = f.usuario_id
GROUP BY u.user_name
ORDER BY total_gastado DESC;
```

hrios	165.0
lramirez	88.0
jlopez	55.0

```
--CONSULTA 28.- PRODUCTOS CON BAJO STOCK (<30)
SELECT id, nombre, stock, precio
FROM vetcare.producto
WHERE stock < 30
ORDER BY stock ASC;
```

5	Vitaminas felinas	25	20.0
---	-------------------	----	------

```
--CONSULTA 29.- HISTORIAL DE PAGOS DE UN CLIENTE
SELECT p.id AS pago_id, f.nro AS factura, p.fecha, p.monto, p.metodo
FROM vetcare.pago p
JOIN vetcare.factura f ON f.nro = p.factura_nro
WHERE f.usuario_id = 9
ORDER BY p.fecha DESC;
```

3	4	2023-09-05	27.5	Efectivo
---	---	------------	------	----------

```
--CONSULTA 30.- CITAS AGRUPADAS POR VETERINARIO
SELECT u.user_name AS veterinario, COUNT(c.nro) AS citas_atendidas
FROM vetcare.cita c
JOIN vetcare.usuario u ON u.id = c.usuario_id
WHERE c.estado = 'Atendida'
GROUP BY u.user_name
ORDER BY citas_atendidas DESC;
```

ltorrez	1
mgomez	1
psuarez	1

```
--CONSULTA 31.- TOP 3 CLIENTES CON MAS CITAS REGISTRADAS
SELECT u.user_name ,COUNT(c.nro) AS cant_citas
FROM vetcare.usuario u
JOIN vetcare.mascota m ON m.usuario_id = u.id
JOIN vetcare.cita c ON c.mascota_id = m.id
GROUP BY u.user_name
ORDER BY cant_citas DESC
LIMIT 3;
```

cmorales	2
jlopez	2
hrios	2

```
-- CONSULTA 32.- CANT DE VACUNAS POR MASCOTA
SELECT m.id ,m.nombre , COUNT(v.id) as cant_vacunas_aplicadas
FROM vetcare.mascota m
LEFT JOIN vetcare.cita c ON c.mascota_id=m.id
LEFT JOIN vetcare.aplicacion_vacuna av ON av.id =c.ap_vac_id
LEFT JOIN vetcare.vacuna v ON v.id =av.vacuna_id
GROUP BY m.id,m.nombre
```

5	Simba	1
18	Bailey	0
2	Rocky	1

```
-- CONSULTA 33.- TOP 3 VETERINARIOS CON MAS CONSULTAS ATENDIDAS
SELECT u.id,u.user_name,R.nombre,COUNT(c2.nro) AS cant_consultas
FROM vetcare.usuario u
JOIN vetcare.rol r ON r.id = u.rol_id
JOIN vetcare.cita c ON c.usuario_id =u.id
JOIN vetcare.consulta c2 ON c2.nro_cita =c.nro
WHERE lower(r.nombre) ='veterinario'
GROUP BY u.id ,u.user_name,r.nombre
ORDER BY cant_consultas DESC
LIMIT 3;
```

3	ltorrez	veterinario	2
2	mgomez	veterinario	1
4	psuarez	veterinario	1

```
-- CONSULTA 34.- INGRESOS POR MES
SELECT fecha_emision,SUM(total)
FROM vetcare.factura f
WHERE lower(estado) ='pagada'
GROUP BY f.fecha_emision
ORDER BY f.fecha_emision asc;
```

2023-09-01	88.0
2023-09-05	27.5

```
-- CONSULTA 35.- CITAS PENDIENTES MES ACTUAL
SELECT CURRENT_DATE AS mes,COUNT(c.nro) as cantidad_citas_pendientes
FROM vetcare.cita c
WHERE lower(c.estado::text) ='pendiente';
/*CITA.ESTADO -> ENUM(PENDIENTE,ATENDIDA,CONFIRMADA)*/
```

2025-09-22	6
------------	---

```
--CONSULTA 36.- LISTAR MASCOTAS SIN CONSULTAS REGISTRADAS
SELECT m.id,m.nombre, COUNT(c2.nro) cant_consult
FROM vetcare.mascota m
LEFT JOIN vetcare.cita c ON c.mascota_id =m.id
LEFT JOIN vetcare.consulta c2 ON c2.nro_cita =c.nro
GROUP BY m.id,m.nombre
HAVING COUNT(c.nro) <1;
```

18	Bailey	0
16	Lucy	0
26	Rocky III	0

Procedimientos almacenados

```
-- DROP PROCEDURE vetcare.p_register_pet(int4, varchar, varchar, date, varchar,
varchar);

CREATE OR REPLACE PROCEDURE vetcare.p_register_pet(IN p_persona_id integer, IN
p_nombre_mascota character varying, IN p_sexo character varying, IN
p_fecha_nacimiento date, IN p_nombre_especie character varying, IN p_nombre_raza
character varying)
LANGUAGE plpgsql
AS $procedure$
DECLARE
    v_especie_id INTEGER;
    v_raza_id INTEGER;
BEGIN
    -- PASO 1: Validar si esta persona ya tiene una mascota con el mismo nombre.
    -- Esto previene registros duplicados para un mismo dueño.
    IF EXISTS (
        SELECT 1
        FROM vetcare.mascota m
        WHERE m.persona_id = p_persona_id AND m.nombre = p_nombre_mascota
    ) THEN
        RAISE EXCEPTION 'El dueño ya tiene una mascota registrada con el nombre
"%".', p_nombre_mascota;
    END IF;

    -- PASO 2: Obtener o crear la especie.
    -- Primero, intentamos obtener el ID de la especie si ya existe.
    SELECT id INTO v_especie_id FROM vetcare.especie WHERE nombre = p_nombre_especie;

    -- Si no se encontró (v_especie_id es NULL), la insertamos.
    IF v_especie_id IS NULL THEN
        INSERT INTO vetcare.especie (nombre) VALUES (p_nombre_especie)
        RETURNING id INTO v_especie_id;
    END IF;
```

```

-- PASO 3: Obtener o crear la raza, vinculada a la especie correcta.
-- Intentamos obtener el ID de la raza si ya existe para esa especie.
SELECT id INTO v_raza_id FROM vetcare.raza WHERE nombre = p_nombre_raza AND
especie_id = v_especie_id;

-- Si no se encontró, la insertamos.
IF v_raza_id IS NULL THEN
    INSERT INTO vetcare.raza (especie_id, nombre) VALUES (v_especie_id,
p_nombre_raza)
        RETURNING id INTO v_raza_id;
END IF;

-- PASO 4: Insertar la nueva mascota con los IDs correctos.
-- NOTA IMPORTANTE: Se usa p_persona_id, que es el ID del dueño.
-- Tu código original usaba incorrectamente el ID del usuario de la sesión.
INSERT INTO vetcare.mascota (persona_id, raza_id, nombre, sexo, fecha_nacimiento)
VALUES (p_persona_id, v_raza_id, p_nombre_mascota, p_sexo, p_fecha_nacimiento);

END;
$procedure$
;
```

```
-- DROP PROCEDURE vetcare.p_register(varchar, varchar, varchar, text, text, date,
varchar, varchar);
```

```

CREATE OR REPLACE PROCEDURE vetcare.p_register(IN p_nombre character varying, IN
p_apellido character varying, IN p_telefono character varying, IN p_email text, IN
p_direccion text, IN p_fecha_nacimiento date, IN p_user_name character varying, IN
p_password_hash character varying)
LANGUAGE plpgsql
AS $procedure$
DECLARE
    v_persona_id INTEGER;
BEGIN
    -- PASO 1: Validar si el nombre de usuario ya existe.
    -- Si la subconsulta encuentra una fila, la condición es verdadera.

    IF EXISTS (SELECT 1 FROM vetcare.usuario WHERE user_name = p_user_name) THEN
        -- Si el usuario ya existe, se levanta una excepción con un mensaje claro.
        -- Este error será capturado por el bloque try...except en el código de
Python.
        RAISE EXCEPTION 'El nombre de usuario "%" ya existe.', p_user_name;
    END IF;
```

```

-- PASO 2: Insertar los datos en la tabla 'persona'.
-- Se usa RETURNING id para obtener el ID recién creado y guardarlo en la
variable v_persona_id.
    INSERT INTO vetcare.persona (nombre, apellido, telefono, email, direccion,
fecha_nacimiento)
```

```

    VALUES (p_nombre, p_apellido, p_telefono, p_email, p_direccion,
p_fecha_nacimiento)
    RETURNING id INTO v_persona_id;

    -- PASO 3: Crear el registro en 'usuario' con los datos proporcionados.
    -- Se usa la variable v_persona_id y el hash de contraseña recibido.
    INSERT INTO vetcare.usuario (persona_id, user_name, password_hash, rol_id,
activo)
    VALUES (v_persona_id, p_user_name, p_password_hash, 3, TRUE); -- Rol 3 = Cliente

END;
$procedure$
;
```

Triggers

trg_actualizar_total_factura

Descripción: Se activa cada vez que se inserta, actualiza o elimina una línea en la tabla detalle_factura.
Automáticamente recalcula el subtotal, impuesto y total en la tabla factura correspondiente, asegurando que los datos siempre sean correctos.

```

-- Primero, creamos la función que se ejecutará
CREATE OR REPLACE FUNCTION vetcare.fn_actualizar_total_factura()
RETURNS TRIGGER AS $$

DECLARE
    v_factura_id INTEGER;
    v_subtotal DECIMAL;
    v_impuesto DECIMAL; -- Por ejemplo, 13%
BEGIN
    -- Determinar el ID de la factura afectada
    IF TG_OP = 'DELETE' THEN
        v_factura_id := OLD.factura_nro;
    ELSE
        v_factura_id := NEW.factura_nro;
    END IF;

    -- Calcular el nuevo subtotal sumando todas las líneas de esa factura
    SELECT COALESCE(SUM(subtotal), 0)
    INTO v_subtotal
    FROM vetcare.detalle_factura
    WHERE factura_nro = v_factura_id;

    -- Calcular el impuesto (puedes ajustar el porcentaje)
    v_impuesto := v_subtotal * 0.13;

    -- Actualizar la tabla principal de factura
    UPDATE vetcare.factura
    SET subtotal = v_subtotal + v_impuesto
    WHERE factura_nro = v_factura_id;

```

```

UPDATE vetcare.factura
SET
    subtotal = v_subtotal,
    impuesto = v_impuesto,
    total = v_subtotal + v_impuesto
WHERE nro = v_factura_id;

RETURN NEW;
END;
$$ LANGUAGE plpgsql;

-- Finalmente, creamos el trigger y lo asociamos a la tabla
CREATE TRIGGER trg_actualizar_total_factura
AFTER INSERT OR UPDATE OR DELETE ON vetcare.detalle_factura
FOR EACH ROW
EXECUTE FUNCTION vetcare.fn_actualizar_total_factura();

```

trg_auditar_cambios

Descripción: Se activa después de cualquier actualización o eliminación en tablas críticas (como persona o factura). Inserta un registro en una tabla de auditoría, guardando la información anterior del dato, quién realizó el cambio y a qué hora.

```

-- Primero, necesitamos una tabla para guardar los logs de auditoría
CREATE TABLE vetcare.auditoria (
    id SERIAL PRIMARY KEY,
    nombre_tabla VARCHAR(100),
    operacion CHAR(1), -- 'U' for Update, 'D' for Delete
    usuario_db NAME,
    fecha_hora TIMESTAMP,
    datos_viejos JSONB
);

```

```

-- Luego, la función genérica que se puede usar en varias tablas
CREATE OR REPLACE FUNCTION vetcare.fn_auditar_cambios()
RETURNS TRIGGER AS $$
BEGIN

```

```

    IF (TG_OP = 'UPDATE') THEN
        INSERT INTO vetcare.auditoria (nombre_tabla, operacion, usuario_db,
fecha_hora, datos_viejos)
        VALUES (TG_TABLE_NAME, 'U', session_user, now(), row_to_json(OLD));
        RETURN NEW;
    ELSIF (TG_OP = 'DELETE') THEN
        INSERT INTO vetcare.auditoria (nombre_tabla, operacion, usuario_db,
fecha_hora, datos_viejos)
        VALUES (TG_TABLE_NAME, 'D', session_user, now(), row_to_json(OLD));
        RETURN OLD;
    END IF;
    RETURN NULL;
END;
$$ LANGUAGE plpgsql;

-- Finalmente, asociamos el trigger a las tablas que queremos auditar
CREATE TRIGGER trg_auditar_cambios_persona
AFTER UPDATE OR DELETE ON vetcare.persona
FOR EACH ROW
EXECUTE FUNCTION vetcare.fn_auditar_cambios();

CREATE TRIGGER trg_auditar_cambios_factura
AFTER UPDATE OR DELETE ON vetcare.factura
FOR EACH ROW
EXECUTE FUNCTION vetcare.fn_auditar_cambios();

trg_prevenir_borrado_con_historial

Descripción: Se activa antes de intentar borrar una fila en la tabla persona o
mascota.
Revisa si existen registros relacionados en otras tablas (citas, facturas, etc.).
Si encuentra alguno, cancela la operación de borrado y devuelve un mensaje de error
claro y descriptivo.

-- Función para verificar el historial de una persona/cliente
CREATE OR REPLACE FUNCTION vetcare.fn_prevenir_borrado_persona()
RETURNS TRIGGER AS $$
BEGIN
    -- Revisar si la persona tiene mascotas o facturas
    IF EXISTS (SELECT 1 FROM vetcare.mascota WHERE persona_id = OLD.id) OR
       EXISTS (SELECT 1 FROM vetcare.factura WHERE persona_id = OLD.id) THEN
        RAISE EXCEPTION 'No se puede borrar la persona con ID % porque tiene mascotas
o facturas asociadas.', OLD.id;
    END IF;
    RETURN OLD;
END;
$$ LANGUAGE plpgsql;

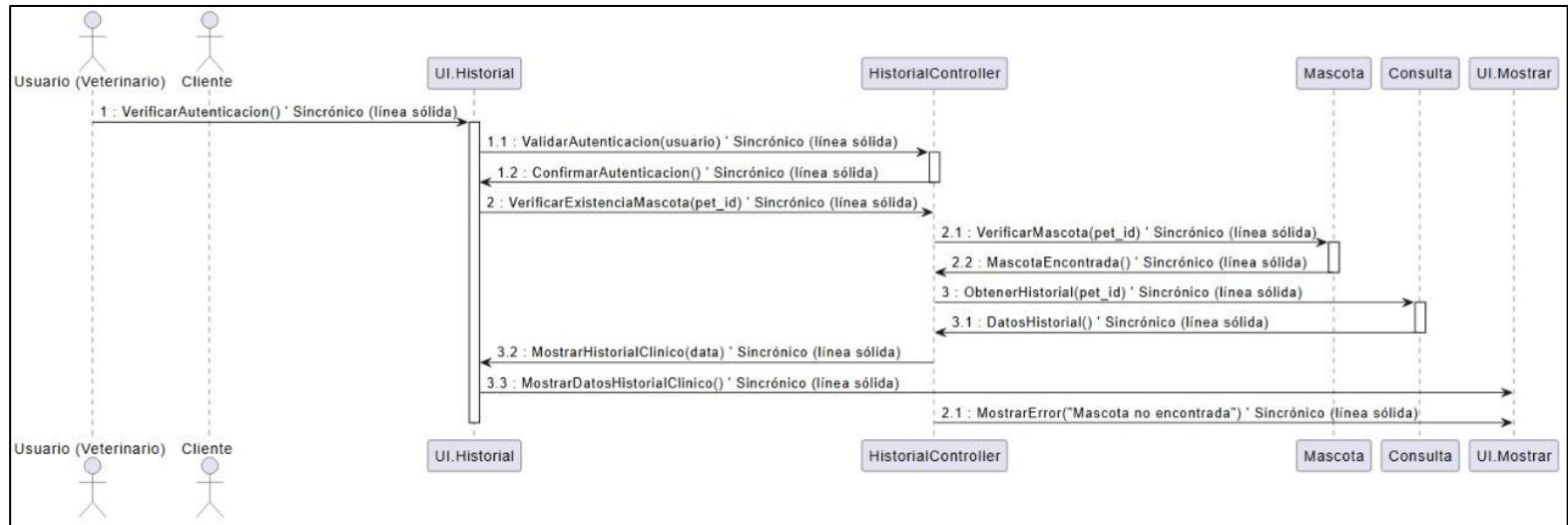
-- Asociar el trigger a la tabla persona
CREATE TRIGGER trg_prevenir_borrado_persona
BEFORE DELETE ON vetcare.persona
FOR EACH ROW
EXECUTE FUNCTION vetcare.fn_prevenir_borrado_persona();

```

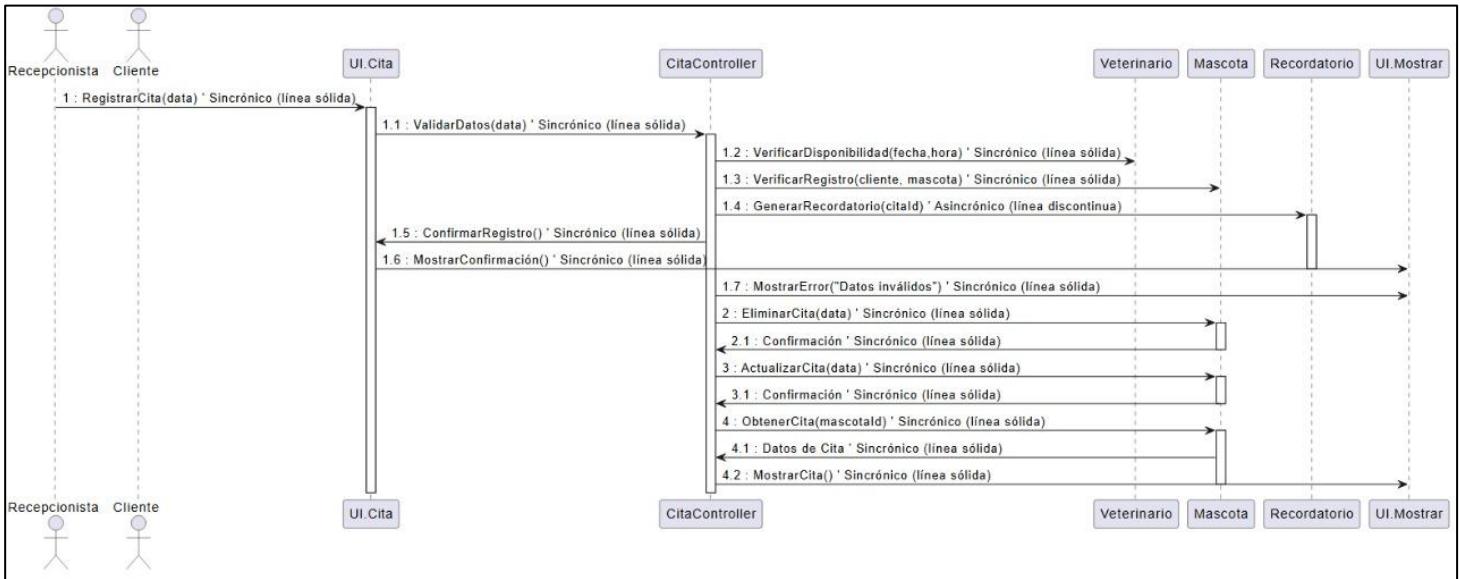
4.3.- Diseño de Caso de Uso

Diagrama de Secuencia

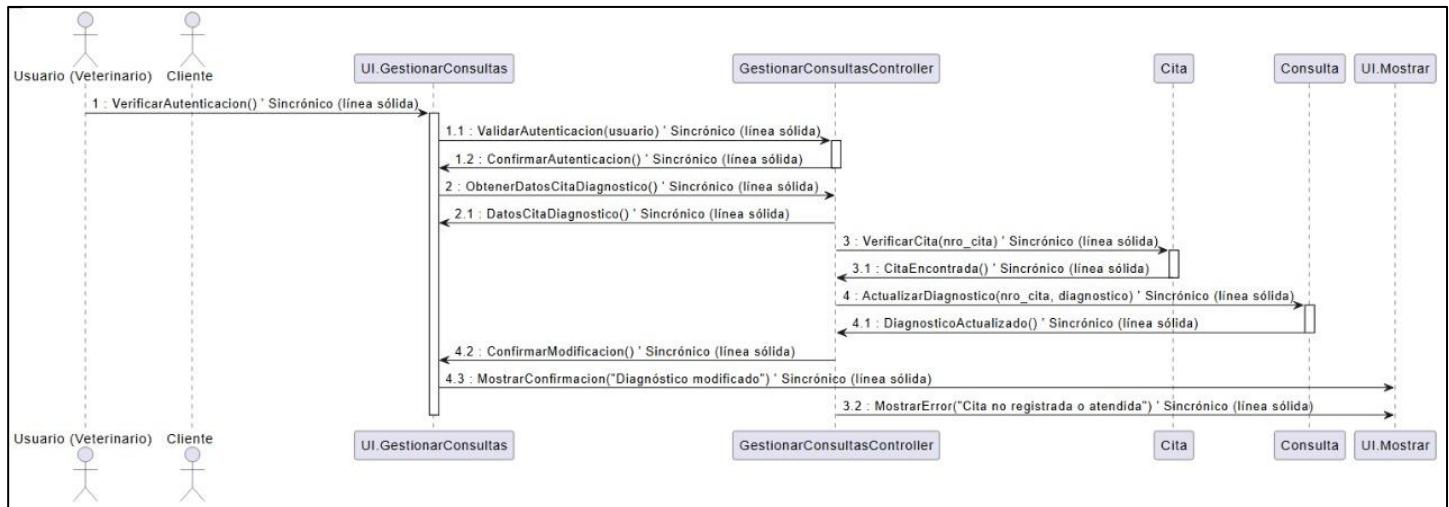
CU5.- Ver y Actualizar Historial Clínico



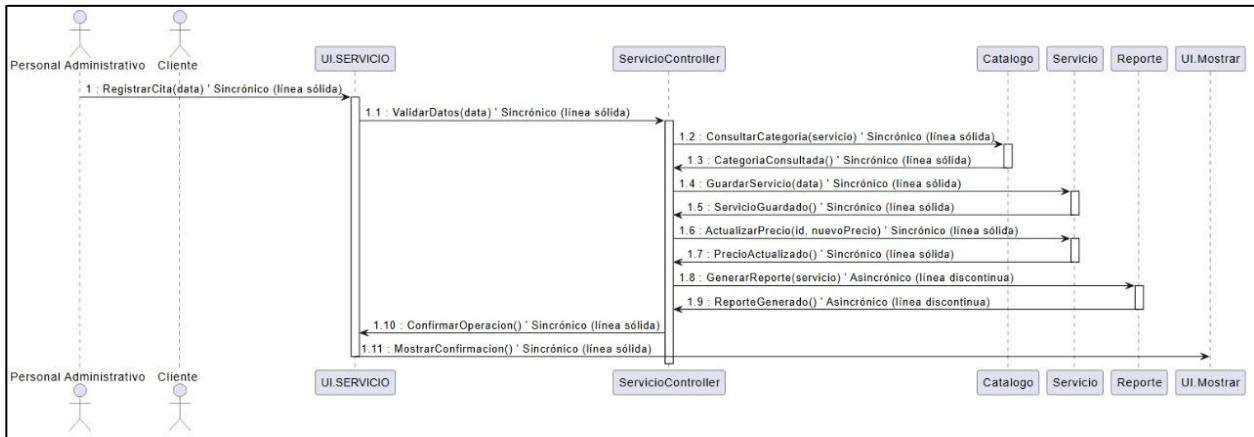
CU6.- Agendar Citas



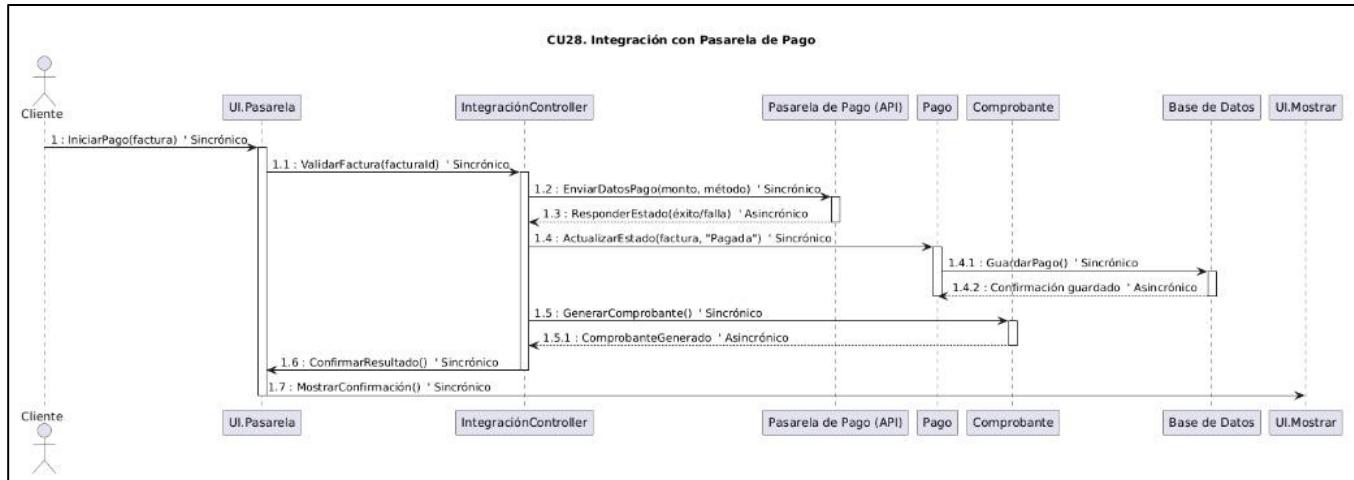
CU8.- Gestionar Consultas y Tratamientos



CU10.- Gestionar Servicios y Precios

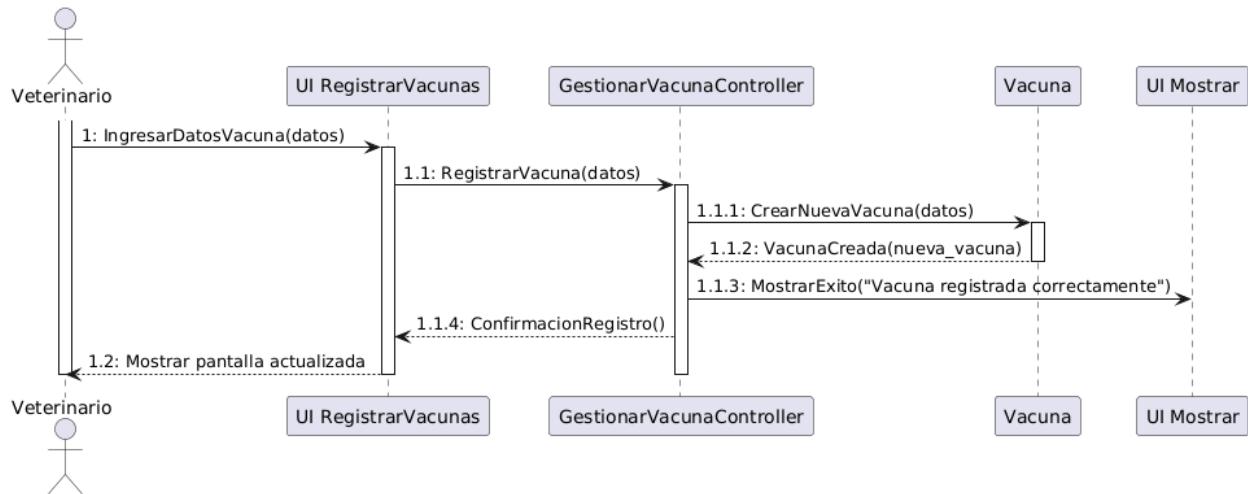


CU28.- Integración con Pasarela de Pago

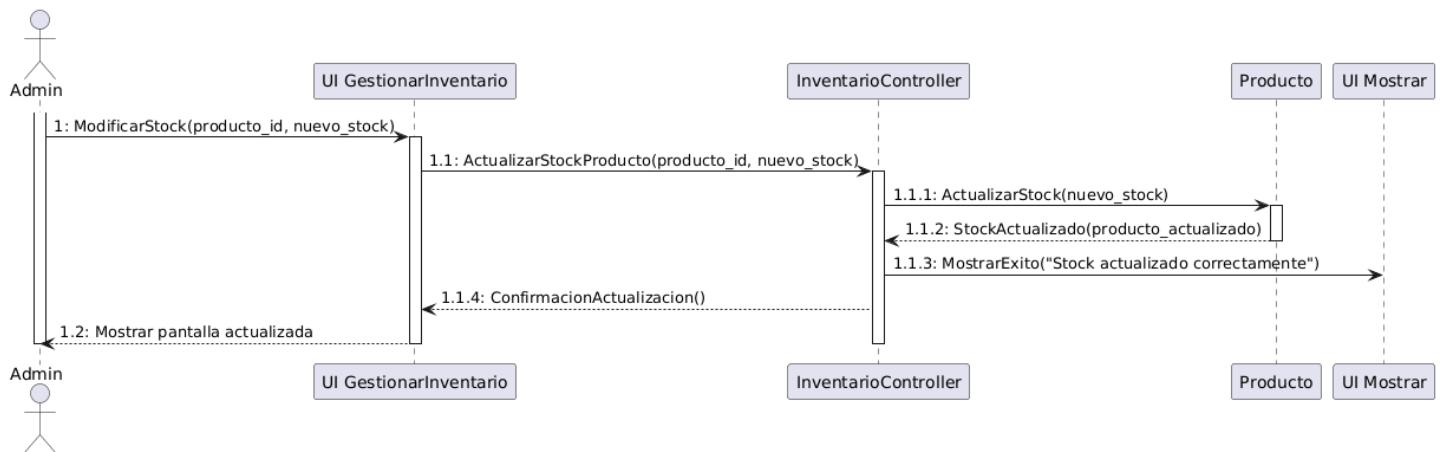


CICLO #3:

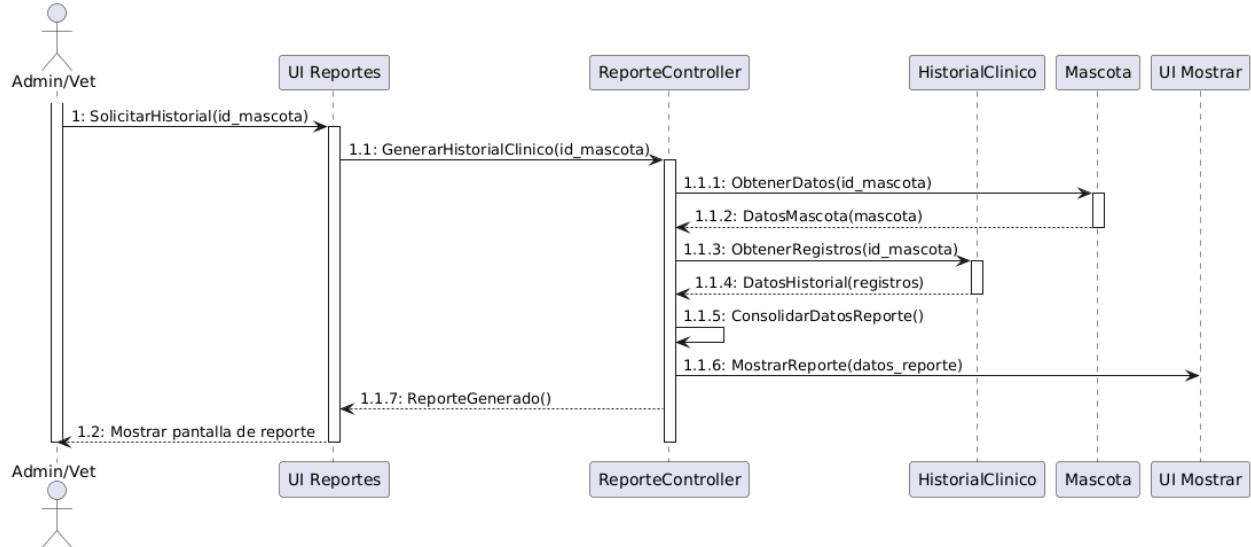
CU7: Registrar Vacunas (Flujo Exitoso)



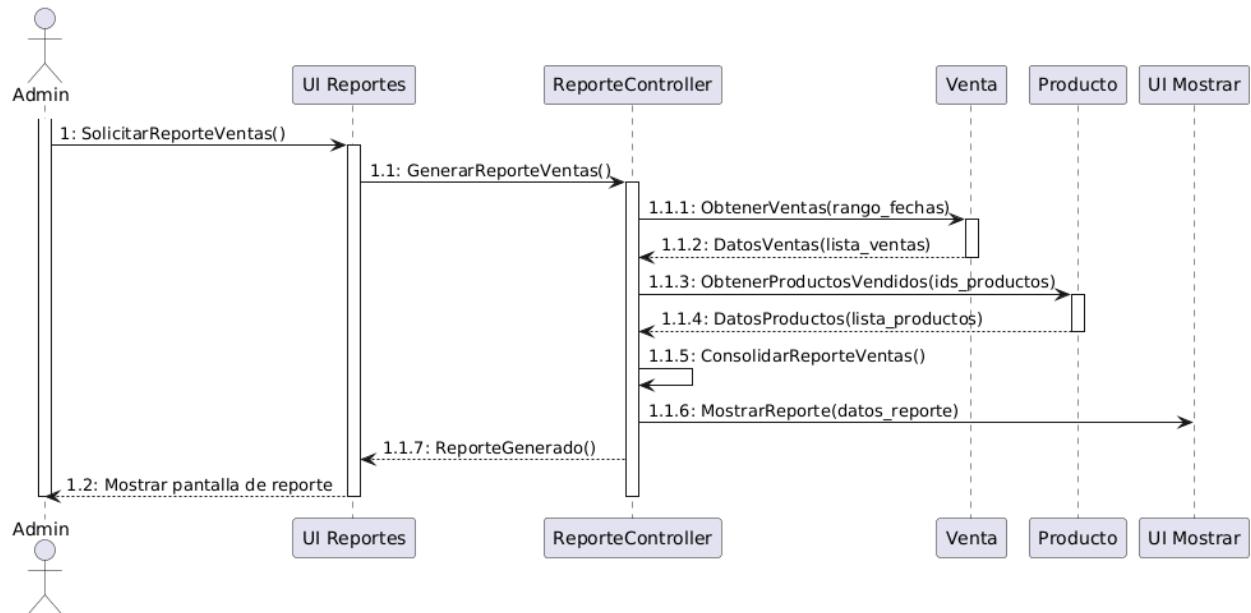
CU9: Gestionar inventario



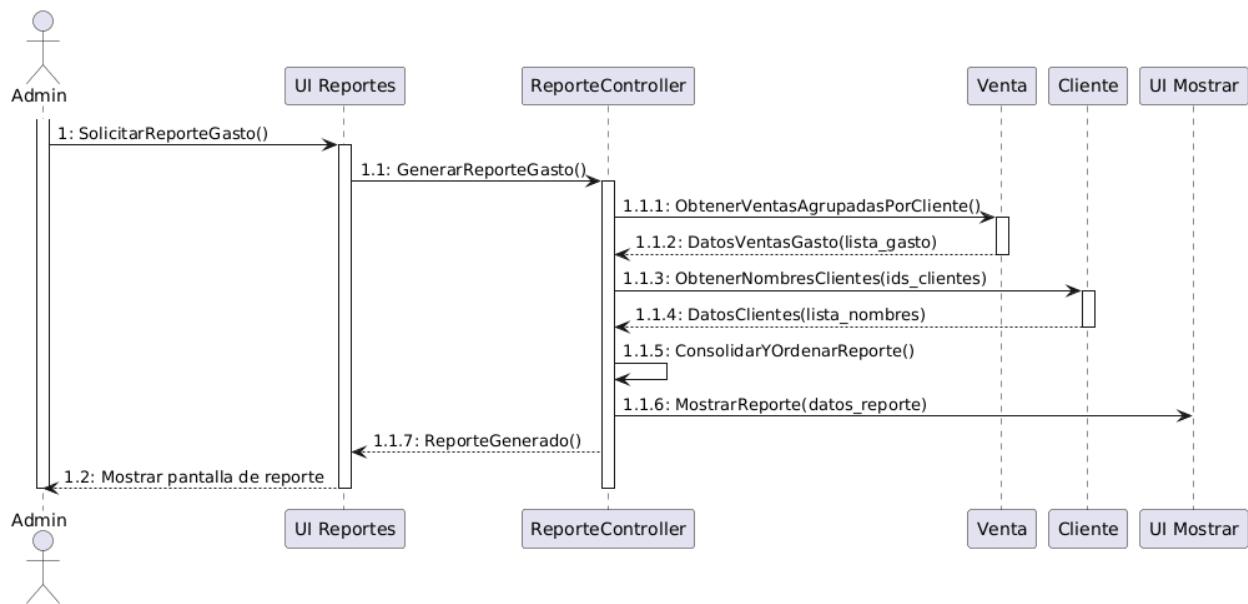
CU12: Reporte de Historial Clínico



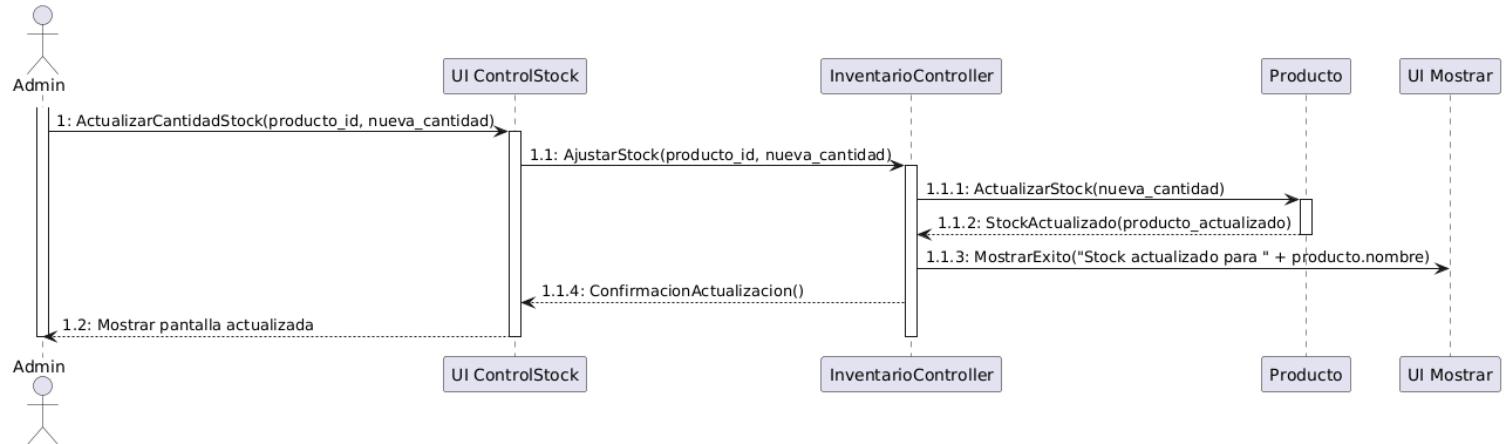
CU13: Reportes de Ventas y Productos



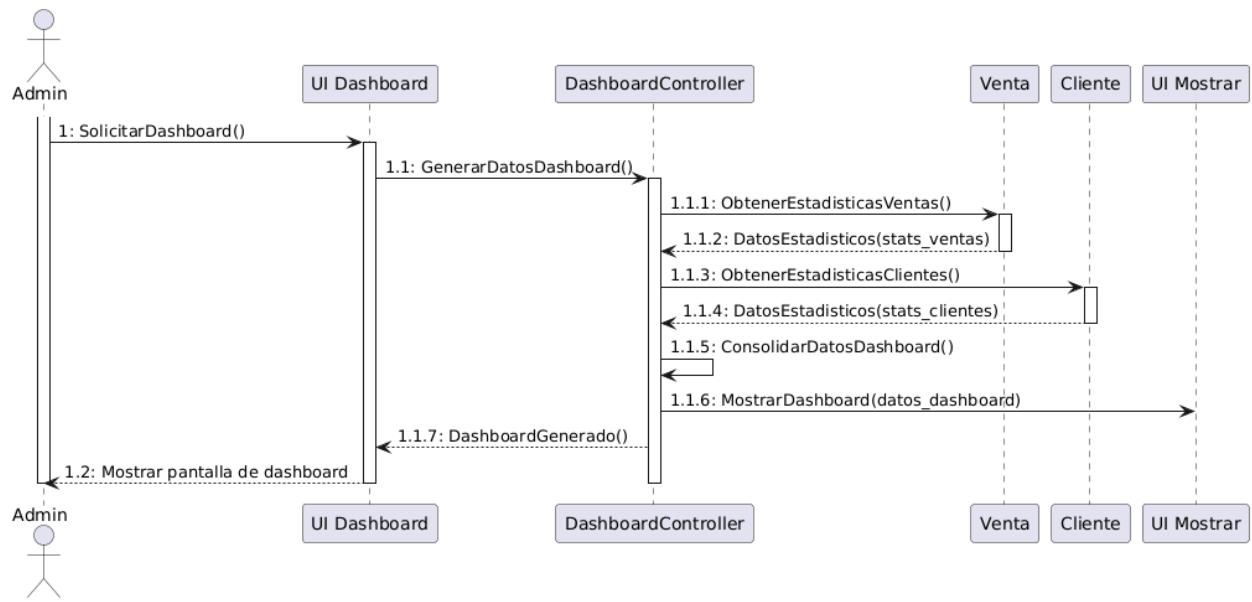
CU14: Reporte de Clientes con Mayor Gasto



CU18: Control de Stock



CU19: Acceder a Dashboard Estadístico



CAPITULO 5.- Flujo de Trabajo: Implementación

5.1 Elección de plataforma de desarrollo del Software

5.1.1 Lenguaje de programación

Para el desarrollo del backend de nuestro proyecto VetCare, optamos por utilizar Python como lenguaje principal, en conjunto con el framework Flask, debido a su ligereza, flexibilidad y facilidad de integración con otras tecnologías. Flask nos permite construir una arquitectura modular basada en controladores y rutas, lo que facilita la organización del código y la escalabilidad de la aplicación. Asimismo, se integró con la base de datos PostgreSQL, aprovechando la compatibilidad de Flask con librerías como SQLAlchemy para la gestión eficiente de las operaciones CRUD y la conexión mediante variables de entorno seguras.

Para la implementación del backend en la nube, utilizamos la plataforma Render, la cual ofrece una infraestructura confiable y automatizada que facilita el despliegue, monitoreo y escalabilidad del servicio. Render permite administrar los procesos del servidor Flask y la base de datos PostgreSQL de manera unificada, garantizando estabilidad y rendimiento óptimo en la ejecución del sistema.

En cuanto al desarrollo del frontend, empleamos React como biblioteca principal para la construcción de interfaces dinámicas y reutilizables. React nos proporciona una estructura basada en componentes que mejora la mantenibilidad del proyecto. Complementamos el desarrollo con Tailwind CSS, un framework de diseño moderno que permite aplicar estilos de manera rápida, eficiente y altamente personalizable. Para el proceso de compilación y despliegue utilizamos Vite, herramienta que optimiza la velocidad de renderizado y facilita la integración entre las distintas capas del sistema.

Finalmente, para el alojamiento del frontend se seleccionó la plataforma Netlify, que ofrece una infraestructura de despliegue automatizada y soporte para aplicaciones de página única (SPA, por sus siglas en inglés). Netlify permite una integración directa con proyectos React y asegura una entrega continua, escalable y segura, garantizando una experiencia fluida para el usuario final.

5.1.2 Base de Datos

Para el sistema VetCare, se eligió PostgreSQL como sistema de gestión de base de datos (SGBD) debido a su alto rendimiento, estabilidad, seguridad y compatibilidad multiplataforma. PostgreSQL es una base de datos relacional de código abierto ampliamente reconocida por su fiabilidad y capacidad para manejar operaciones complejas de manera eficiente. Su robustez y flexibilidad la convierten en una opción ideal para proyectos que requieren integridad de datos, consistencia y soporte a gran escala.

La integración con el backend desarrollado en Flask se realizó mediante el uso de la librería SQLAlchemy, la cual proporciona un potente ORM que permite interactuar con la base de datos de manera sencilla y estructurada. Gracias a SQLAlchemy, es posible definir las entidades del sistema como clases de Python, realizar consultas SQL seguras, manejar relaciones entre tablas y facilitar la portabilidad del código entre entornos de desarrollo y producción.

A continuación, se destacan algunas ventajas clave de PostgreSQL y su integración con SQLAlchemy:

- **Escalabilidad:**

PostgreSQL es altamente escalable, tanto a nivel de datos como de conexiones simultáneas. Esto permite manejar un gran volumen de registros —por ejemplo, historiales clínicos, consultas o facturas— sin comprometer el rendimiento. Esta característica resulta fundamental para garantizar la expansión del sistema a medida que crece la cantidad de clientes o información almacenada.

- **Seguridad:**

PostgreSQL incluye múltiples mecanismos de seguridad, como autenticación por roles, cifrado de contraseñas y control de acceso por usuario. Además, mediante SQLAlchemy se implementan consultas parametrizadas que previenen ataques de inyección SQL, asegurando la integridad y confidencialidad de los datos almacenados.

- **Sopporte JSON:**

PostgreSQL ofrece soporte nativo para el almacenamiento y manipulación de datos en formato JSON y JSONB, lo que permite manejar estructuras semiestructuradas, como históricos o configuraciones del sistema. Esta funcionalidad brinda flexibilidad en el diseño de la base de datos y facilita la interoperabilidad con el frontend desarrollado en React.

- **Extensibilidad:**

PostgreSQL es altamente extensible mediante funciones definidas por el usuario, extensiones o triggers personalizados. En VetCare, esto permite agregar validaciones automáticas, auditorías o cálculos internos dentro de la base de datos, optimizando los procesos sin sobrecargar la lógica del servidor Flask.

La base de datos fue alojada en la nube utilizando Render, aprovechando su compatibilidad con PostgreSQL gestionado. Esto permite contar con un entorno seguro, con respaldos automáticos, monitoreo continuo y escalabilidad dinámica, garantizando la disponibilidad y el rendimiento del sistema en todo momento.

5.1.3 Sistemas Operativos

Como sistema de información basado en web con tecnología responsive, nuestro sistema está diseñado para adaptarse a cualquier pantalla y ejecutarse en cualquier sistema operativo. Esto significa que nuestra aplicación es compatible no sólo con Windows, sino también con macOS, Linux, Android e iOS. El diseño responsive ayuda a que la interfaz de usuario se ajuste automáticamente al tamaño de la pantalla de su computadora de escritorio, tableta o teléfono inteligente. Esto proporciona una experiencia de usuario consistente y accesible, permitiéndole interactuar con el sistema de manera eficiente y sin problemas, independientemente del dispositivo que utilice. Además, la compatibilidad con múltiples sistemas operativos asegura que nuestra aplicación pueda usarse en una variedad de entornos y contextos, facilitando la accesibilidad y usabilidad para muchos usuarios. Al admitir sistemas operativos tanto de escritorio como móviles, garantizamos que nuestros sistemas de información sean flexibles y adaptables a las necesidades de una amplia gama de usuarios y casos de uso diferentes. Este enfoque también permite a

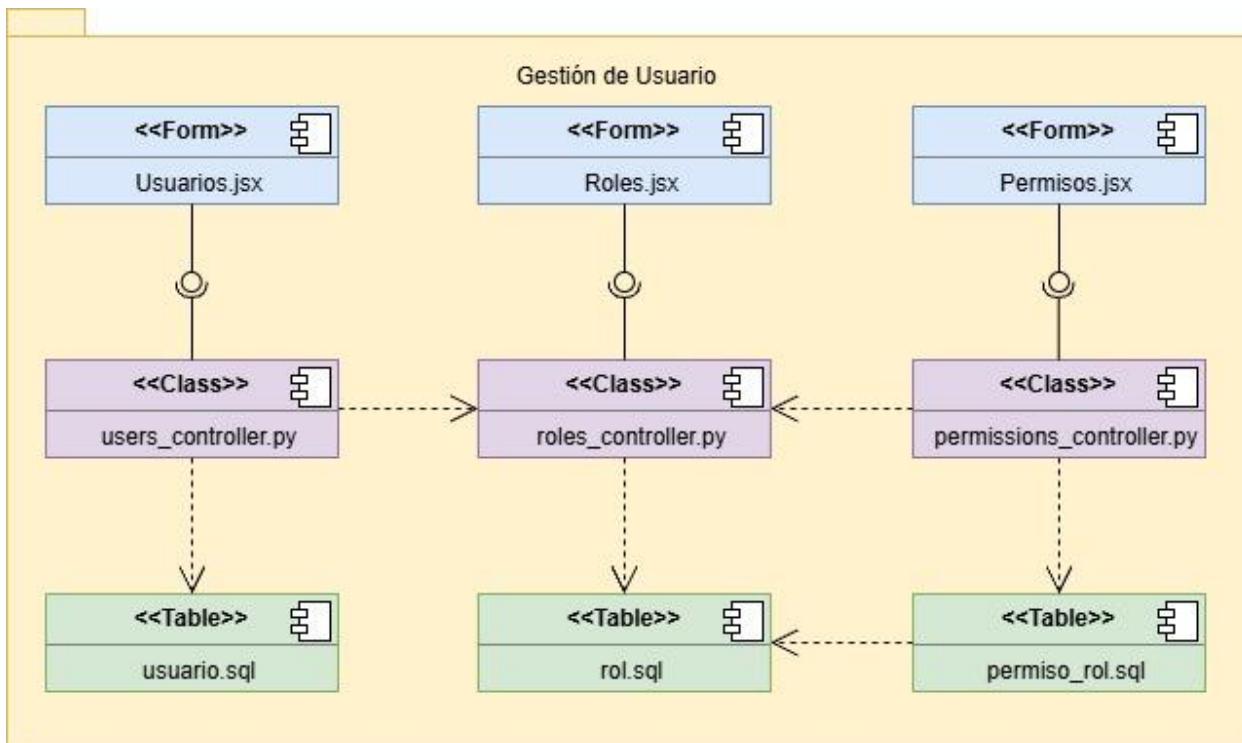
las organizaciones implementar nuestra solución sin preocuparse por la compatibilidad del sistema operativo porque nuestras aplicaciones están diseñadas para funcionar de manera óptima en cualquier plataforma. Esto no solo mejora la accesibilidad y la experiencia del usuario para los usuarios finales, sino que también simplifica el proceso de implementación y reduce los costos asociados con la administración de múltiples versiones de software en diferentes sistemas operativos.

5.1.4 Otros

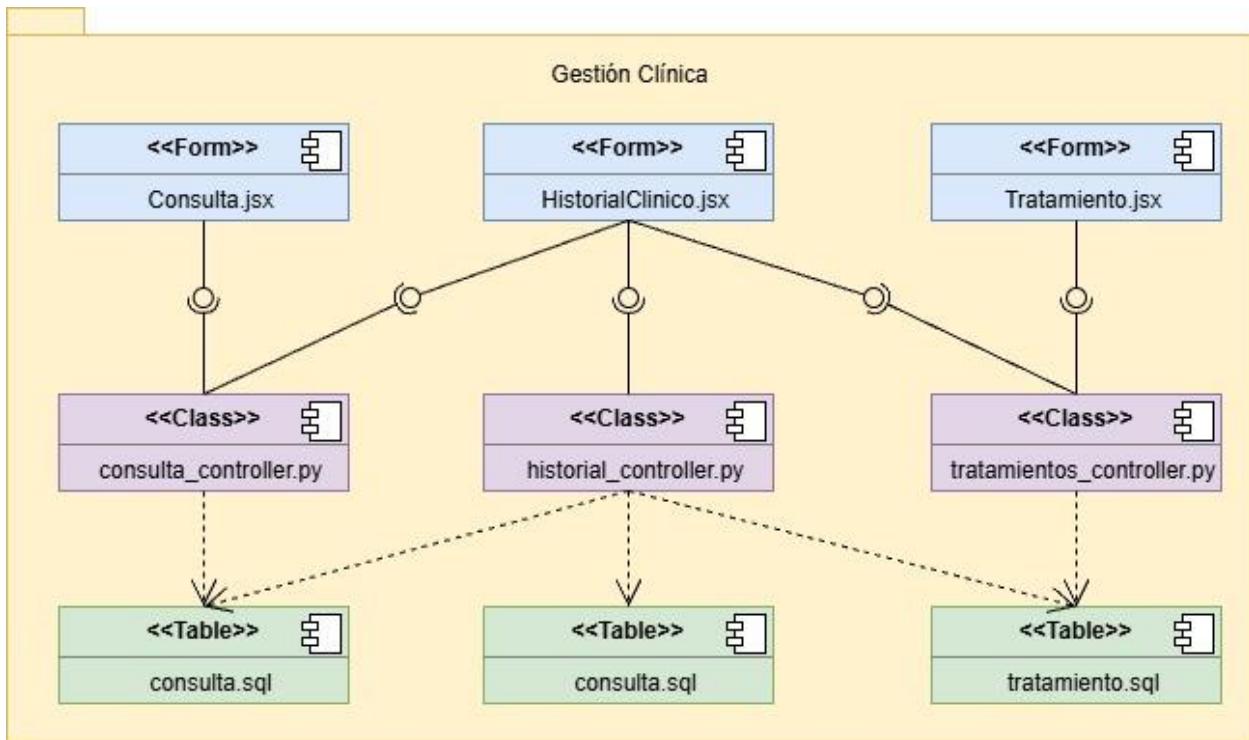
Se designó GitHub como plataforma central para gestionar nuestro código fuente por su control de versiones con Git, facilitando la colaboración y revisión de código. Además, ofrece seguimiento de problemas y la integración continua con servicios de CI. Su comunidad y recursos disponibles también son valiosos para el desarrollo del proyecto. GitHub proporciona una solución integral para la gestión del ciclo de vida del desarrollo de software.

5.2.- Arquitectura Subsistema

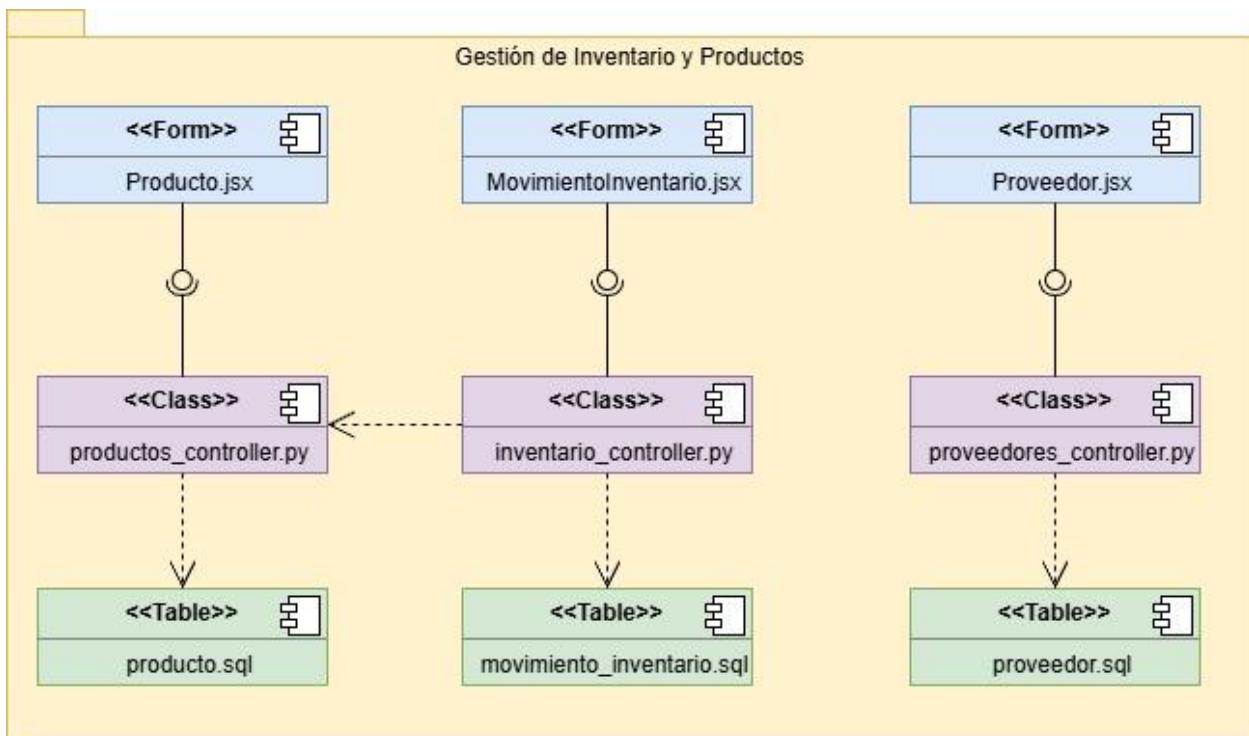
P1.- GESTION DE USUARIO



P2.- GESTION CLINICA



P3.- GESTION DE INVENTARIO Y PRODUCTOS



CONCLUSIÓN

El proyecto “Sistema de Información VetCare” constituye una propuesta integral para transformar la gestión administrativa, clínica y comunicacional de la veterinaria, respondiendo a la creciente demanda de servicios más eficientes y modernos. A través de la digitalización y automatización de los procesos, se logra superar las limitaciones de los métodos manuales, reduciendo errores, optimizando recursos y garantizando mayor seguridad en la información.

La solución planteada integra módulos clave como gestión de citas, historias clínicas digitales, control de inventarios, facturación y comunicación con clientes, ofreciendo un entorno centralizado que mejora la eficiencia operativa, eleva la calidad del servicio y fortalece la competitividad de las clínicas en el mercado actual. Además, su diseño modular y escalable permite una futura expansión hacia servicios más avanzados, como pagos en línea o integración con plataformas externas.

RECOMENDACIÓN

A pesar de que el sistema VetCare representa una solución a los problemas identificados, se recomienda fuertemente fortalecer su implementación, toda vez que solo se abordaron las áreas más críticas, quedando aún desafíos en la adopción del personal, la migración completa de los datos y la consolidación de los servicios digitales.

Las recomendaciones que se pueden realizar para futuros proyectos basados en este están referidas a continuación:

- Implementar un plan de capacitación continua para que el personal administrativo y médico se familiarice con el sistema y se reduzca la resistencia al cambio.
- Realizar la migración de históricos clínicos y registros de forma progresiva mediante un cronograma por etapas que asegure la calidad de los datos.
- Establecer políticas estrictas de seguridad informática, como copias de respaldo, control de accesos y encriptación de datos sensibles.
- Programar mantenimientos periódicos y actualizaciones del software para garantizar la estabilidad, escalabilidad y adaptación del sistema a futuras necesidades.
- Fortalecer la comunicación digital con los clientes mediante recordatorios automáticos, notificaciones personalizadas y un portal de autoservicio.
- Evaluar periódicamente el impacto del sistema en cuanto a ahorro de tiempo, reducción de errores y satisfacción de clientes para justificar nuevas mejoras.

ANEXO

- Entrevistas

Objetivo de la entrevista: Recopilar información sobre el funcionamiento actual de la clínica, identificar los problemas de la falta de un sistema de información y definir los requisitos clave para el diseño y desarrollo de un sitio web funcional.

Entrevista #1

Fecha	23/08/2025
Duración	20 min
Lugar	Clinica Veterinaria ServertCorp
Dirección	5to Anillo - Radial 13. C/ San Antonio

Datos de la empresa	
Nombre	ServertCorp
Tipo	Privada
Datos del entrevistado	
Nombre	Carlos Guzman