PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

FACULTAD DE HÁBITAT, INFRAESTRUCTURA Y CREATIVIDAD

CARRERA DE INGENIERÍA EN SISTEMAS DE INFORMACIÓN



Plan de Titulación

Tema: Prototipo de Sistema de información para la gestión de consultorios veterinarios

TUTOR:

ING. Miguel Dimitri Ortiz Navarrete

AUTOR:

Daniel Sebastián Calvopiña Orellana

QUITO DM, 2026

DEDICATORIA

Dedico este trabajo de titulación a mi familia que estuvieron pendiente de mis estudios, estuvieron dándome apoyo en cada momento de mi carrera, a mi novia que fue una parte fundamental en estos 4 años de mi vida.

AGRADECIMIENTO

Agradezco a mis profesores por sus enseñanzas en estos 4 años

Agradezco a mi familia por su atención, acompañamiento y ayuda en a lo largo de mi carrera.

Agradezco a mis compañeros por su ayuda

Agradezco a mi novia por ser mi guía y mi compañera estos años que han sido los mas importantes de mi vida.

RESUMEN

El avance tecnológico ha transformado la manera en que se gestiona la información en distintos sectores, incluyendo la atención veterinaria. Por esta razón, decidí desarrollar un prototipo de sistema de información que permita centralizar y digitalizar los procesos operativos y clínicos de los consultorios veterinarios, facilitando el registro de pacientes, la gestión de citas y el seguimiento del historial médico de cada animal.

Este prototipo se desarrollará bajo la metodología ágil Scrum, que organiza el trabajo en sprints iterativos y permite incorporar mejoras de forma continua, asegurando que el producto final responda a las necesidades reales del consultorio. El desarrollo se realizará utilizando C# con ASP.NET en el backend, React con TypeScript en el frontend, y PostgreSQL como motor de base de datos, incluyendo replicación lógica entre una instancia local y una en la nube.

El proyecto surge en respuesta a una problemática real identificada en Quito: según datos del INEC, el Distrito Metropolitano alberga más de 1,08 millones de perros y gatos, lo que genera una demanda constante de servicios veterinarios. Sin embargo, muchos consultorios de pequeña escala aún gestionan sus registros de forma manual o con herramientas dispersas, lo que provoca pérdida de trazabilidad clínica, duplicidad de datos y dificultades para dar seguimiento a tratamientos prolongados. De esta necesidad nace la idea de construir una herramienta tecnológica que centralice la información, automatice la gestión de citas y garantice una atención más organizada, continua y profesional para cada paciente.

INDICE

DEDICATORIA 2

AGRADECIMIENTO 3

RESUMEN 4

ÍNDICE DE TABLAS 11

ÍNDICE DE FIGURAS 12

1 CAPÍTULO I: INTRODUCCIÓN 13

1.1 Introducción 13

1.2 Planteamiento de los problemas 14

1.3 Justificación 15

1.4 Objetivos 16

1.4.1 Objetivo general 16

1.4.2 Objetivos específicos 16

1.5 Alcance y limitaciones 16

1.5.1 Alcance 16

1.5.2 Limitaciones 17

2 Capítulo II: Marco Teórico y Conceptual 18

2.1 Gestión de la información en salud animal 18

2.2 Conceptos fundamentales del ámbito clínico veterinario 20

2.2.1 Actores del proceso clínico 21

2.2.2 Historia clínica y proceso de consulta 22

2.2.3 Medicina preventiva 25

2.2.4 Parámetros fisiológicos del examen físico 26

2.2.5 Áreas de especialidad clínica 29

2.3 Fundamentos de desarrollo de software 30

2.3.1 Metodología de desarrollo ágil: Scrum 31

2.3.2 Metodologías ágiles en el desarrollo de software 32

2.3.3 Ciclo de vida del software 33

2.3.4 Frontend 34

2.3.5 Backend 34

2.3.6 Framework de desarrollo 35

2.3.7 Base de datos relacional, procedimientos almacenados, triggers, usuarios y roles 35

2.3.8 Replicación de base de datos 37

2.4 Herramientas tecnológicas 37

2.4.1 Herramientas de desarrollo frontend 38

2.4.2 Herramientas de desarrollo backend 39

2.4.3 Motor de base de datos 40

3 CAPÍTULO 3: Metodología 41

3.1 Metodología de investigación 41

3.1.1 Enfoque de la investigación....41
3.1.2 Técnicas e instrumentos de recolección de información....42
3.2 Metodología de desarrollo del prototipo....45
3.2.1 Marco de Trabajo....45
3.2.2 Integración del ciclo de vida del software con Scrum....46
3.2.3 Levantamiento de requerimientos....49
3.2.4 Diseño del sistema....49
3.2.5 Justificación de las herramientas tecnológicas....50
4 CAPÍTULO IV: DESARROLLO....53
4.1 Levantamiento de información....53
4.1.1 Entrevista....53
4.1.2 Observación directa....57
4.1.3 Evidencia documental....60
4.2 Identificación de problemas....62
4.2.1 Problemas detectados en el consultorio....63
4.2.2 Necesidades del sistema....65
4.3 Requerimientos del sistema....68
4.3.1 Requerimientos funcionales....68
4.3.2 Requerimientos no funcionales....70
4.4 Product Backlog....71

4.4.1 Objetivo del producto 72
4.4.2 Criterios de priorización 73
4.4.3 Lista de elementos del Product Backlog 73
4.4.4 Historias de usuario principales 74
4.4.5 Refinamiento del Product Backlog 74
4.5 Planificación de sprints y Sprint Backlog 76
4.5.1 Diferencia aplicada entre Product Backlog y Sprint Backlog 76
4.5.2 Desarrollo planificado por sprint 77
4.5.3 Sprint 4: 1 de junio al 22 de junio de 2026 (tres semanas) 81
4.5.4 Resumen de planificación de Sprint Backlog 83
4.6 Modelado de procesos 84
4.6.1 Proceso actual de atención en el consultorio 85
4.6.2 Proceso propuesto con el sistema 86
4.6.3 Flujo de atención clínica propuesto 88
4.6.4 Flujo de agendamiento de citas 88
4.7 Desarrollo y validación funcional del prototipo 89
4.7.1 Sprint 1: base del sistema 90
4.7.2 Sprint 2: configuración del consultorio 92
4.7.3 Sprint 3: procesos principales y portal 96
4.7.4 Sprint 4: historial clínico 102

4.7.5 Integración de procedimientos almacenados y triggers en el desarrollo . 110
4.7.6 Cierre del desarrollo del prototipo 110
4.8 Arquitectura del sistema 111
4.8.1 Arquitectura lógica 112
4.8.2 Arquitectura física 113
5 CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES 115
5.1 Conclusiones 115
5.2 Recomendaciones 117
6 BIBLIOGRAFÍA 119
8 ANEXOS 124
8.1 Anexo 1. Guía de entrevista semiestructurada 124
8.2 Anexo 2. Guía de observación directa 126
8.3 Anexo 3. Ficha clínica general del consultorio 127
8.4 Anexo 4. Hoja de seguimiento de tratamientos 128
8.5 Anexo 5. Formulario de catamnesis oncológica 129
8.6 Anexo 6. Hoja de ubicación anatómica 130
8.7 Anexo 7. Modelo entidad-relación completo 131
8.8 Anexo 8. Historias de usuario del prototipo 132
8.8.1 Historia de usuario HU-01 132
8.8.2 Historia de usuario HU-02 132

8.8.3 Historia de usuario HU-03.....132

8.8.4 Historia de usuario HU-04.....133

8.8.5 Historia de usuario HU-05.....133

8.8.6 Historia de usuario HU-06.....133

8.8.7 Historia de usuario HU-07.....134

8.8.8 Historia de usuario HU-08.....134

8.8.9 Historia de usuario HU-09.....134

ÍNDICE DE TABLAS

Tabla 1. Requerimientos funcionales del sistema **69**

Tabla 2. Requerimientos no funcionales del sistema **70**

Tabla 3. Listado de elementos de Product Backlog **75**

Tabla 4. Planificación de Sprints y Sprint Backlog del prototipo **83**

Tabla 5. Proceso actual de atención veterinaria **86**

Tabla 6. Proceso propuesto con el prototipo **87**

Tabla 7. Flujo de atención clínica propuesto **88**

Tabla 8. Flujo de agendamiento de citas **89**

Tabla 9. Desarrollo y validación del módulo de autenticación **91**

Tabla 10. Desarrollo y validación del módulo de parametrización **93**

Tabla 11. Desarrollo y validación del módulo de Veterinarios **94**

Tabla 12. Desarrollo y validación del módulo de catalogo **96**

Tabla 13. Desarrollo y validación del módulo de clientes y pacientes **97**

Tabla 14. Desarrollo y validación del módulo de agenda y citas **99**

Tabla 15. Desarrollo y validación del portal público **101**

Tabla 16. Desarrollo y validación del módulo de historial clínico **104**

Tabla 17. Desarrollo y validación de prescripciones, exámenes y tratamientos **105**

Tabla 18. Desarrollo y validación de especialidades clínicas **108**

Tabla 19. Validación funcional de reglas implementadas en base de datos **110**

Tabla 20. Componentes de la arquitectura lógica **113**

Tabla 21. Organización física de los proyectos **113**

ÍNDICE DE FIGURAS

Figura 1 **92**

Figura 2 **93**

Figura 3 **95**

Figura 4 **96**

Figura 5 **97**

Figura 6 **98**

Figura 7 **100**

Figura 8 **100**

Figura 9 **102**

Figura 10 **103**

Figura 11 **105**

Figura 12 **106**

Figura 13 **106**

Figura 14 **107**

Figura 15 **108**

Figura 16 **109**

Figura 17 **109**

1 CAPÍTULO I: INTRODUCCIÓN

1.1 Introducción

En Ecuador, los animales de compañía se han consolidado como miembros esenciales de los hogares. Esta tendencia se refleja en datos del INEC (Censo 2022), que indican una convivencia generalizada: más de 2 millones de niños viven con perros o gatos (2.014.575), y 1.326.537 hogares con menores de 12 años tienen al menos una mascota. Apenas el 32,3% de estos hogares carece de ellas, lo que subraya una demanda constante y significativa de servicios de salud y cuidado animal.

Para satisfacer esta demanda, la atención primaria veterinaria se desarrolla principalmente en consultorios y clínicas privadas, cuya formalidad operativa está definida por requisitos sanitarios. La normativa exige que estos establecimientos cuenten con Registro y Permiso Sanitario de Funcionamiento otorgado por Agrocalidad. Además, la misma entidad formaliza la actividad a través de la figura del médico veterinario registrado para diversos trámites, demostrando la existencia de un marco regulatorio profesional.

A pesar de este marco, es frecuente observar que muchos consultorios manejan el registro de clientes, pacientes y la programación de citas mediante métodos manuales o poco sistematizados. Esta falta de integración tecnológica se convierte en un impedimento crítico para la continuidad clínica. Durante tratamientos prolongados o seguimientos postquirúrgicos, la ausencia de un historial clínico digital y centralizado ocasiona una pérdida de trazabilidad, dificultando seriamente el seguimiento oportuno y eficiente del paciente.

Por lo tanto, ante esta realidad, se hace indispensable implementar una herramienta tecnológica que no solo organice la operación interna del consultorio, sino que también centralice la información clínica, esta solución no solo optimizará el tiempo del personal, sino que garantizará que el historial de cada paciente esté siempre disponible y actualizado, fundamentando una atención profesional y continua.

1.2 Planteamiento de los problemas

En Quito, la alta convivencia con animales de compañía se traduce en una demanda sostenida de atención veterinaria, especialmente para controles preventivos, vacunación y tratamientos. De acuerdo con información difundida a partir de datos del INEC, en el Distrito Metropolitano existen más de 1,08 millones de perros y gatos, superando incluso al número de niños y adolescentes, lo que evidencia la magnitud del mercado y la necesidad de servicios ágiles y bien organizados.

Por otro lado, muchos consultorios veterinarios aún trabajan con registros manuales, herramientas básicas o sistemas incompletos y obsoletos que no cubren integralmente los procesos del consultorio. Esta falta de sistematización genera desorden operativo, limita la capacidad de atención y deriva en problemas recurrentes como duplicidad de datos y, principalmente, la fragmentación del historial clínico. Cuando la información no está centralizada, se dificulta dar seguimiento al paciente ya que no se registra con claridad su evolución y se pierde trazabilidad en tratamientos prolongados o controles postquirúrgicos, y el retorno del paciente depende más de la memoria o de anotaciones aisladas que de un flujo ordenado.

Además, la ausencia de automatización incrementa la probabilidad de omisiones en recordatorios esenciales como son las vacunaciones, desparasitaciones y seguimientos, lo que puede afectar directamente el bienestar animal y la continuidad del cuidado. En este contexto, surge la necesidad de desarrollar un sistema de información que centralice el registro de propietarios, mascotas e historial de tratamientos, y que automatice la gestión de citas y recordatorios, reduciendo errores del manejo manual y garantizando una atención más profesional, continua y trazable para cada paciente.

1.3 Justificación

La atención veterinaria integral no se limita a consultas puntuales; también abarca un conjunto amplio de acciones preventivas y terapéuticas, entre ellas la vacunación, la desparasitación, el monitoreo postquirúrgico y el control de tratamientos de corta o larga duración. En estos casos, el seguimiento periódico es fundamental para evaluar la evolución del paciente y mantener su estado de salud.

En Ecuador, la alta presencia de mascotas en los hogares evidencia una demanda sostenida de estos servicios profesionales. Sin embargo, cuando la gestión de información se realiza de forma manual o fragmentada, suelen presentarse inconsistencias, pérdida de datos evolutivos y omisiones en la programación de controles, lo que compromete la continuidad y la calidad de la atención veterinaria.

Por ello, la implementación de un sistema de información que centralice la gestión médica y operativa resulta indispensable para un consultorio moderno. Esta herramienta se justifica por su capacidad de estandarizar el registro clínico, asegurar la trazabilidad del

paciente. Al reducir la dependencia de procesos informales, el sistema optimiza la carga administrativa y facilita una supervisión médica continua, permitiendo elevar la calidad del servicio y fortalecer los estándares de cuidado y salud animal.

1.4 Objetivos

1.4.1 Objetivo general

Desarrollar un prototipo de sistema de información para la gestión de consultorios veterinarios, orientado a la administración integral de la consulta médica.

1.4.2 Objetivos específicos

- Identificar los procesos, requerimientos funcionales y no funcionales del consultorio, mediante el análisis del flujo de trabajo actual y entrevistas con los usuarios.
- Diseñar la arquitectura del sistema y el modelo de datos necesario para soportar la gestión administrativa y el historial clínico de los pacientes.
- Desarrollar el prototipo del sistema de información integrando los módulos de gestión de usuarios, expedientes médicos y control de citas.
- Validar la funcionalidad del prototipo mediante la ejecución de pruebas para verificar el correcto registro, seguimiento y seguridad de la información.

1.5 Alcance y limitaciones

1.5.1 Alcance

El presente proyecto de titulación finalizará con la entrega de un prototipo de sistema de información funcional para la gestión de consultorios veterinarios. Esta aplicación incluirá los siguientes módulos y funcionalidades:

- **Gestión de Usuarios y Seguridad:** Control de acceso mediante autenticación, con perfiles definidos para Administrador, Médico Veterinario y Personal de Recepción.
- **Gestión de Propietarios y Pacientes:** Administración del expediente digital de la mascota y los datos de contacto del dueño.
- **Gestión de citas y agenda:** Programación de turnos, validación de conflictos de horario y gestión de estados.
- **Gestión Clínica Integral:** Registro detallado de la consulta médica, así como el historial de vacunación, desparasitación e historial clínico del paciente.

Entregables documentales:

- Documento de especificación de requisitos del sistema.
- Diseño de arquitectura y modelo de base de datos.
- Registro de pruebas funcionales y validación del prototipo

1.5.2 Limitaciones

El sistema se limita a los procesos operativos y clínicos descritos anteriormente. Por lo tanto, quedan excluidos del alcance del proyecto:

- Módulos de facturación electrónica, contabilidad o integración con el SRI.
- Control de inventario de medicamentos, insumos médicos o tienda de mascotas.
- Integración con hardware externo
- La implementación en un entorno de producción real a gran escala, entregándose el producto en un ambiente de pruebas funcional.
- Pruebas avanzadas de seguridad informática y análisis de vulnerabilidades complejos.

2 Capítulo II: Marco Teórico y Conceptual

2.1 Gestión de la información en salud animal

La gestión de la información en salud se define como el conjunto de procesos orientados a la recolección, organización, almacenamiento, acceso y uso de datos clínicos con el propósito de apoyar la toma de decisiones médicas y garantizar la continuidad de la atención (Shortliffe & Cimino, 2014). En el contexto de la salud animal, esta disciplina adquiere una dimensión particular: a diferencia del paciente humano, el animal no puede comunicar verbalmente sus síntomas ni describir su evolución, condición que convierte al registro histórico de cada consulta en un instrumento clínico de primera necesidad. Cada anotación sobre peso corporal, temperatura, tratamiento previo o reacción a un fármaco constituye información que el médico veterinario requiere al momento de evaluar un cuadro clínico, establecer un diagnóstico diferencial o diseñar un plan terapéutico adecuado para el paciente.

El expediente clínico veterinario es el instrumento central de la gestión de información en salud animal. Su estructura convencional se apoya en la metodología SOAP es decir contiene un Subjetivo, Objetivo, Análisis y Plan, adoptada ampliamente en la práctica clínica para estandarizar el registro de consultas y facilitar la comunicación entre profesionales (Radostits et al., 2007). Mediante este esquema, se documentan tanto los datos aportados por el propietario como los hallazgos físicos obtenidos en el examen clínico, los resultados de pruebas diagnósticas y el tratamiento indicado. La coherencia y completitud de estos registros determinan la calidad de la atención que el paciente puede recibir en consultas posteriores, especialmente en enfermedades crónicas, recuperaciones postquirúrgicas o tratamientos prolongados que exigen un seguimiento periódico y documentado.

Un aspecto crítico en la gestión de información veterinaria es la trazabilidad clínica, entendida como la capacidad de reconstruir el historial de intervenciones realizadas sobre un paciente en cualquier punto de su línea de atención. En el ámbito de las vacunaciones y desparasitaciones, la Asociación Mundial de Veterinarios de Pequeños Animales (WSAVA) establece que un registro adecuado de los esquemas aplicados permite tanto verificar la inmunización activa del animal como planificar los refuerzos futuros con base en evidencia clínica documentada (WSAVA Vaccination Guidelines Group, 2015). La ausencia de este registro no solo afecta la eficacia preventiva, sino que puede comprometer directamente la seguridad del animal frente a enfermedades prevenibles, trasladando la responsabilidad hacia la memoria del propietario o del profesional en lugar de sustentarse en datos objetivos.

En la realidad operativa de muchos consultorios veterinarios de pequeña y mediana escala, la gestión de información ha dependido históricamente de registros físicos, fichas en papel o, en el mejor de los casos, hojas de cálculo desconectadas entre sí. Esta fragmentación documental genera consecuencias operativas significativas: duplicidad de datos, pérdida de información ante la rotación de personal, dificultad para recuperar historiales en situaciones de urgencia y ausencia de mecanismos de alerta para controles pendientes. Weed (1969), pionero del sistema de registro médico orientado al problema, ya señalaba que la organización sistémica de la información clínica es condición necesaria para una atención coherente y continua, principio que mantiene plena vigencia en el contexto veterinario contemporáneo y que sustenta la necesidad de soluciones tecnológicas integrales.

La transición hacia sistemas de información digitales en la atención veterinaria responde, precisamente, a la necesidad de superar estas limitaciones estructurales. Un sistema de información bien diseñado centraliza en un único repositorio los datos del propietario, el

expediente clínico del paciente, el historial de vacunaciones, los tratamientos activos y las citas programadas, garantizando su disponibilidad inmediata y su integridad a lo largo del tiempo (Pressman & Maxim, 2019). Esta centralización no solo reduce la carga administrativa del personal, sino que potencia la capacidad diagnóstica y el seguimiento clínico, al ofrecer al médico veterinario una visión longitudinal del paciente que ningún registro manual puede proporcionar de manera eficiente ni confiable. En este sentido, el sistema de información se convierte en un habilitador de calidad asistencial y no simplemente en una herramienta de organización administrativa.

Finalmente, la gestión de información en salud animal no puede disociarse de consideraciones sobre confidencialidad, integridad y disponibilidad de los datos. Aunque la normativa ecuatoriana no establece estándares técnicos específicos para registros veterinarios digitales, el principio de protección de datos personales aplicable a los propietarios de las mascotas en tanto titulares de la información y las buenas prácticas en seguridad de la información establecidas por la OWASP Foundation (2021) orientan el diseño de sistemas responsables y confiables. La adopción de estas prácticas no es optativa en un entorno profesional moderno: constituye un estándar ético que respalda la confianza del cliente, protege la integridad del dato clínico y asegura que la información del paciente esté disponible únicamente para quienes están autorizados a consultarla.

2.2 Conceptos fundamentales del ámbito clínico veterinario

El presente apartado reúne y define los conceptos clínicos y operativos fundamentales que enmarcan el desarrollo del sistema de información. Su propósito es establecer un lenguaje común entre el dominio veterinario y el dominio tecnológico, de modo que cada funcionalidad del sistema responda a un concepto médico preciso. Las definiciones se organizan en cinco

grupos temáticos: actores del proceso clínico, historia clínica y proceso de consulta, medicina preventiva, parámetros fisiológicos del examen físico, y áreas de especialidad veterinaria.

2.2.1 Actores del proceso clínico

Médico veterinario. Profesional con formación universitaria en ciencias veterinarias, habilitado legalmente para el diagnóstico, tratamiento y prevención de enfermedades en animales. En Ecuador, su ejercicio profesional está regulado por Agrocalidad, entidad que emite el registro habilitante requerido para operar en establecimientos de atención animal. El médico veterinario puede desarrollar competencias en diversas especialidades clínicas: medicina general, cirugía, oncología, dermatología, odontología, entre otras (Radostits et al., 2007).

Propietario o cliente. Persona natural o jurídica responsable legal del animal de compañía y titular de la relación con el establecimiento veterinario. Es quien otorga el consentimiento informado para los procedimientos clínicos, recibe las indicaciones del profesional y es el receptor de las comunicaciones del consultorio. La American Veterinary Medical Association (AVMA) subraya que la relación veterinario-cliente-paciente es el fundamento ético de la atención veterinaria responsable (AVMA PLIT, s. f.).

Paciente. Animal que es objeto de atención médica veterinaria. A diferencia del paciente humano, el animal no puede comunicar verbalmente sus síntomas, lo que hace del examen físico sistemático y del registro de antecedentes aportados por el propietario las herramientas diagnósticas primarias. En medicina de pequeños animales, los pacientes más frecuentes son caninos (perros) y felinos (gatos), aunque la práctica clínica incluye también aves, roedores, reptiles y conejos (Ettinger & Feldman, 2017).

Especie. Categoría taxonómica que agrupa a animales con características biológicas comunes y capacidad de reproducción entre sí. En el contexto de la clínica de pequeños animales, las especies más atendidas son el canino (Canis lupus familiaris) y el felino (Felis catus). La especie determina parámetros fisiológicos de referencia, protocolos de vacunación, rangos de dosificación farmacológica y predisposiciones a ciertas patologías (Plumb, 2018).

Raza. Subpoblación de una especie con características morfológicas, de comportamiento y predisposiciones sanitarias homogéneas, resultado de selección artificial a lo largo de generaciones. Determinadas razas presentan mayor susceptibilidad a enfermedades específicas: los Bóxer y Rottweiler tienen predisposición oncológica, los Bulldogs y Persas presentan patologías braquiocefálicas, y los Golden Retriever son propensos a displasia de cadera. El conocimiento de la raza orienta el diagnóstico diferencial y la selección del plan preventivo (Ettinger & Feldman, 2017).

2.2.2 Historia clínica y proceso de consulta

Historia clínica. Documento que registra de forma cronológica y sistemática todos los datos relativos a la salud de un paciente: antecedentes, hallazgos de cada consulta, exámenes realizados, diagnósticos, tratamientos instituidos y evolución. En medicina veterinaria, la historia clínica es el instrumento central de continuidad asistencial y constituye evidencia documentada de la atención profesional brindada (Radostits et al., 2007). Su organización convencional sigue la metodología SOAP: Subjetivo, Objetivo, Análisis y Plan.

Anamnesis. Primera fase de la consulta clínica, en la que el veterinario recopila información sobre los antecedentes del paciente a través de la entrevista al propietario. Comprende el motivo de consulta, la historia de enfermedades anteriores, los esquemas de

vacunación y desparasitación previos, el régimen alimenticio, el ambiente en que vive el animal y cualquier medicación en curso. La calidad de la anamnesis determina en gran medida la orientación del diagnóstico diferencial (Radostits et al., 2007).

Catamnesis o examen físico. Evaluación sistemática del estado de salud de la paciente realizada por el veterinario mediante inspección, palpación, percusión y auscultación. Genera datos objetivos sobre los sistemas cardiovascular, respiratorio, digestivo, nervioso, musculoesquelético, tegumentario y linfático. Los hallazgos se registran junto con los signos vitales medidos durante la consulta. La catamnesis es el equivalente a la sección Objetivo de la metodología SOAP y constituye la base sobre la que se formulan los diagnósticos (Ettinger & Feldman, 2017).

Diagnóstico presuntivo. Hipótesis clínica inicial que el veterinario formula a partir de los hallazgos del examen físico y la anamnesis, antes de disponer de los resultados de exámenes complementarios. Orienta la solicitud de pruebas diagnósticas y el inicio de un tratamiento empírico cuando la condición del paciente lo requiere. Se distingue del diagnóstico definitivo en que está sujeto a revisión conforme se obtiene nueva evidencia clínica (Radostits et al., 2007).

Diagnóstico definitivo. Conclusión clínica confirmada que establece la causa de los signos presentados por el paciente, formulada con base en la integración del examen físico, la anamnesis y los resultados de exámenes complementarios. Constituye el fundamento del plan terapéutico y queda registrado permanentemente en la historia clínica del animal (Ettinger & Feldman, 2017).

Pronóstico. Estimación fundamentada sobre la evolución esperada de la condición clínica del paciente, considerando la enfermedad diagnosticada, la respuesta probable al tratamiento y las condiciones del animal. Se clasifica habitualmente como favorable, reservado, grave o desfavorable, y se comunica al propietario como parte del proceso de consentimiento informado (Radostits et al., 2007).

Prescripción veterinaria. Documento formal emitido por el médico veterinario que indica al propietario los medicamentos que debe administrar al animal en el domicilio, con especificación de la dosis, la vía de administración, la frecuencia y la duración del tratamiento. La prescripción es un acto médico privativo del profesional habilitado y constituye un instrumento de trazabilidad farmacológica que vincula el medicamento administrado con el episodio clínico que lo originó (Plumb, 2018).

Tratamiento médico veterinario. Conjunto de medidas terapéuticas instauradas por el veterinario para controlar, curar o paliar una enfermedad. Incluye la administración de medicamentos in situ durante la consulta, la realización de procedimientos cirugía, curas, nebulizaciones y el seguimiento periódico de la evolución del paciente. Los tratamientos de larga duración requieren sesiones de control sistemáticas en las que se reevalúan los parámetros clínicos del animal (Ettinger & Feldman, 2017).

Examen complementario. Prueba diagnóstica solicitada por el veterinario para obtener información adicional que no es accesible mediante el examen físico. Incluye análisis de laboratorio clínico hemograma, bioquímica sérica, uroanálisis, coprologí-a, citología, pruebas de imagen radiografía, ecografía, tomografía y otras pruebas específicas según la

condición sospechada. Sus resultados, interpretados en conjunto con el examen físico, permiten confirmar o descartar el diagnóstico presuntivo (Ettinger & Feldman, 2017).

Medicamento de uso veterinario. Sustancia o preparado de origen biológico, químico o biotecnológico destinado a la prevención, diagnóstico o tratamiento de enfermedades en animales. Se identifica por su nombre comercial y su principio activo, que es el componente farmacológicamente activo responsable del efecto terapéutico. Sus características de presentación inyectable, tableta, suspensión, gotas, crema, pipeta, spray y concentración determinan la forma de administración y el cálculo de la dosis según el peso corporal del paciente (Plumb, 2018).

Cita clínica. Reserva programada de un tiempo de atención en el consultorio para la evaluación de un paciente. Define la fecha, la hora y el motivo de la visita, que puede corresponder a una consulta de primera vez, un control periódico, una vacunación, una urgencia, un procedimiento quirúrgico o una sesión de seguimiento de tratamiento. La gestión eficiente de la agenda de citas es un indicador de calidad organizacional del consultorio y determina su capacidad de atención diaria (Pressman & Maxim, 2019).

2.2.3 Medicina preventiva

Vacunación. Procedimiento de inmunización activa mediante el cual se inocula al animal con una preparación biológica vacuna que contiene antígenos capaces de estimular una respuesta inmunitaria protectora frente a una o varias enfermedades infecciosas. Las vacunas se clasifican en esenciales recomendadas para todos los animales independientemente de su estilo de vida, como el complejo DA2PPL en caninos y el trivalente felino y no esenciales o electivas, indicadas según el riesgo individual de exposición. Los esquemas vacunales

incluyen una serie primaria y refuerzos periódicos cuya frecuencia varía según el producto utilizado (WSAVA Vaccination Guidelines Group, 2015).

Desparasitación. Tratamiento preventivo y terapéutico dirigido a eliminar parásitos del organismo del animal o de su superficie corporal. Se distingue entre desparasitación interna, orientada a helmintos gastrointestinales nematodos, cestodos y protozoarios, y desparasitación externa, dirigida a ectoparásitos como pulgas, garrapatas y ácaros. La periodicidad recomendada varía según el tipo de parásito, el estilo de vida del animal y el riesgo ambiental de exposición. Algunas zoonosis relevantes toxocarosis, equinococosis son prevenibles mediante la desparasitación regular (Bowman, 2014).

Carné de vacunación. Documento oficial que certifica los esquemas de vacunación y desparasitación aplicados a un animal, con indicación del producto utilizado, el número de lote, la fecha de aplicación y la fecha de próximo refuerzo. Es requerido por autoridades sanitarias y establecimientos de hospedaje animal para acreditar el estado inmunitario del paciente. Su pérdida implica la imposibilidad de verificar el historial preventivo sin acceso al registro del consultorio, lo que justifica su digitalización como medida de continuidad sanitaria (WSAVA Vaccination Guidelines Group, 2015).

2.2.4 Parámetros fisiológicos del examen físico

Frecuencia cardíaca. Número de contracciones del músculo cardíaco por minuto, medida por palpación del pulso femoral o por auscultación precordial. Refleja el estado del sistema cardiovascular y puede estar alterada por enfermedad cardíaca, dolor, estrés, fiebre, deshidratación o shock. Los valores normales de referencia en reposo oscilan entre 60 y 140 latidos por minuto en caninos, y entre 120 y 240 en felinos (Ettinger & Feldman, 2017).

Frecuencia respiratoria. Número de ciclos respiratorios completos inspiración y espiración por minuto, evaluado por observación del movimiento torácico o por auscultación pulmonar. La taquipnea aumento de la frecuencia puede indicar patología respiratoria, dolor, fiebre o ansiedad; la bradipnea puede asociarse a sedación o depresión del sistema nervioso central. Los valores normales en reposo son de 10 a 30 respiraciones por minuto en caninos y de 20 a 40 en felinos (Ettinger & Feldman, 2017).

Temperatura corporal. Medida de la temperatura interna del organismo, obtenida convencionalmente mediante termometría rectal. La fiebre o hipertermia temperatura superior a 39,5 °C en perros y gatos es un signo inespecífico que orienta hacia procesos infecciosos, inflamatorios o neoplásicos. La hipotermia indica shock, exposición al frío, hipoglucemia o depresión grave del sistema nervioso central. El rango fisiológico normal es de 38,0 a 39,2 °C para ambas especies (Ettinger & Feldman, 2017).

Puntuación de Condición Corporal (PCC). Escala estandarizada para evaluar el estado nutricional y la composición corporal del animal mediante observación visual y palpación de depósitos grasos y masa muscular. La escala de 9 puntos de la AAHA es la más utilizada en pequeños animales: una puntuación de 1 indica caquexia severa, 5 corresponde a condición ideal, y 9 a obesidad marcada. La PCC es un indicador sensible de la respuesta a tratamientos nutricionales y terapéuticos, y su seguimiento entre consultas tiene valor pronóstico (Laflamme, 1997).

Mucosas. Membranas que tapizan las cavidades corporales que están en contacto con el exterior oral, nasal, ocular, genital, anal cuya coloración y grado de humectación son indicadores directos del estado de perfusión, oxigenación e hidratación sistémica. Las mucosas

sanas son de color rosado y están húmedas. La palidez sugiere anemia o vasoconstricción periférica; la cianosis indica hipoxia; la ictericia apunta a disfunción hepática o hemólisis; y el color rojo intenso puede asociarse a sepsis o intoxicación (Ettinger & Feldman, 2017).

Tiempo de llenado capilar. Tiempo que tarda en recuperarse la coloración rosada de la mucosa oral tras aplicar presión digital durante dos segundos con el dedo sobre la encía. Un tiempo menor a dos segundos se considera normal; valores entre dos y tres segundos sugieren hipoperfusión leve, y superiores a tres segundos indican shock o compromiso cardiovascular grave. Es uno de los parámetros de evaluación más rápidos y confiables en situaciones de urgencia (Ettinger & Feldman, 2017).

Ganglios linfáticos. Órganos del sistema inmunitario localizados en puntos estratégicos del organismo cuello, axila, ingle, región poplítea, abdomen que filtran la linfa y coordinan la respuesta inmune. Su evaluación por palpación forma parte del examen físico general: el aumento de tamaño (linfadenopatía) puede indicar infección local, enfermedad sistémica o neoplasia, especialmente linfoma, que es uno de los tumores más frecuentes en caninos (Withrow & Vail, 2007).

Deshidratación. Estado clínico resultante del déficit de agua corporal total, expresado convencionalmente como porcentaje del peso corporal. Se evalúa mediante el signo del pliegue cutáneo, el estado de las mucosas y el tiempo de llenado capilar. Una deshidratación del 5 % produce pérdida de elasticidad cutánea; del 8 al 10 % genera signos sistémicos marcados; y superior al 12 % constituye una emergencia que requiere fluidoterapia intravenosa urgente (Ettinger & Feldman, 2017).

Reflejos clínicos. Respuestas neurológicas involuntarias evaluadas durante el examen físico para determinar la integridad funcional del sistema nervioso central y periférico. El reflejo tusígeno valora la sensibilidad traqueal y la capacidad de toser; el reflejo palpebral evalúa el nervio facial y el trigémino mediante el cierre del párpado ante estímulo del canto medial; el reflejo podal o de retirada evalúa la sensibilidad nociceptiva del miembro ante un estímulo doloroso; y el reflejo rotuliano evalúa el nervio femoral y el segmento medular L4-L6 mediante la percusión del ligamento patelar. Su ausencia o exacerbación localiza lesiones en el sistema nervioso (de Lahunta & Glass, 2009).

2.2.5 Áreas de especialidad clínica

Oncología veterinaria. Rama de la medicina veterinaria dedicada al estudio, diagnóstico y tratamiento de neoplasias en animales. Un tumor o neoplasia es una proliferación celular anormal que puede ser benigna crecimiento local sin invasión o maligna con capacidad de invasión tisular y metástasis a distancia. La evaluación oncológica incluye la caracterización del tumor por su tamaño, consistencia, movilidad, adherencia y presencia de ulceración, parámetros que orientan la estadificación y el plan terapéutico (Withrow & Vail, 2007).

Traumatología veterinaria. Especialidad clínica que atiende las lesiones del aparato locomotor y los traumatismos físicos producidos por causas externas: atropellos, caídas, peleas o mordeduras. Comprende el manejo de fracturas, luxaciones, esguinces, contusiones y heridas. Su evaluación incluye la valoración radiológica de la lesión ósea o articular y la determinación de la gravedad leve, moderada, severa o crítica para definir el tratamiento adecuado: inmovilización, cirugía ortopédica, analgesia o manejo conservador (Piermattei et al., 2006).

Odontología veterinaria. Especialidad dedicada al diagnóstico y tratamiento de las enfermedades del aparato estomatognático en animales. La enfermedad periodontal que engloba la acumulación de sarro o cálculo dental, la gingivitis y la periodontitis es la patología más prevalente en perros y gatos adultos, afectando al 80 % de los animales mayores de tres años. La halitosis, definida como el mal olor del aliento de origen oral, es uno de los signos más frecuentes que motiva la consulta por problemas dentales (Holmstrom et al., 2013).

Examen mamario. Evaluación clínica de las glándulas mamarias mediante inspección y palpación, indicada especialmente en hembras enteras o con esterilización tardía, dado que los tumores mamarios representan el tipo de neoplasia más frecuente en perras y el tercero en gatas. La exploración sistemática de cada glándula valora la presencia de nódulos, su tamaño, consistencia, adherencia a planos profundos, existencia de ulceración y tipo de secreción, parámetros que permiten estimar la probabilidad de malignidad y orientar la decisión quirúrgica (Withrow & Vail, 2007).

2.3 Fundamentos de desarrollo de software

El desarrollo de un sistema de información moderno requiere la comprensión de un conjunto de conceptos y patrones que determinan cómo se estructura, organiza y despliega el software. Este apartado establece las bases teóricas sobre las que se sustenta el sistema desarrollado, abarcando la metodología de gestión del proyecto, la arquitectura del software, los roles de las capas de presentación y lógica, los mecanismos de persistencia de datos y los conceptos de virtualización y comunicación en tiempo real.

2.3.1 Metodología de desarrollo ágil: Scrum

Scrum es un marco de trabajo ágil para el desarrollo iterativo e incremental de productos de software complejos. Propuesto formalmente por Schwaber y Sutherland (2020), define un conjunto de roles, eventos y artefactos que organizan el trabajo del equipo en ciclos de duración fija denominados sprints, al final de cada uno de los cuales se obtiene un incremento funcional del producto. Su enfoque prioriza la adaptación continua a los requerimientos del usuario, la transparencia del proceso y la entrega de valor de manera progresiva, características que lo hacen especialmente adecuado para proyectos en los que los requisitos evolucionan durante el desarrollo (Pressman & Maxim, 2019).

El ciclo de vida de Scrum se estructura en cinco fases que se repiten de forma iterativa a lo largo del proyecto. La primera es la planificación del producto, en la que se construye y prioriza el product backlog, lista ordenada de todas las funcionalidades y requerimientos que el sistema debe cumplir. La segunda es la planificación del sprint, reunión en la que el equipo selecciona los elementos del backlog que abordará durante el siguiente ciclo y define el sprint backlog con las tareas concretas a ejecutar. La tercera es la ejecución del sprint, período de trabajo de duración fija, habitualmente de una a cuatro semanas, durante el cual el equipo desarrolla las funcionalidades comprometidas; en esta fase se realizan las reuniones diarias de sincronización, denominadas daily scrum, para identificar avances e impedimentos. La cuarta es la revisión del sprint, donde el equipo presenta el incremento funcional obtenido a los interesados del proyecto para recibir retroalimentación directa sobre el producto. La quinta y última es la retrospectiva del sprint, instancia de reflexión interna del equipo orientada a identificar oportunidades de mejora en el proceso de trabajo antes de iniciar el siguiente ciclo (Schwaber & Sutherland, 2020).

2.3.2 Metodologías ágiles en el desarrollo de software

Las metodologías ágiles en el desarrollo de software constituyen un conjunto de enfoques orientados a la construcción progresiva de sistemas mediante ciclos cortos de trabajo, colaboración continua con el usuario y adaptación constante a los cambios. A diferencia de los modelos tradicionales, que suelen organizar el proyecto en fases rígidas y secuenciales, las metodologías ágiles permiten que el producto evolucione de manera incremental, incorporando retroalimentación durante todo el proceso de desarrollo. Este enfoque resulta especialmente útil en proyectos de software donde los requerimientos pueden modificarse mientras el usuario comprende mejor sus necesidades y observa avances funcionales del sistema (Pressman & Maxim, 2019).

El desarrollo ágil se fundamenta en la entrega temprana y continua de valor, la comunicación frecuente entre los participantes del proyecto, la validación constante del producto y la capacidad de respuesta ante cambios. En este sentido, el software no se puede decir que es un producto que se entrega únicamente al final del proyecto, sino como un conjunto de incrementos funcionales que se construyen, prueban y ajustan de forma progresiva. Metodologías como Scrum, Kanban y Extreme Programming comparten estos principios, aunque cada una propone mecanismos distintos para organizar el trabajo, controlar el avance y asegurar la calidad del producto (Sommerville, 2016).

De forma general, las metodologías ágiles se desarrollan mediante iteraciones o ciclos de trabajo repetitivos. La primera fase consiste en la identificación y priorización de requerimientos, donde se define qué funcionalidades aportan mayor valor al usuario. La segunda corresponde a la planificación de la iteración, en la que se seleccionan las tareas que serán desarrolladas en un ciclo corto. La tercera fase es la construcción del incremento, donde

se diseña, programa e integra una parte funcional del sistema. La cuarta es la validación, en la que se realizan pruebas y revisiones con los interesados para comprobar si el incremento cumple con lo esperado. Finalmente, la quinta fase corresponde a la retroalimentación y mejora, donde se ajustan los requerimientos, se corrigen errores y se prepara la siguiente iteración. Este ciclo se repite hasta alcanzar un producto funcional acorde con los objetivos del proyecto.

2.3.3 Ciclo de vida del software

El ciclo de vida del software se refiere al conjunto de etapas que estructuran el proceso de creación, implementación, validación y mantenimiento de un sistema informático. Su propósito es organizar el desarrollo de software de manera ordenada, permitiendo transformar una necesidad o problema inicial en una solución tecnológica funcional. Este ciclo proporciona una guía para comprender qué actividades deben realizarse desde la identificación de requerimientos hasta la entrega y evolución del sistema (Sommerville, 2016).

En términos generales, el ciclo de vida del software está compuesto por varias fases. La primera es el análisis de requerimientos, donde se identifican las necesidades del usuario, las funciones que debe cumplir el sistema y las restricciones técnicas o de negocio que deben considerarse. La segunda fase es el diseño del sistema, en la que se define la arquitectura, los módulos, la estructura de datos, las interfaces y la forma en que los componentes interactuarán entre sí. La tercera fase corresponde a la implementación, donde los diseños se convierten en código fuente mediante el uso de lenguajes de programación, frameworks y herramientas tecnológicas. La cuarta fase es la prueba o validación, cuyo objetivo es verificar que el sistema funcione correctamente, cumpla con los requerimientos definidos y no presente errores críticos. Finalmente, la quinta fase es el despliegue y mantenimiento, en la que el sistema se

pone en funcionamiento, se corrigen fallos, se realizan mejoras y se adapta a nuevas necesidades del usuario (Pressman & Maxim, 2019).

Aunque estas fases pueden presentarse de forma secuencial en modelos tradicionales como el modelo en cascada, en los enfoques modernos de desarrollo suelen ejecutarse de manera iterativa e incremental. Esto significa que el análisis, diseño, implementación y pruebas no ocurren una sola vez, sino que se repiten en cada ciclo de desarrollo para construir progresivamente el sistema. En proyectos orientados a prototipos, este enfoque resulta conveniente porque permite validar partes del software conforme se desarrollan, realizar ajustes tempranos y reducir el riesgo de construir una solución que no responda a las necesidades reales del usuario.

2.3.4 Frontend

El frontend es la capa del sistema con la que el usuario interactúa directamente. Se ejecuta en el navegador web del dispositivo del usuario y es responsable de presentar la interfaz gráfica, gestionar la interacción del usuario y comunicarse con el backend a través de servicios expuestos para obtener o enviar datos. Un frontend bien diseñado traduce la lógica del negocio en una experiencia de uso clara e intuitiva, sin ejecutar por sí mismo operaciones sensibles sobre los datos (Garrett, 2011). Su construcción suele apoyarse en bibliotecas de componentes reutilizables y frameworks de estilos que agilizan el desarrollo y garantizan la consistencia visual de la interfaz.

2.3.5 Backend

El backend es la capa del sistema que ejecuta la lógica de negocio, gestiona el acceso a los datos y expone los servicios que el frontend consume. Es invisible para el usuario final,

pero constituye el núcleo funcional del sistema: valida las reglas del negocio, controla los permisos de acceso, procesa las operaciones y garantiza la integridad de la información almacenada. Al estar separado del frontend mediante una interfaz de comunicación definida, puede ser consumido por distintos tipos de clientes sin modificación alguna (Fowler, 2002).

2.3.6 Framework de desarrollo

Un framework es un marco de trabajo es un conjunto predefinido de herramientas, bibliotecas y buenas prácticas que establece una estructura estándar para el desarrollo de software. A diferencia de una biblioteca simple, el framework dicta la arquitectura de la aplicación y proporciona funcionalidades genéricas como son el manejo de diversas rutas, la seguridad o el acceso a bases de datos que pueden ser extendidas o modificadas según las necesidades del proyecto. Su propósito fundamental es aumentar la eficiencia del desarrollador al evitar la programación desde cero de componentes comunes, permitiendo que el esfuerzo se centre en la lógica de negocio específica (Gamma et al., 1994). Al imponer un patrón de diseño consistente, facilita además el mantenimiento del código, la colaboración en equipos de trabajo y sobre todo facilita el desarrollo.

2.3.7 Base de datos relacional, procedimientos almacenados, triggers, usuarios y roles

Una base de datos relacional es un sistema de almacenamiento de información que organiza los datos en tablas con filas y columnas, relacionadas entre sí mediante claves primarias y foráneas. El modelo relacional, introducido por Codd (1970), garantiza la integridad referencial de los datos y permite realizar consultas complejas mediante el lenguaje estándar SQL.

Un procedimiento almacenado, conocido también como stored procedure o SP, es un bloque de código SQL compilado, nombrado y guardado directamente en el servidor de base de datos. Puede recibir parámetros de entrada, ejecutar una o varias operaciones sobre los datos y devolver resultados a la aplicación que lo invoca. Su principal ventaja radica en que centraliza la lógica de acceso a datos en la base de datos y no en el código de la aplicación, reduciendo el tráfico de red, mejorando el rendimiento al estar precompilado y facilitando el mantenimiento (Fowler, 2002).

Un trigger es un procedimiento especial almacenado en la base de datos que se ejecuta de forma automática como respuesta a un evento concreto sobre una tabla: la inserción, actualización o eliminación de un registro. A diferencia de un procedimiento almacenado, el trigger no es invocado explícitamente por la aplicación, sino que es disparado por el propio motor de base de datos cuando ocurre el evento que lo activa, convirtiéndolo en un mecanismo eficaz para garantizar la coherencia de los datos de forma automática e independiente de la lógica de la aplicación (Codd, 1970).

En el contexto de la seguridad de una base de datos relacional, un usuario es una identidad reconocida por el motor de base de datos a la que se le conceden permisos para conectarse al sistema y ejecutar operaciones sobre los objetos que lo componen, como tablas, vistas o procedimientos almacenados. Un rol es un conjunto de permisos agrupados bajo un nombre que puede ser asignado a uno o varios usuarios. En lugar de definir individualmente los privilegios de cada usuario, el administrador crea roles que representan niveles de acceso concretos y los asigna según las responsabilidades de cada actor, simplificando la gestión de

permisos y garantizando que modificar un rol actualice automáticamente los privilegios de todos los usuarios que lo tienen asignado (Silberschatz et al., 2019).

2.3.8 Replicación de base de datos

La replicación de base de datos es el proceso mediante el cual los datos de un servidor de base de datos se copian y sincronizan automáticamente hacia otro servidor denominado réplica, con el propósito de mantener dos o más nodos con información consistente y permitir que cada uno opere de forma autónoma (Silberschatz et al., 2019).

Existen dos enfoques principales. La replicación física copia el estado completo del servidor de origen bloque a bloque, produciendo una copia exacta. La replicación lógica selecciona un subconjunto de tablas o datos específicos para replicar, lo que la hace más flexible cuando dos sistemas necesitan compartir solo una parte de la información sin replicar el estado completo de la base de datos (PostgreSQL Global Development Group, s. f.).

En el sistema desarrollado, la replicación lógica cumple una función arquitectónica específica: el backend del sistema de gestión mantiene la base de datos clínica principal, mientras que el backend de la landing page opera sobre una base de datos reducida conformada únicamente por las tablas que necesita disponibilidad de agenda, información del consultorio y registros de propietarios y mascotas ingresados desde la landing. La replicación garantiza que estas tablas se mantengan sincronizadas y que los datos nuevos generados en la landing page se incorporen a la base de datos clínica principal.

2.4 Herramientas tecnológicas

Las herramientas tecnológicas son los instrumentos de software concretos seleccionados para implementar los fundamentos descritos en el apartado anterior. Su elección

responde a criterios de madurez tecnológica, compatibilidad entre componentes, disponibilidad de documentación oficial y adopción en la industria del desarrollo de software. A continuación, se presentan las herramientas utilizadas en el desarrollo del sistema, organizadas según la capa del sistema a la que pertenecen.

2.4.1 Herramientas de desarrollo frontend

React es una biblioteca de JavaScript de código abierto creada y mantenida por Meta, orientada a la construcción de interfaces de usuario mediante componentes reutilizables e independientes entre sí. Su modelo de programación declarativo y su sistema de actualización eficiente del DOM mediante un árbol virtual la posicionan como una de las opciones más adoptadas en la industria para el desarrollo de aplicaciones web de mediana y gran escala (React, s. f.).

TypeScript es un superconjunto de JavaScript desarrollado por Microsoft que incorpora tipado estático opcional al lenguaje. Al añadir tipos, TypeScript permite detectar errores durante el desarrollo en lugar de en tiempo de ejecución, mejora el autocompletado y la navegación del código en los entornos de desarrollo, y hace que el código sea más legible y mantenible a medida que el proyecto crece (Microsoft, s. f.). Su uso en conjunto con React es una práctica extendida en proyectos de desarrollo profesional.

Tailwind CSS es un framework de estilos utilitario que proporciona un conjunto extenso de clases predefinidas que se aplican directamente en el marcado del componente para definir su apariencia visual. A diferencia de los frameworks de componentes tradicionales, Tailwind no impone un diseño predeterminado, sino que entrega las herramientas para construir cualquier diseño personalizado sin necesidad de escribir hojas de estilo adicionales,

lo que agiliza el desarrollo de interfaces y mantiene el código de estilos coubicado con el de la interfaz (Tailwind Labs, s. f.).

2.4.2 Herramientas de desarrollo backend

C# es un lenguaje de programación orientado a objetos, fuertemente tipado y compilado, desarrollado por Microsoft como parte de la plataforma .NET. Ofrece características modernas de productividad como inferencia de tipos, programación asíncrona nativa, expresiones lambda y LINQ, que permiten escribir código conciso y de alto rendimiento. Su ecosistema maduro y su integración nativa con las herramientas de Microsoft lo convierten en una elección sólida para el desarrollo de sistemas de información empresariales (Microsoft, s. f.).

ASP.NET es el framework de Microsoft para el desarrollo de aplicaciones y servicios web sobre la plataforma .NET. Proporciona un conjunto de herramientas para construir APIs REST, gestionar solicitudes HTTP, implementar autenticación y autorización, y conectarse con bases de datos relacionales. Su modelo de inyección de dependencias integrado y su arquitectura basada en middleware facilitan la construcción de backends modulares y testables (Microsoft, s. f.).

Linq2DB es una biblioteca de acceso a datos para C# que permite construir consultas SQL de forma tipada mediante expresiones LINQ integradas en el lenguaje. Su enfoque se distingue de los ORM tradicionales en que ofrece un control más cercano al SQL generado, siendo especialmente adecuado para proyectos que hacen uso intensivo de procedimientos almacenados y consultas personalizadas, sin sacrificar la seguridad de tipos que provee C#.

2.4.3 Motor de base de datos

PostgreSQL es un sistema gestor de bases de datos relacionales de código abierto con más de tres décadas de desarrollo activo. Ofrece soporte completo para transacciones ACID, control de concurrencia multiversión, procedimientos almacenados, triggers, roles y usuarios, y una amplia gama de tipos de datos nativos. Su estabilidad, rendimiento en cargas de trabajo concurrentes y extensibilidad lo posicionan como uno de los motores de bases de datos más utilizados en sistemas de información de nivel productivo a nivel mundial (PostgreSQL Global Development Group, s. f.).

3 CAPÍTULO 3: Metodología

El presente capítulo describe la metodología que se seguirá para llevar a cabo el proyecto, tanto desde la perspectiva de la investigación como del desarrollo del prototipo. Se define el enfoque y tipo de investigación, las técnicas para recolectar información en campo, la metodología de desarrollo adoptada y cómo se articula con el ciclo de vida del software, las fases que estructurarán el trabajo, y la justificación del uso de cada herramienta tecnológica seleccionada.

3.1 Metodología de investigación

La metodología de la investigación define la estrategia general para abordar el problema de conocimiento identificado y cumplir los objetivos planteados. Establece cómo se obtendrá la información necesaria para comprender el dominio del problema antes de construir una solución tecnológica.

3.1.1 Enfoque de la investigación

La presente investigación tiene un enfoque cualitativo, debido a que se orienta al análisis de los procesos operativos y clínicos de un consultorio veterinario con el fin de identificar sus necesidades de gestión de información. La información que sustentará el proyecto no se expresará en datos numéricos sino en la comprensión de cómo funcionan los procesos, qué problemas generan y qué estructura tienen los documentos físicos actualmente en uso (Hernández et al., 2014).

Corresponde a una investigación aplicada, ya que su propósito es resolver un problema práctico concreto: la ausencia de un sistema de información que digitalice la gestión clínica y operativa de un consultorio veterinario de pequeña escala. El nivel es descriptivo, porque se

buscará caracterizar con precisión cómo funciona actualmente el consultorio antes de proponer una solución. El diseño es no experimental, dado que no se manipularán variables, sino que se observará y analizará el funcionamiento real de los procesos para definir a partir de ese análisis los requerimientos del sistema (Hernández et al., 2014).

3.1.2 Técnicas e instrumentos de recolección de información

Para obtener la información necesaria sobre el dominio del problema se aplicarán tres técnicas complementarias durante las visitas presenciales al consultorio veterinario: entrevista semiestructurada, observación directa y análisis documental. Su combinación permitirá comprender el problema desde distintos ángulos y garantizará que los requerimientos del sistema respondan a la realidad operativa y no a suposiciones del equipo de desarrollo.

a) Entrevista semiestructurada

Las entrevistas se aplicarán al médico veterinario. Se utilizará el formato semiestructurado, que combina preguntas predefinidas con la libertad de explorar en profundidad los temas que emerjan durante la conversación. Las preguntas serán predominantemente abiertas, ya que este tipo de pregunta invita al entrevistado a describir con sus propias palabras cómo realiza su trabajo y qué problemas enfrenta, sin condicionar la respuesta a opciones predeterminadas. Se incluirán también algunas preguntas cerradas para confirmar datos puntuales como la frecuencia de consultas diarias o el número de personas que trabajan en el consultorio (Hernández et al., 2014).

Las preguntas se organizarán en cinco bloques temáticos. El primero abordará la rutina diaria: cual es el flujo de trabajo, cuántas consultas se atienden en promedio y cuáles son los momentos que más tiempo les llega a consumir. Este bloque permitirá de una manera más

rápida y eficiente. El segundo bloque abordará la gestión de citas: cómo se reciben y registran las solicitudes, qué ocurre cuando llegan fuera del horario de atención y con qué frecuencia se producen conflictos de horario o entre citas previas. El tercero cubrirá el registro clínico: cómo se crea y archiva una ficha, qué tan frecuente es no encontrar un registro cuando se necesita y cómo se maneja el seguimiento de tratamientos prolongados. El cuarto bloque abordará la medicina preventiva: cómo se controlan los esquemas de vacunación y si existe algún mecanismo para detectar pacientes con refuerzos vencidos. El quinto recogerá los problemas más frecuentes y las funcionalidades que el profesional consideraría indispensables en un sistema digital.

La información obtenida mediante estas entrevistas servirá para definir los requerimientos funcionales del sistema, identificar qué procesos deben automatizarse con mayor urgencia y conocer las expectativas del usuario sobre la solución propuesta.

b) Observación directa

La observación directa se realizará durante jornadas reales de atención mediante una modalidad no participante: el investigador estará presente en el consultorio registrando lo que ocurre sin intervenir en los procesos ni influir en la forma en que el personal trabaja. Este enfoque permite capturar situaciones que el entrevistado no menciona porque las percibe como parte natural de su rutina, incluyendo problemas que no identifica como tales (Sommerville, 2016).

Durante las observaciones se prestará atención a siete aspectos específicos: el flujo completo de atención al paciente desde que el propietario llega hasta que se retira, también el cómo se registra una mascota nueva y cuánto tiempo demanda el proceso por otro lado

importante seria el cómo se agenda una cita y cómo se gestiona ante imprevistos como cancelaciones o cómo el veterinario completa la ficha clínica durante la consulta y qué secciones le resultan más difíciles de completar por ejemplo el cómo se registra y controla la vacunación y desparasitación de cada paciente o también cómo se recupera el historial de un paciente en consultas de seguimiento viendo también el cómo y cuándo el veterinario recurre a los formularios especializados de catamnesis oncológica y a la hoja de ubicación anatómica.

La información obtenida mediante la observación permitirá validar lo que el personal describe en las entrevistas, identificar inconsistencias entre el proceso declarado y el proceso real y ayudando a detectar necesidades implícitas que el usuario se le haya olvidado o se le paso por alto al momento de contestar las preguntas.

c) Análisis documental

El análisis documental consistirá en revisar y estudiar de forma sistemática los formularios físicos que el consultorio utiliza actualmente para registrar su información clínica y operativa. El objetivo principal es comprender la estructura, el contenido y la lógica de organización de esos documentos, de modo que el sistema digital pueda replicar esa estructura en lugar de imponer una nueva que el personal deba aprender desde cero (Hernández et al., 2014).

Para cada documento se analizarán tres aspectos: qué información recoge y cómo está organizada, qué relaciones existen entre los datos que contiene, y qué limitaciones presenta en su formato físico, es decir, qué información no puede capturar, qué datos se pierden o resultan difíciles de recuperar. Los documentos para analizar serán varios documentos que el veterinario pueda llegar a darnos o mostrarnos al momento de realizar la visita, en caso de

poseer un sistema ya implementado se contempla la posibilidad de solicitar tomar fotos para poder realizar un análisis que permita entender el funcionamiento y contenido de dicho sistema.

La información obtenida mediante este análisis alimentará directamente el diseño del modelo de datos del sistema y la estructura de los formularios digitales, asegurando que cada campo del sistema corresponda a un dato real que el consultorio ya registra y que ninguna información relevante quede fuera del prototipo.

3.2 Metodología de desarrollo del prototipo

La metodología de desarrollo define cómo se construirá el sistema, qué framework guiará la gestión del proceso, cómo se articula con el ciclo de vida del software y qué herramientas tecnológicas se utilizarán en cada etapa. A diferencia de la metodología de la investigación, que responde a cómo se obtiene el conocimiento, esta parte responde a cómo se construye la solución.

3.2.1 Marco de Trabajo

Para el desarrollo del prototipo se aplicará Scrum como marco de trabajo ágil, adaptado al contexto de un proyecto de titulación desarrollado por un equipo unipersonal. Como se describió en el capítulo 2, Scrum organiza el trabajo en ciclos iterativos denominados sprints, los cuales tienen una duración fija y permiten obtener un incremento funcional del sistema al finalizar cada ciclo. En este proyecto, los esprints tendrán una duración de una semana, con fecha de inicio el 20 de marzo de 2026 y fecha estimada de cierre el 15 de junio de 2026.

El equipo de trabajo quedará conformado de la siguiente manera: el director de tesis asumirá el rol de Product Owner, siendo responsable de validar el product backlog y aprobar los incrementos desarrollados al cierre de cada sprint. El médico veterinario del consultorio participará como usuario clave, aportando retroalimentación clínica sobre los módulos relacionados con formularios médicos, atención de pacientes e información veterinaria. El autor del presente trabajo asumirá los roles de Scrum Master y desarrollador, encargándose de la planificación técnica, organización de los sprints, implementación de funcionalidades, seguimiento de tareas y control del avance del prototipo.

La gestión del backlog y el seguimiento de las actividades se realizarán mediante GitHub Projects, vinculando cada tarea con los commits correspondientes del repositorio de código fuente. Al inicio del proyecto se construirá el product backlog a partir de los requerimientos identificados durante el trabajo de campo. Los ítems serán priorizados en conjunto con el Product Owner, considerando tanto las dependencias técnicas entre módulos como el valor funcional que cada uno aporta al usuario final.

3.2.2 Integración del ciclo de vida del software con Scrum

El marco de trabajo Scrum se fundamenta de manera inherente en el modelo de ciclo de vida de prototipado evolutivo. Este enfoque permite que el sistema se construya y refine de forma incremental a través de versiones operativas (o prototipos funcionales) que evolucionan continuamente en cada iteración. En este proyecto, la combinación de ambos modelos implica que cada sprint no constituye una etapa aislada, sino un ciclo dinámico de desarrollo donde el software progresa de manera iterativa, entregando incrementos de valor sobre un conjunto específico de funcionalidades seleccionadas (Pressman & Maxim, 2019).

A pesar de la naturaleza iterativa del marco ágil, para el desarrollo de este proyecto se tomó la decisión estratégica de completar de forma exhaustiva la fase de análisis de requerimientos de manera global antes de iniciar con la etapa de construcción y desarrollo. Esta postura metodológica se justifica bajo dos premisas fundamentales:

Durante el transcurso del sprint, las fases de diseño detallado e implementación se ejecutan de manera integrada. Esto significa que cada funcionalidad seleccionada es modelada técnicamente y programada dentro de la misma iteración. De este modo, se asegura que cada incremento cumpla con los criterios de calidad establecidos, aportando de forma controlada y predecible a la evolución del prototipo final del sistema.

Para la optimización de tiempo y recursos durante el transcurso del sprint, las fases de diseño detallado e implementación se ejecutan de manera integrada. Esto significa que cada funcionalidad seleccionada es modelada técnicamente (definiendo su estructura de datos particular, la lógica del backend, los servicios necesarios y la interfaz del frontend) y programada dentro de la misma iteración. De este modo, se asegura que cada incremento cumpla con los criterios de calidad establecidos, aportando de forma controlada y predecible a la evolución del prototipo final del sistema.

Bajo este esquema, una vez consolidado el *product backlog* inicial derivado del análisis global, el proceso metodológico se traslada a la ejecución de los sprints. Cada iteración inicia con la reunión de planificación y creación del *sprint planning*, en la cual se seleccionan los requerimientos prioritarios que conformarán el *sprint backlog*.

Durante el transcurso del sprint, las fases de diseño detallado e implementación se ejecutan de manera integrada. Esto significa que cada funcionalidad seleccionada es modelada

técnicamente (definiendo su estructura de datos particular, la lógica del backend, los servicios necesarios y la interfaz del frontend) y programada dentro de la misma iteración. De este modo, se asegura que cada incremento cumpla con los criterios de calidad establecidos, aportando de forma controlada y predecible a la evolución del prototipo final del sistema.

Debido a que el proyecto será desarrollado por un equipo unipersonal, las reuniones diarias de Scrum serán adaptadas como revisiones personales de avance. En estas revisiones se registrará el estado de las tareas, los problemas encontrados y las actividades pendientes dentro de GitHub Projects. De esta manera, se mantendrá un control continuo del avance del sprint, aun cuando no exista un equipo de desarrollo conformado por varios integrantes.

Al cierre de cada sprint se realizará la revisión del incremento desarrollado. En esta actividad, el autor verificará técnicamente el funcionamiento de la funcionalidad implementada, mientras que el Product Owner validará si el resultado cumple con lo esperado. Cuando el incremento esté relacionado con formularios médicos o procesos clínicos, también se considerará la retroalimentación del médico veterinario como usuario clave. Finalmente, se realizará una retrospectiva breve para identificar dificultades, ajustes técnicos o mejoras en la planificación del siguiente sprint.

Esta integración es especialmente valiosa para los módulos clínicos del sistema, ya que dependen de la correcta interpretación de los procesos reales del consultorio veterinario. Al desarrollarlos en sprints separados y revisarlos al final de cada iteración, será posible corregir la estructura de los formularios digitales antes de avanzar al siguiente módulo, evitando que un error de diseño clínico se propague a otras partes del sistema. En un modelo secuencial, este

tipo de corrección tendría un costo mayor al descubrirse en etapas avanzadas del desarrollo (Sommerville, 2016).

3.2.3 Levantamiento de requerimientos

El levantamiento de requerimientos es el proceso mediante el cual se identificarán las necesidades, restricciones y expectativas de los usuarios del sistema antes de su diseño y desarrollo (Sommerville, 2016). A partir de la información recolectada mediante las tres técnicas descritas en la sección anteriormente, se definirán los requerimientos funcionales y no funcionales que conformarán el product backlog inicial.

Los requerimientos funcionales describirán qué debe hacer el sistema y como debe de funcionar permitiéndonos crear un modelo de procesos y flujo de trabajo con el cual realizamos el esquema de la base de datos y la forma de mostrar la información en las distintas pantallas, también identificando si algún proceso no está correctamente o esta poco entendido para poder generar otra reunión con los consultorios veterinarios con los que se haya hablado anteriormente para que nos permitan aclarar dudas sobre, procesos, información y otras dudas que puedan llegar a surgir mientras que realiza el desarrollo del prototipo del sistema.

3.2.4 Diseño del sistema

La fase de diseño permitirá traducir los requerimientos identificados y los flujos modelados en decisiones técnicas concretas para la construcción del prototipo. Esta fase se desarrollará a partir de la información obtenida durante el levantamiento de requerimientos, por lo que sus elementos serán definidos conforme se identifiquen los actores, procesos, funcionalidades y necesidades reales del consultorio veterinario.

El diseño comprenderá principalmente la elaboración de los casos de uso, los cuales permitirán representar las acciones que podrá realizar cada actor que intervenga en el funcionamiento del sistema. Posteriormente, se definirá la arquitectura lógica del prototipo, estableciendo la organización de sus componentes y la separación de responsabilidades entre la base de datos, el backend y el frontend. A partir de ello, se determinará el flujo de trabajo interno del sistema, considerando la forma en que se gestionará la información, se procesarán las solicitudes y se presentarán las interfaces visuales al usuario.

Finalmente, se diseñarán los elementos necesarios para el soporte de datos, incluyendo la estructura de la base de datos, sus relaciones y, en caso de ser requeridos, procedimientos almacenados, triggers u otros mecanismos que contribuyan al funcionamiento del sistema. De esta manera, la fase de diseño permitirá establecer una base técnica coherente con los requerimientos que sean identificados previamente.

3.2.5 Justificación de las herramientas tecnológicas

La selección de las herramientas tecnológicas para el desarrollo del prototipo responde a las necesidades técnicas identificadas para el sistema. Aunque estas herramientas ya fueron descritas en el capítulo 2, en este apartado se explica por qué fueron consideradas adecuadas para el contexto específico del proyecto.

PostgreSQL fue seleccionado como motor de base de datos gratuita debido a las características que ofrece para el manejo de información estructurada y sensible. En primer lugar, permite trabajar con procedimientos almacenados y triggers, elementos que pueden ser útiles para centralizar ciertas reglas del sistema directamente en la base de datos. Esto resulta importante en un sistema veterinario, donde parte de la información clínica debe mantenerse

organizada, validada y disponible de forma consistente. Al evaluarlo frente a otras alternativas de código abierto como MySQL, PostgreSQL destaca por su cumplimiento más estricto del estándar SQL, un soporte superior para la concurrencia de datos y una mayor capacidad para ejecutar lógicas complejas. Mientras que MySQL es excelente para transacciones simples y lecturas rápidas, PostgreSQL ofrece mayores garantías de integridad referencial y seguridad transaccional mediante el pleno cumplimiento de las propiedades ACID, lo cual es vital para evitar inconsistencias cuando se realicen operaciones simultáneas por parte de distintos usuarios (PostgreSQL Global Development Group, s. f.; Oracle, s. f.).

C# con ASP.NET fue seleccionado para el desarrollo del backend porque permite construir aplicaciones robustas, organizadas y con un alto nivel de control sobre la lógica del sistema. Al ser un lenguaje fuertemente tipado, facilita la detección temprana de errores durante el desarrollo, lo cual es relevante en un prototipo de sistema de consultorios veterinarios. Además, ASP.NET ofrece un entorno adecuado para la creación de APIs REST, permitiendo estructurar la comunicación entre el frontend, el backend y la base de datos de manera clara. En comparación con tecnologías basadas en JavaScript para el servidor, como Node.js, ASP.NET presenta ventajas significativas en la construcción de arquitecturas empresariales gracias a su naturaleza multiproceso nativa y un rendimiento superior en el procesamiento intensivo de datos. Mientras Node.js es eficiente para operaciones de entradas y salidas asíncronas ligeras, el ecosistema de C# proporciona una estructura más estandarizada y segura a largo plazo, y su integración con Linq2DB facilita el acceso a los datos desde el código de manera directa (Microsoft, s. f.; OpenJS Foundation, s. f.).

React con TypeScript y Tailwind CSS fue seleccionado para el desarrollo del frontend porque permite construir interfaces dinámicas mediante componentes reutilizables. Esta

característica se ajusta al proyecto, ya que el sistema contempla más de una interfaz y algunos elementos visuales pueden repetirse. En contraposición a frameworks más rígidos y dogmáticos como Angular, React proporciona una curva de aprendizaje más manejable, un Virtual DOM más ágil y una mayor flexibilidad para definir la arquitectura de la aplicación. Angular impone un patrón de diseño estructurado que resulta excesivamente pesado para el alcance de este sistema. TypeScript aporta la seguridad que React necesita al incorporar tipado estático, reduciendo errores al consumir los datos de la API. Por su parte, Tailwind CSS facilita la construcción visual directamente desde los componentes, evitando el exceso de archivos de estilo que demandan otros enfoques tradicionales (React, s. f.; Google, s. f.).

Linq2DB fue seleccionado como biblioteca de acceso a datos porque permite trabajar con consultas tipadas desde C#, sin perder control sobre las operaciones que se ejecutan en la base de datos. A diferencia del ORM tradicional y predeterminado del ecosistema .NET, como Entity Framework (EF) Core, Linq2DB se posiciona como una herramienta más ligera y orientada al rendimiento. EF Core, aunque automatiza en gran medida el mapeo, tiende a abstraer demasiado el control sobre la base de datos, lo que en ocasiones genera consultas SQL ineficientes. Linq2DB ofrece una forma más directa de interactuar con el código SQL, manteniendo la sintaxis de LINQ, pero permitiendo aprovechar al máximo las funciones y procedimientos almacenados definidos en PostgreSQL. Esto resulta conveniente para el proyecto, combinando la seguridad del tipado de C# con la velocidad y control que requiere la lógica transaccional del backend (Ilin et al., s. f.; Microsoft, s. f.).

4 CAPÍTULO IV: DESARROLLO

Este capítulo presenta los resultados obtenidos durante el trabajo de campo realizado en un consultorio veterinario ubicado en la ciudad de Quito para lo cual se realizaron varias visitas presenciales, cada una con una duración aproximada de entre una hora y media y dos horas. Durante estas visitas se aplicaron las tres técnicas de recolección definidas en el capítulo anterior y, a partir de lo observado, conversado y revisado, se fueron identificando los elementos que posteriormente dieron forma a los requerimientos del sistema y orientaron las decisiones de diseño del prototipo.

4.1 Levantamiento de información

El levantamiento de información se desarrolló mediante tres técnicas complementarias aplicadas directamente en el contexto real del consultorio: la entrevista semiestructurada al médico veterinario titular, la observación directa del flujo de trabajo durante jornadas de atención y el análisis de los documentos físicos que actualmente se utilizan. Esta combinación permitió comprender el problema desde diferentes perspectivas, ya que no solo se recogió la opinión del profesional, sino que también se constató con lo que ocurre durante la atención diaria y con la forma en que la información queda registrada en los formularios existentes, estas entrevistas se realizaron a gracias al médico veterinario Ruben Ortiz.

4.1.1 Entrevista

La entrevista semiestructurada, cuya guía de preguntas se presenta en el Anexo 1, se realizó durante la primera visita de campo al médico veterinario titular del consultorio. El profesional atiende principalmente a animales domésticos, sobre todo caninos y felinos, y además administra el consultorio de manera independiente. La reunión tuvo una duración aproximada una hora a dos horas y medio, además se llevó a cabo en las mismas instalaciones

que se encuentra en el sur de quito al nivel de la plataforma gubernamental, además se resero y se habló previamente para poder realizarlo en un horario en el que no había pacientes citados. Esto permitió desarrollar la entrevista con mayor tranquilidad, conversar con más detalle sobre la rutina diaria del consultorio y profundizar en los problemas que se presentan durante la atención para lograr organizar de una mejor forma la información, además las preguntas se agruparon en cinco temas principales los cuales se describen a continuación.

a) Rutina diaria y organización

Durante la entrevista con el veterinario comentó que en una jornada normal atiende entre ocho y doce consultas, aunque aclaró que los lunes y viernes suelen concentrar una mayor demanda. Según explicó, la jornada suele iniciar con la búsqueda de la ficha del primer paciente en el archivo de carpetas y ese mismo proceso se repite cada vez que llega una nueva mascota. Desde su experiencia, lo que más tiempo consume no es necesariamente la consulta médica, sino todo el trabajo previo y posterior relacionado con la ficha: encontrarla, ubicar la página correcta y completarla manualmente mientras revisa al animal, además se mencionó que en los días de mayor carga estos retrasos se acumulan a lo largo de la mañana, especialmente cuando el historial del paciente ya tiene varias hojas y resulta más difícil revisar rápidamente la información anterior.

b) Gestión de citas

Respecto a la gestión de citas, el médico veterinario señaló que la clínica carece de un sistema formal de agendamiento actual, las solicitudes de atención se reciben a través de llamadas telefónicas o mensajes de WhatsApp, y el registro se realiza de manera manual en libretas o papeles sueltos, este proceso puede llevar a generar diversas deficiencias operativas, tales como la pérdida de trazabilidad en el orden de las reservas y la superposición de turnos.

Estos cruces de horarios ocurren debido a la falta de validación de disponibilidad en tiempo real o por alteraciones derivadas de emergencias médicas, adicionalmente las solicitudes recibidas fuera del horario laboral quedan desatendidas hasta el día siguiente, lo que frecuentemente resulta en la pérdida de clientes que deciden buscar atención en otros centros veterinarios.

c) Registro clínico

En cuanto al registro clínico, el veterinario indicó que cada paciente cuenta con una ficha física numerada, la cual se guarda en una carpeta organizada por un número especial por mascota que no se puede repetir, pero el problema aparece cuando se necesita localizar esa carpeta en el archivero o en el escritorio del veterinario, ya que la búsqueda puede tomar varios minutos si no se recuerda con exactitud dónde fue colocada por otro lado también señaló que, en más de una ocasión, el mismo animal fue registrado dos veces sin que se detectara de inmediato, y el error solo se hizo evidente cuando se encontraron ambas fichas del mismo paciente. En el caso de los tratamientos prolongados, explicó que utiliza una hoja de seguimiento adjunta a la ficha, no obstante, al no existir ningún recordatorio, a veces esa hoja no se revisa antes de la consulta y la sesión pendiente puede pasar desapercibida y se puede llegar a pasar información muy importante que sea necesaria para una consulta con la mascota.

d) Medicina preventiva

Respecto a las vacunaciones y desparasitaciones, el veterinario explicó que el seguimiento depende principalmente de tres elementos: la anotación en la ficha clínica, la indicación verbal que se entrega al propietario al finalizar la consulta y el carnet físico de vacunación por lo que, como consecuencia, muchos animales regresan al consultorio con varios meses de atraso en su esquema preventivo, porque no existe un mecanismo que avise

oportunamente al propietario o al personal, además también se mencionó que el carnet se entrega en papel y que, en varias ocasiones, los propietarios han solicitado reconstruir el historial de su mascota debido a que el documento se perdió o se deterioró, lo que demuestra que la información preventiva queda demasiado expuesta a pérdidas.

e) Problemas frecuentes y necesidades

Cuando se le preguntó qué aspectos le gustaría que resolviera un sistema informático, el veterinario señaló tres necesidades principales: buscar cualquier paciente de forma rápida, revisar el historial completo sin tener que hojear varias páginas físicas y contar con una agenda donde pueda visualizar su disponibilidad sin cruces de horario. También consideró útil que los propietarios puedan solicitar citas sin depender de una llamada directa, ya que durante la jornada no siempre puede contestar y las solicitudes terminan acumulándose. Finalmente, comentó que en algún momento intentó trabajar con hojas de cálculo, pero estas se volvieron difíciles de mantener y, con el tiempo, terminó regresando al uso de formularios en papel.

f) Visibilidad del consultorio y captación de clientes

Cuando se estaba terminando la entrevista se identificó un aspecto relacionado con la forma en que el consultorio se da a conocer a nuevos propietarios. El veterinario comentó que, actualmente, la mayoría de sus pacientes llegan por clientes antiguos o por recomendaciones de otros propietarios que ya conocen el servicio, aunque esta forma de referencia ha permitido mantener una base estable de atención amplia, también limita la posibilidad de que nuevas personas encuentren información del consultorio por su cuenta sobre todo por la ubicación del consultorio.

En este sentido, se evidenció la necesidad de contar con una presencia digital más clara, principalmente mediante una página web o portal público que funcione como una carta

de presentación del consultorio. Este espacio permitiría mostrar los servicios ofrecidos, horarios de atención, ubicación, datos de contacto e información general del veterinario. Además, serviría como un canal para orientar a nuevos propietarios y facilitar que puedan solicitar una cita sin depender únicamente de una llamada, un mensaje directo o una recomendación previa.

4.1.2 Observación directa

La observación directa se realizó durante varias visitas al consultorio en jornadas de atención real, cada una con una duración aproximada de entre una hora y media y dos horas. El registro de lo observado se encuentra según la guía incluida en el Anexo 2, se efectuó bajo una modalidad donde únicamente de observador, es decir que se permaneció presente en el lugar, pero sin intervenir en los procesos ni modificar la forma en que el personal atendía a los pacientes logrando que de esta manera observar el funcionamiento cotidiano del consultorio de una forma más natural y cercana a la realidad.

a) Flujo de atención al paciente

Durante las visitas se pudo observar el flujo completo de atención, desde el momento en que el propietario llega con su mascota hasta que se retiraba del consultorio. El proceso inicia con la búsqueda de la ficha en el archivo el tiempo requerido dependía en gran medida de si el veterinario recordaba el lugar donde se encontraba ubicada. Cuando tenía presente esa ubicación, el proceso avanzaba con mayor rapidez, sin embargo, cuando no era así, debía revisar distintos ficheros y documentos, lo que podía tomar varios minutos. Una vez iniciada la consulta, el veterinario alternaba entre la revisión del animal y el llenado de la ficha, por lo que en algunos momentos el registro escrito interrumpía el flujo natural de la evaluación

clínica. En otras ocasiones, primero examinaba a la mascota y luego completaba las observaciones y hallazgos al finalizar la revisión.

b) Registro de una mascota nueva

En una de las visitas se presenció el registro de un paciente que acudía por primera vez al consultorio. Para ello, el personal tomó una ficha en blanco, revisó cuál era el último número asignado en el archivo y escribió manualmente el siguiente. El llenado de los datos del propietario y del animal tomó aproximadamente entre cinco y diez minutos, ya que el propietario no recordaba con exactitud la edad de la mascota ni tenía certeza sobre vacunas anteriores. Debido a que no había una forma de verificar esa información en ese momento, algunos campos quedaron incompletos en la ficha.

c) Gestión de citas en la jornada

También se observó que varios propietarios llegaban sin cita previa y solicitaban atención para el mismo día. En algunos casos fue posible atenderlos de inmediato porque existía un espacio libre, sin embargo, en otros tuvieron que esperar un tiempo considerable o regresar en otro momento, además mientras el veterinario se encontraba en consulta, llegaban mensajes de WhatsApp solicitando nuevas citas, pero estos no podían ser atendidos hasta que terminara la revisión del paciente. En al menos una ocasión, una solicitud quedó pendiente hasta el día siguiente, lo que evidenció la dificultad de gestionar la agenda mientras se mantiene la atención clínica activa.

d) Llenado de la ficha durante la consulta

Durante la consulta resultó evidente cómo el veterinario debía completar la sección de catamnesis mientras examinaba al paciente. Los signos vitales eran anotados en el momento

en que se medían, mientras que las observaciones diagnósticas y el diagnóstico definitivo se registraban al final, cuando el profesional ya contaba con una visión más clara del caso. En una de las atenciones se observó que el espacio disponible para observaciones no fue suficiente, por lo que el veterinario tuvo que continuar escribiendo en el margen de la hoja. Este hecho mostró una limitación importante del formato físico, ya que el espacio de registro no siempre se adapta a la complejidad de cada consulta.

e) Control de vacunación

Durante las visitas no se identificó un proceso activo para el seguimiento de esquemas vacunales. En la práctica, el tema surgía únicamente cuando el propietario llevaba el carnet físico o preguntaba por la siguiente dosis. En uno de los casos observados, el propietario no tenía el carnet consigo y, por esa razón, el personal tuvo que buscar la ficha clínica del paciente para revisar manualmente la última vacunación registrada. El proceso tardó varios minutos y, además, se presentó dificultad para interpretar la letra de una anotación anterior, lo que mostró que el soporte físico puede generar errores o demoras al momento de recuperar información preventiva.

f) Recuperación del historial clínico

En una consulta de seguimiento, el veterinario necesitó revisar qué tratamiento había indicado en la visita anterior, pero para encontrar esa información, el personal buscó la carpeta correspondiente, aunque el proceso tomó varios minutos porque el número de ficha no estaba disponible de inmediato y la búsqueda tuvo que realizarse por el nombre del propietario. Una vez localizada la ficha, el veterinario revisó manualmente los bloques de catamnesis anteriores hasta encontrar la información que necesitaba. Esta situación evidenció que el historial clínico existe, pero su consulta depende de un proceso físico lento y poco práctico.

g) Uso de formularios especializados

Durante una de las visitas se observó el uso de la hoja de ubicación anatómica, específicamente en una consulta donde el veterinario necesitaba registrar la posición de un hallazgo en el cuerpo del paciente. La hoja permanecía adjunta a la ficha mediante un clip, pero no estaba integrada formalmente al resto del expediente. Además, el veterinario mencionó que también utiliza con cierta regularidad el formulario de catamnesis oncológica para el seguimiento de pacientes con diagnóstico de tumor, aunque durante las jornadas observadas no se presentó un caso que permitiera ver ese formulario en uso. Aun así, su existencia permitió identificar la necesidad de contemplar formularios especializados dentro del sistema.

4.1.3 Evidencia documental

Durante las visitas se recopilaron y analizaron los formularios físicos que el consultorio utiliza para registrar su información clínica y operativa. Estos documentos fueron elaborados por el propio consultorio y, con autorización del médico veterinario, se tomaron como referencia para comprender la información que actualmente se registra y la forma en que esta se organiza. Los documentos revisados se presentan en los Anexos 3, 4, 5 y 6. En cada caso se analizó qué datos recoge el formulario, cómo están distribuidos y qué limitaciones presenta al mantenerse en formato físico.

a) Ficha clínica general

La ficha clínica general es el documento principal del consultorio la cual se puede visualizar en el anexo 3, ya que reúne en un mismo formulario impreso la información del propietario, los datos del paciente, la anamnesis, los antecedentes, el estado previo de vacunación, una tabla de vacunas con las abreviaturas de los productos utilizados y varios

bloques de catamnesis para registrar consultas posteriores. Cada bloque incluye campos relacionados con el examen físico, el diagnóstico, el pronóstico y el tratamiento. A pesar de su utilidad, presenta limitaciones importantes: no permite buscar al paciente por criterios distintos al número de ficha, el espacio para observaciones es fijo y puede resultar insuficiente, y no cuenta con un campo específico para registrar la próxima fecha de refuerzo de cada vacuna.

b) Hoja de seguimiento de tratamientos.

La hoja de seguimiento de tratamientos que se ve adjunto en el Anexo 4 se utiliza como complemento de la ficha clínica y permite registrar las sesiones de control de tratamientos activos. En este formulario se anota, por cada sesión, la fecha, el peso del paciente, la temperatura, la puntuación de condición corporal, el medicamento administrado en el consultorio, el medicamento indicado para continuar en casa y el costo de la sesión. Su uso confirmó que el consultorio ya tiene incorporada la práctica de controlar la evolución del paciente durante tratamientos prolongados; sin embargo, su principal limitación es que no queda vinculada de forma directa con la catamnesis que originó el tratamiento, por lo que reconstruir el contexto clínico exige revisar nuevamente toda la ficha.

c) Formulario de catamnesis oncológica.

El formulario de la catamnesis oncológica que se puede evidenciar en el anexo 5 mantiene una estructura similar al bloque de catamnesis general, pero está orientado específicamente al seguimiento de pacientes con diagnósticos neoplásicos. Incluye campos de signos vitales, examen físico y una tabla de seguimiento de tratamientos. El hecho de que el consultorio cuente con un formulario propio para estos casos evidenció que los pacientes oncológicos requieren un control más detallado y continuo, por lo que el sistema debería

considerar un espacio específico para registrar este tipo de información sin separarla del expediente general del paciente.

d) Hoja de referencia de ubicación anatómica.

La hoja del anexo 6 hace referencia de la ubicación anatómica es un documento gráfico que el veterinario utiliza durante la consulta para marcar con mayor precisión la localización de lesiones o hallazgos clínicos en el cuerpo del animal. Incluye representaciones del cuerpo del perro desde distintos ángulos, un esquema de dentición y un diagrama de las glándulas mamarias con sus posiciones numeradas. Aunque resulta útil para describir visualmente un hallazgo, al mantenerse en papel las marcas realizadas no quedan integradas de manera permanente al expediente del paciente ni pueden recuperarse fácilmente en consultas posteriores.

El análisis conjunto de estos documentos permitió concluir que el consultorio ya maneja una estructura de información bastante definida y estandarizada. Esto fue importante para el diseño del prototipo, porque no fue necesario crear una organización de datos desde cero; por el contrario, los formularios físicos sirvieron como punto de partida para identificar tablas, campos y relaciones del modelo de datos. De esta manera, el sistema propuesto puede adaptarse mejor a la forma en que el consultorio ya trabaja, pero superando las limitaciones propias del registro manual.

4.2 Identificación de problemas

A partir del levantamiento de información descrito en la sección anterior, se identificaron con mayor claridad los problemas que afectan la operación actual del consultorio y las necesidades que el sistema deberá atender. Estos problemas no surgieron de una sola fuente, sino de la relación entre lo que el veterinario manifestó en la entrevista, lo que se

observó durante las jornadas de atención y lo que revelaron los documentos físicos sobre las limitaciones del registro actual. De igual manera, las necesidades del sistema se definieron considerando tanto las solicitudes expresadas por el profesional como los requerimientos derivados del análisis realizado.

4.2.1 Problemas detectados en el consultorio

Los problemas identificados se agrupan en cinco áreas principales: la gestión de citas, el manejo de la información clínica, el seguimiento preventivo, las limitaciones propias de los documentos físicos y la baja visibilidad digital del consultorio. Esta clasificación permite entender con mayor orden dónde se concentran las dificultades del consultorio y cómo cada una de ellas afecta tanto la atención diaria como la posibilidad de llegar a nuevos propietarios de mascotas.

a) Gestión de citas sin soporte digital

El consultorio no dispone de un sistema digital de agenda. Actualmente, las solicitudes de cita llegan por llamada telefónica o por WhatsApp y se registran en una libreta, sin una validación automática de disponibilidad por lo que como resultado, pueden producirse conflictos de horario que muchas veces se detectan cuando el propietario ya se encuentra en el consultorio, además las solicitudes enviadas fuera del horario de atención suelen quedar pendientes hasta el día siguiente y, durante ese tiempo, el propietario puede acudir a otro establecimiento a esto se le< suma que no existe un historial de citas asociado a cada paciente, por lo que conocer la última visita de un animal requiere revisar manualmente su ficha clínica.

b) Ineficiencia en el manejo de la información clínica

La búsqueda de fichas físicas constituye uno de los problemas más visibles en la operación diaria del consultorio, cuando el personal no recuerda el lugar donde fue guardada la ficha se puede llegar a perder varios minutos buscando en un fichero los documentos necesarios para comenzar la intención por lo que en una atención regular esto ya representa una demora, sin embargo, en situaciones de urgencia puede afectar directamente la rapidez de la atención, además como no existe una verificación automática al registrar nuevos pacientes, se han presentado casos de fichas duplicadas para el mismo animal. A esto se suma que el llenado manual de la catamnesis puede resultar limitado cuando el espacio del formulario no alcanza para registrar todas las observaciones clínicas necesarias.

c) Ausencia de seguimiento preventivo activo

Otro problema relevante es la ausencia de un seguimiento preventivo activo. En la actualidad no existe un mecanismo que identifique de manera anticipada qué pacientes tienen vacunas o desparasitaciones próximas a vencer o ya vencidas. Por ello, el control depende casi por completo de la memoria del propietario y de la indicación verbal que el veterinario entrega al finalizar la consulta por lo que, como consecuencia, algunos animales regresan con meses de atraso en su esquema preventivo sumando también que cuando el carnet físico se pierde o se daña, reconstruir el historial de vacunación exige revisar manualmente la ficha clínica, lo que puede generar demoras e incluso errores de lectura.

d) Fragmentación de los documentos físicos

Aunque los formularios físicos del consultorio están bien estructurados, funcionan de manera aislada entre sí por lo que la hoja de seguimiento de tratamientos no mantiene una referencia directa con la catamnesis que originó el tratamiento, el formulario de catamnesis

oncológica no se encuentra integrado al historial general del paciente y la hoja de ubicación anatómica se conserva únicamente como un documento adjunto. Debido a esto, la información clínica queda fragmentada y depende de la presencia física de las carpetas en el consultorio por lo que estos documentos no permiten búsquedas rápidas, alertas automáticas ni acceso remoto, lo que limita la disponibilidad de la información cuando se la necesita.

e) Baja visibilidad digital del consultorio

Otro problema identificado, especialmente a partir de la conversación con el veterinario, es la limitada presencia digital del consultorio ya que actualmente, el establecimiento es conocido principalmente por clientes antiguos o por recomendaciones personales, lo que significa que una persona que no ha recibido una referencia previa tiene pocas formas de conocer los servicios, horarios, ubicación o medios de contacto disponibles.

Esta situación no solo reduce las posibilidades de captar nuevos clientes, sino que también genera una carga adicional para el veterinario, ya que debe responder de manera individual preguntas frecuentes sobre servicios, disponibilidad o ubicación por lo que cuando una persona interesada no recibe respuesta inmediata por llamada o WhatsApp, puede optar por acudir a otro consultorio que sí tenga información visible y accesible en línea.

4.2.2 Necesidades del sistema

Las necesidades del sistema se derivan directamente de los problemas identificados y de las funcionalidades que el veterinario consideró indispensables durante la entrevista. Para organizarlas de mejor manera, se agrupan en cinco áreas relacionadas con las dificultades descritas anteriormente: gestión digital de citas, expediente clínico integrado, seguimiento preventivo, digitalización de formularios especializados y presencia digital del consultorio.

a) Gestión digital de citas

El sistema deberá contar con una agenda digital que permita registrar, visualizar y administrar las citas programadas por lo que se deberá validar la disponibilidad del veterinario antes de confirmar un turno, de modo que se reduzcan los cruces de horario y se mejore la organización de la jornada, también será necesario que los propietarios puedan solicitar citas sin llamar directamente al consultorio, especialmente cuando el veterinario se encuentra atendiendo o cuando la solicitud se realiza fuera del horario laboral, por lo que finalmente cada cita deberá quedar asociada al paciente correspondiente para construir un historial de atenciones consultable en cualquier momento.

b) 4.2.2.2 Expediente clínico digital e integrado

El sistema deberá centralizar en un solo expediente digital toda la información relacionada con el paciente, esto puede llegar a incluir los datos del propietario, los datos del animal, la anamnesis, las consultas realizadas, las catamnesis, las vacunas, las desparasitaciones, los tratamientos activos y los hallazgos registrados en formularios especializados , logrando tener un expediente que podrá buscar de forma rápida por el nombre de la mascota o por la cédula del propietario, sin depender del número de ficha física por lo que de igual manera, será importante incorporar mecanismos de verificación que ayuden a evitar registros duplicados antes de crear un nuevo expediente.

c) Seguimiento preventivo con trazabilidad

El sistema deberá permitir registrar, por cada evento de vacunación o desparasitación, la fecha de aplicación y la próxima fecha de refuerzo. De esta forma, el estado preventivo de cada paciente podrá visualizarse desde su expediente y no dependerá únicamente del carnet físico o de la memoria del propietario. Asimismo, el historial completo de vacunaciones y

desparasitaciones deberá mantenerse disponible de manera digital tanto para el veterinario como para el propietario, fortaleciendo la trazabilidad y reduciendo el riesgo de pérdida de información.

d) Digitalización y vinculación de los formularios especializados

Los módulos especializados del sistema deberán trasladar al entorno digital los formularios físicos que el consultorio ya utiliza, como la catamnesis oncológica, la hoja de seguimiento de tratamientos y la referencia de ubicación anatómica, sin embargo, a diferencia del formato actual que es en papel, cada registro deberá quedar vinculado al expediente del paciente y disponible dentro del historial clínico. De esta manera, el veterinario podrá revisar el contexto completo de una consulta o tratamiento anterior sin buscar documentos adicionales, lo que facilitará la continuidad de la atención y reducirá la fragmentación de la información.

e) Presencia digital y portal informativo

El sistema deberá incluir un portal público o página web que permita mostrar información general del consultorio veterinario. Este portal deberá presentar datos como el nombre del consultorio, ubicación, horarios de atención, medios de contacto, servicios disponibles, información del veterinario y anuncios relevantes para los propietarios.

Esta necesidad se relaciona con la forma en que el consultorio puede darse a conocer a nuevos clientes. Al contar con una página informativa, los propietarios podrán encontrar información básica antes de comunicarse directamente con el veterinario, lo que reduce la dependencia de recomendaciones personales y facilita el primer contacto con el consultorio.

4.3 Requerimientos del sistema

Después de identificar los problemas del consultorio y las necesidades principales del médico veterinario, se definieron los requerimientos del sistema que nos permiten establecer qué debe realizar el prototipo y qué condiciones debe cumplir para apoyar la gestión clínica y operativa del consultorio.

Para mantener una organización clara, los requerimientos se agruparon en dos tipo, en primer lugar, están los requerimientos funcionales que describen las acciones que el sistema debe permitir realizar a los usuarios, mientras que los requerimientos no funcionales establecen condiciones de calidad relacionadas con seguridad, usabilidad, integridad de la información, disponibilidad y mantenimiento del prototipo.

Esta definición sirve como base para las siguientes etapas del desarrollo, debido a que a partir de ella se podrán construir el Product Backlog, los casos de uso, el modelo de datos y la planificación de los Sprints.

4.3.1 Requerimientos funcionales

Los requerimientos funcionales describen las funciones principales que el sistema debe cumplir para responder a los problemas encontrados durante el levantamiento de información, en este caso se priorizaron las funcionalidades relacionadas con usuarios, propietarios, pacientes, citas, historial clínico, seguimiento médico y presencia digital del consultorio, el levantamiento de todos estos requerimientos se pueden evidenciar en la tabla 1 la cual nos da un detalle del código que se coloca al requerimiento funcional junto con el nombre asignado y una pequeña descripción de lo que se espera de dicha funcionalidad.

Los requerimientos funcionales que se van a resumir son las necesidades principales del prototipo sin separar cada acción básica de un requerimiento independiente ya que, de esta

manera, cada módulo concentra las operaciones necesarias para su funcionamiento, como registrar, consultar, actualizar o cambiar estados, manteniendo una estructura más sencilla y alineada con el alcance del proyecto.

Tabla 1. Requerimientos funcionales del sistema

Código	Requerimiento funcional	Descripción
RF-01	Gestionar usuarios y roles	El sistema debe permitir el acceso de usuarios internos mediante autenticación, diferenciando el acceso del usuario interno es decir el médico veterinario del acceso del cliente al portal.
RF-02	Gestionar propietarios	El sistema debe permitir registrar, consultar y actualizar los datos de los propietarios, incluyendo información de contacto y dirección.
RF-03	Gestionar pacientes	El sistema debe permitir registrar, consultar y actualizar la información de las mascotas, asociándolas correctamente con su propietario.
RF-04	Gestionar citas y agenda	El sistema debe permitir crear, consultar, confirmar, cancelar o reprogramar citas, validando la disponibilidad del veterinario.
RF-05	Consultar historial clínico	El sistema debe permitir visualizar el historial clínico completo de cada paciente, con sus consultas, tratamientos y registros preventivos.
RF-06	Registrar consulta clínica	El sistema debe permitir registrar anamnesis, catamnesis, signos clínicos, diagnóstico, pronóstico, observaciones y tratamiento indicado.
RF-07	Registrar vacunación y desparasitación	El sistema debe permitir registrar vacunas y desparasitaciones aplicadas, conservando el historial preventivo del paciente.
RF-08	Registrar tratamientos, prescripciones y exámenes	El sistema debe permitir registrar tratamientos, controles posteriores, prescripciones médicas, pedidos de exámenes y resultados asociados al paciente.
RF-09	Gestionar formularios especializados	El sistema debe permitir registrar información relacionada con formularios clínicos especializados, como oncología, odontología, traumatología o examen mamario.
RF-10	Gestionar catálogos básicos	El sistema debe permitir administrar catálogos necesarios para el registro clínico, como especies, razas, medicamentos, vacunas, servicios y tipos de examen.
RF-11	Permitir solicitud de citas desde el portal	El sistema debe permitir que los propietarios soliciten citas desde el portal, reduciendo la dependencia de llamadas o mensajes directos.
RF-12	Gestionar información del consultorio	El sistema debe permitir administrar información básica del consultorio, servicios, datos de contacto e información visible para el portal público.
RF-13	Mostrar información pública del consultorio	El sistema debe contar con una página pública que muestre información general del consultorio, como nombre, ubicación, horarios, servicios, datos de contacto e información del veterinario.
RF-14	Gestionar contenido del portal público	El sistema interno debe permitir administrar la información que se mostrará en la página web, incluyendo servicios, imágenes, anuncios o comunicados.
RF-15	Orientar a nuevos propietarios desde el portal	El portal público debe servir como un canal de orientación para propietarios que aún no conocen el consultorio, facilitando el primer contacto y la solicitud de atención.

Nota. Listado de los diferentes requerimientos funcionales que se han levantado para el prototipo del sistema de consultorios veterinarios. Elaborado por el autor Calvopiña Daniel.

4.3.2 Requerimientos no funcionales

Tabla 2. Requerimientos no funcionales del sistema

Código	Requerimiento no funcional	Descripción
RNF-01	Seguridad de acceso	El sistema debe proteger el acceso mediante autenticación y control de roles, evitando que usuarios no autorizados ingresen a información clínica o administrativa.
RNF-02	Integridad de la información	El sistema debe mantener relaciones correctas entre propietarios, pacientes, citas, consultas y tratamientos, evitando duplicidad o pérdida de información.
RNF-03	Trazabilidad clínica	El sistema debe conservar el historial de cada paciente de forma cronológica, permitiendo consultar su evolución médica, tratamientos y controles anteriores.
RNF-04	Usabilidad	Las pantallas del sistema deben ser claras y fáciles de usar, de modo que el personal pueda registrar información sin complicar el flujo normal de atención.
RNF-05	Disponibilidad de la información	La información registrada debe estar disponible para su consulta cuando el personal autorizado la necesite, especialmente durante una consulta o seguimiento clínico.
RNF-06	Validación de datos	El sistema debe validar campos obligatorios, formatos de correo, números de contacto, fechas, horarios y datos necesarios para evitar registros incompletos.
RNF-07	Mantenibilidad	El sistema debe organizarse en módulos y capas que faciliten futuras modificaciones, correcciones o ampliaciones del prototipo.
RNF-08	Compatibilidad tecnológica	El sistema debe desarrollarse con las herramientas definidas para el proyecto, considerando backend, frontend y base de datos relacional.
RNF-09	Respaldo documental	El sistema debe permitir que la información registrada sirva como soporte digital del expediente clínico, reduciendo la dependencia de documentos físicos.
RNF-10	Alcance controlado del prototipo	El sistema debe limitarse a la gestión clínica y operativa definida en el proyecto, sin incluir facturación electrónica, contabilidad, inventario o integración con hardware externo.
RNF-11	Accesibilidad de la información pública	La información publicada en la página web debe ser clara, fácil de encontrar y comprensible para propietarios que visiten el portal por primera vez.
RNF-12	Actualización de contenido público	El sistema debe permitir que la información pública del consultorio pueda actualizarse desde el sistema interno sin modificar directamente el código fuente.
RNF-13	Responsividad del portal público	La página web debe visualizarse correctamente en computadoras, tablets y teléfonos móviles, considerando que muchos propietarios buscarán información desde su celular.

Nota. Listado de los diferentes requerimientos no funcionales que se han levantado para el prototipo del sistema de consultorios veterinarios. Elaborado por el autor Calvopiña Daniel.

Los requerimientos no funcionales definen las condiciones que debe cumplir el sistema para que sus funcionalidades puedan utilizarse de manera segura, ordenada y confiable, aunque no representan módulos visibles para el usuario, son importantes porque influyen directamente en la calidad del prototipo y en la forma en que se manejará la información

clínica del consultorio, esto podemos apreciarlo en la tabla 2 la cual nos da un detalle del código que se coloca al requerimiento no funcional junto con el nombre asignado y una pequeña descripción de lo que se espera de dicha funcionalidad.

En conjunto, estos requerimientos permiten que el prototipo responda a los principales problemas detectados durante el levantamiento de información, especialmente la búsqueda manual de fichas, la fragmentación del historial clínico, la falta de seguimiento preventivo, la gestión poco organizada de citas y la baja visibilidad digital del consultorio. Además, ayudan a mantener el desarrollo dentro de los límites establecidos para el proyecto, evitando incluir módulos que no forman parte de la propuesta inicial.

4.4 Product Backlog

El Product Backlog se definió como la lista ordenada de funcionalidades, necesidades y trabajos técnicos necesarios para alcanzar el objetivo del producto. En Scrum, este artefacto no corresponde a una planificación cerrada por fechas, sino a un conjunto vivo de elementos que se ordena de acuerdo con el valor que aportan, las dependencias entre módulos y la claridad alcanzada durante el refinamiento por esta razón, en esta sección se presenta primero el Product Backlog como una lista priorizada del producto completo y, posteriormente, en la sección de planificación, se explica qué elementos fueron seleccionados para cada Sprint.

La corrección principal frente a una organización inicial del backlog consiste en separar claramente el Product Backlog del Sprint Backlog. El Product Backlog se refiere a lo que el producto necesita desde un punto de vista general, mientras que el Sprint Backlog contiene únicamente los elementos seleccionados para un Sprint específico, junto con el objetivo del Sprint y el plan de trabajo para convertirlos en un incremento funcional, por lo

que esta separación evita confundir la priorización general del sistema con la planificación concreta de cada iteración.

En el caso del prototipo veterinario, los elementos del Product Backlog se obtuvieron a partir del levantamiento de información, los problemas identificados en el consultorio, los requerimientos funcionales y no funcionales. De esta manera, no se priorizó únicamente lo que era visible en las pantallas, sino también aquello que debía existir antes para que los módulos funcionaran correctamente, como reglas de negocio, modelo de datos, catálogos, parametrización del consultorio y validaciones de base de datos.

4.4.1 Objetivo del producto

El objetivo del producto es construir un prototipo de sistema de información que centralice la gestión clínica y operativa del consultorio veterinario, para ello el prototipo debe permitir registrar propietarios y pacientes, administrar citas y agenda, mantener un historial clínico digital, controlar información preventiva como vacunaciones y desparasitaciones, gestionar tratamientos, prescripciones y exámenes, y ofrecer una página pública que ayude al consultorio a darse a conocer a nuevos propietarios.

Este objetivo permitió ordenar el Product Backlog desde los elementos más estructurales hasta los más especializados, por esa razón, la primera prioridad no fue diseñar pantallas aisladas, sino establecer las reglas de negocio y el modelado de datos que sostienen todo el sistema para acto seguido ubicar los módulos de seguridad, parametrización, catálogos, propietarios, pacientes, agenda, historia clínica y procesos médicos avanzados.

4.4.2 Criterios de priorización

La priorización del Product Backlog se realizó considerando tres criterios principales, el primero fue la dependencia funcional, ya que algunos módulos necesitan que otros existan previamente como es el caso de que no es posible registrar una consulta médica si antes no existen un paciente, un propietario, un veterinario y una cita asociada, el segundo criterio fue el impacto sobre los problemas detectados en el consultorio, especialmente la búsqueda manual de fichas, la falta de agenda digital, la pérdida de trazabilidad clínica y la baja visibilidad del consultorio frente a nuevos clientes y para finalizar el tercer criterio fue la complejidad técnica, diferenciando entre CRUDs simples, procesos compuestos y reglas avanzadas de base de datos ya que algunas partes de la lógica que se va a implementar puede llegar a ser compleje y necesita estar correctamente validada.

A partir de estos criterios, las funcionalidades se ordenaron de forma progresiva en primer lugar se ubicaron los elementos de análisis y diseño base para acto seguido después comenzar con la seguridad, parametrización y catálogos de los elementos clave luego, acto seguido la gestión de clientes, pacientes, citas y agenda para acto seguido comenzar con la creación y visualización del portal web y finalmente la parte médica general, los procesos clínicos avanzados, los procedimientos almacenados, triggers y pruebas funcionales.

4.4.3 Lista de elementos del Product Backlog

La lista que se puede visualizar en la tabla 3 se encarga de priorizar el Product Backlog y presentar un conjunto de elementos que representan funcionalidades del sistema y trabajos técnicos necesarios para construir el prototipo donde esta tabla presenta 6 apartados donde se puede evidenciar un código designado el nombre del elementro el tipo del que se trata el elemento seguido de su prioridad en el sistema para posteriormente tener un puntaje de

priorización del 1 al 10 para finalizar con una breve descripción de lo que se realizara en ese apartado.

4.4.4 Historias de usuario principales

Las historias de usuario permitieron expresar las funcionalidades desde la perspectiva de los actores del sistema. A diferencia de una descripción técnica, una historia de usuario ayuda a entender quién necesita una funcionalidad, qué desea lograr y para qué le sirve dentro del proceso del consultorio. En este proyecto se utilizaron historias de usuario para representar las necesidades del médico veterinario, el cliente y el visitante de la página pública. El detalle completo de cada historia de usuario, con su número, nombre, prioridad, descripción y observaciones, se presentan todas ellas en el Anexo 8.

4.4.5 Refinamiento del Product Backlog

El Product Backlog no se consideró un documento cerrado desde el inicio, ya que durante el desarrollo, algunos elementos se refinaron conforme se comprendían mejor las necesidades del consultorio y las dependencias entre módulos por ejemplo tenemos el caso de la gestión de citas ya que no se podía tratarse como un CRUD simple, porque requería disponibilidad, calendario, estados, bloqueos y validaciones de horario para su utilización de igual manera, la parte médica general necesitaba conectarse con la cita, el paciente y el historial clínico antes de incorporar tratamientos, prescripciones o exámenes ya que no tendríamos una base sobre lo cual trabajar, por otra parte se identificó que la página pública se tenía que tomar en cuenta en el desarrollo ya que el veterinario no solo necesitaba ordenar su operación interna, sino también contar con un canal digital que permitiera mostrar sus servicios y facilitar el contacto con nuevos propietarios, por esta razón, el Product Backlog

incluyó tanto módulos internos como funcionalidades orientadas al portal público y al portal de clientes.

Tabla 3. Listado de elementos de Product Backlog

Código	Elemento del Product Backlog	Tipo	Prioridad	Puntos	Descripción
PB-01	Reglas de negocio y modelado de datos	Técnico	Alta	8	Definir entidades, relaciones y reglas base del sistema veterinario.
PB-02	Infraestructura de seguridad y autenticación	Técnico	Alta	5	Montar el motor de autenticación con JWT y las políticas que distinguen al veterinario del cliente, sobre la estructura de tablas de usuarios.
PB-03	Parametrización del consultorio y veterinarios	Funcional	Alta	5	Configurar los datos del consultorio, datos de diversas categorías, parámetros de agenda y el registro de veterinarios y sus horarios.
PB-04	Creación de clientes y acceso al sistema	Funcional	Alta	5	Crear clientes y habilitar el acceso operativo, tanto el inicio de sesión interno como el de clientes, que comparten la misma lógica de autenticación.
PB-05	CRUDs simples y catálogos	Funcional	Alta	8	Gestionar especies, razas, tipos de examen, tipos de vacuna, medicamentos, servicios y enunciados.
PB-06	Gestión de propietarios y pacientes	Funcional	Alta	8	Registrar propietarios, mascotas y su relación para crear expedientes digitales.
PB-07	Agenda, calendario y citas	Funcional	Alta	8	Crear citas, validar disponibilidad, generar el calendario, manejar estados y bloquear horarios.
PB-08	Landing page y portal del cliente	Funcional	Media alta	5	Mostrar la información pública del consultorio y permitir la reserva de citas del cliente, reutilizando la lógica de agenda y citas del sistema.
PB-09	Historial clínico y consulta médica general	Funcional	Alta	8	Registrar anamnesis, catamnesis, diagnóstico, pronóstico, vacunación y desparasitación.
PB-10	Procesos médicos avanzados	Funcional	Media alta	8	Registrar tratamientos, seguimientos, prescripciones, pedidos y resultados de exámenes.
PB-11	Especialidades clínicas	Funcional	Media	5	Registrar oncología, odontología, traumatología y examen mamario.
PB-12	Procedimientos almacenados, triggers y reglas de integridad	Técnico	Media alta	8	Reforzar validaciones, consistencia e integridad de la base de datos.
PB-13	Pruebas funcionales y ajustes finales	Técnico	Alta	5	Validar flujos principales y corregir errores antes de la entrega.

Nota. Listado del Producto Backlog. Elaborado por el autor Calvopiña Daniel.

4.5 Planificación de sprints y Sprint Backlog

La planificación de sprints se realizó después de contar con el Product Backlog. En Scrum, durante la planificación del Sprint, el equipo selecciona elementos del Product Backlog y define un objetivo para la cada iteración, a partir de esa selección se construye el Sprint Backlog, el cual incluye el objetivo del Sprint, los elementos seleccionados y el plan de trabajo necesario para entregar una iteración funcional, por esta razón, en esta sección no se repite el Product Backlog, sino que se explica qué se tomó cada Sprint y cómo se los transformó en tareas concretas.

El período de desarrollo considerado para el prototipo fue desde el 16 de marzo hasta el 22 de junio de 2026, debido a que el trabajo fue desarrollado por un equipo unipersonal y desde el comienzo del primer semestre del año 2026, los sprints se organizaron en cuatro bloques de distinta duración según la complejidad de los módulos que abarcaba cada uno. Los dos primeros sprints se trabajaron como bloques únicos, mientras que el tercero y el cuarto se dividieron en iteraciones por módulo. Esta adaptación permitió mantener la lógica de Scrum sin perder de vista las limitaciones reales de tiempo del proyecto.

4.5.1 Diferencia aplicada entre Product Backlog y Sprint Backlog

El Product Backlog se utilizó como la lista general y priorizada de todo lo que el producto necesitaba, en cambio, el Sprint Backlog se construyó al inicio de cada Sprint con los elementos seleccionados para esa iteración, esta diferencia es importante porque un elemento puede existir en el Product Backlog sin estar todavía comprometido para un Sprint, mientras que el Sprint Backlog representa el trabajo que se planifica ejecutar en un período específico.

En la práctica del proyecto, el Product Backlog permitió ordenar las funcionalidades por valor y dependencia, mientras que el Sprint Backlog permitió dividirlas en tareas

concretas. Por ejemplo, el elemento de agenda, calendario y citas del Product Backlog se convirtió dentro del Sprint correspondiente en tareas como definir disponibilidad, crear bloques de calendario, validar cruces de horario, registrar citas, cambiar estados y probar cancelaciones o confirmaciones.

4.5.2 Desarrollo planificado por sprint

a) Sprint 1: 16 de marzo al 5 de abril de 2026 (tres semanas)

Objetivo del Sprint: Definir las reglas de negocio, los requerimientos y el modelo de datos del sistema, y dejar montada la infraestructura de seguridad y autenticación.

Elementos seleccionados del Product Backlog: PB-01, PB-02.

Este primer Sprint se trabajó como un bloque único, sin dividirse en iteraciones por módulo, porque su propósito fue establecer la base conceptual y técnica sobre la que se construirían los demás Sprints, durante estas tres semanas se consolidaron las reglas de negocio obtenidas en el levantamiento de información, se depuraron los requerimientos funcionales y no funcionales, y se elaboró el modelo de datos con las entidades principales de consultorio, veterinario, cliente, paciente, agenda, cita e historial clínico.

A partir del modelo se creó la estructura base de tablas en PostgreSQL y se montó la infraestructura de seguridad, que incluye el motor de autenticación con emisión de token JWT y las políticas que diferencian al veterinario del cliente. En este punto se dejó preparado el mecanismo para validar a un usuario, pero el inicio de sesión todavía no operaba con usuarios reales, porque la creación de veterinarios y clientes corresponde al Sprint siguiente ya que al cerrar el Sprint se verificó que el modelo de datos soportara las relaciones necesarias y que la base de autenticación quedara lista.

Plan de trabajo del Sprint Backlog:

- Consolidar las reglas de negocio del consultorio.
- Depurar los requerimientos funcionales y no funcionales.
- Elaborar el modelo entidad-relación del sistema.
- Crear la estructura base de tablas en PostgreSQL.
- Montar la infraestructura de autenticación con JWT y las políticas de acceso para veterinario y cliente.
- Verificar la integridad de las relaciones del modelo.

Incremento obtenido: Modelo de datos, reglas de negocio y motor de seguridad listos como base del prototipo.

b) Sprint 2: 6 de abril al 19 de abril de 2026 (dos semanas)

Objetivo del Sprint: Configurar el consultorio y los veterinarios, crear los clientes y habilitar el acceso operativo al sistema, y desarrollar los CRUD simples y catálogos.

Elementos seleccionados del Product Backlog: PB-03, PB-04, PB-05.

El segundo Sprint también se desarrolló como un bloque único, sin iteraciones por módulo, porque agrupa la configuración inicial y los mantenimientos base, que comparten una estructura de operaciones muy similar. Lo primero fue la parametrización del consultorio, donde se registran sus datos, los parámetros de agenda, los veterinarios con sus horarios la lista de los catálogos que se van a implementar en el sistema y la creación de veterinarios dentro del sistema, este orden es necesario porque sin veterinarios y sin clientes creados no es posible iniciar sesión, ya que la infraestructura de autenticación montada en el Sprint 1 solo se vuelve operativa cuando ya existen los usuarios a quienes autenticar.

Una vez creados los veterinarios y los clientes, se habilitó el acceso operativo al sistema. El inicio de sesión interno del veterinario y el de los clientes comparten la misma lógica de autenticación, por lo que se trabajaron juntos en este Sprint y al mismo nivel. A

continuación, se desarrollaron los CRUD simples y catálogos, que abarcan especies, razas, tipos de examen, tipos de vacuna, medicamentos, servicios y enunciados de la clínica. Estos catálogos son la información reutilizable que necesitan los procesos posteriores. Se aclara que la categoría no constituye un catálogo independiente, sino un atributo del servicio, por lo que se trabajó como un campo dentro del CRUD de servicios. Para el borrado seguro de los catálogos se incorporaron procedimientos almacenados que verifican que el registro no esté en uso antes de eliminarlo.

Plan de trabajo del Sprint Backlog:

- Parametrizar el consultorio con sus datos institucionales y los parámetros de agenda.
- Registrar veterinarios y sus horarios.
- Crear clientes y habilitar el acceso operativo (login interno y de clientes, con la misma lógica de autenticación).
- Desarrollar el CRUD de especies y razas.
- Desarrollar el CRUD de tipos de examen, tipos de vacuna y medicamentos.
- Desarrollar el CRUD de servicios (categoría como atributo) y de enunciados.
- Implementar procedimientos almacenados de eliminación segura y probar cada CRUD.

Incremento obtenido: Consultorio y veterinarios configurados, acceso al sistema operativo y catálogos base disponibles para los demás módulos.

c) Sprint 3: 20 de abril al 31 de mayo de 2026 (seis semanas)

Objetivo del Sprint: Desarrollar los CRUD particulares de propietarios y pacientes, construir el módulo de agenda y citas, y desarrollar la landing page y el portal del cliente.

Elementos seleccionados del Product Backlog: PB-06, PB-07, PB-08.

A diferencia de los dos primeros Sprints, este Sprint sí se organizó en iteraciones por módulo, porque cada módulo tiene reglas propias y depende de los anteriores. Los procedimientos almacenados y triggers de cada módulo se construyeron dentro de su propia

iteración, en el momento en que se necesitaban, y no al final del desarrollo. A continuación, se describe cada iteración con su duración aproximada.

Iteración 1. Propietarios y pacientes (dos semanas). Se desarrolló el registro y la consulta de propietarios y, a continuación, el de mascotas, asociando cada mascota a su propietario. El registro de propietarios permite listar, consultar, crear y actualizar la información del cliente. En el caso de las mascotas, además del registro y la actualización, se incorporó el cambio de estado para activar o inactivar a la mascota sin borrarla, de modo que no se pierda la información clínica asociada. En esta iteración se incorporó el trigger que genera el identificador del paciente y se validó que no se pudiera registrar una mascota sin un propietario y una raza existentes.

Iteración 2. Creación de horario, agenda y citas (tres semanas). Esta fue la iteración más extensa por la cantidad de pasos encadenados que tiene el módulo. Se construyó de forma incremental. Primero se definió la disponibilidad a partir de los horarios del veterinario. Luego se generó el calendario mediante un procedimiento almacenado que crea los bloques de atención según la duración de la cita y los parámetros de la agenda. Después se habilitó la reserva de la cita sobre un bloque libre del calendario y, por último, el manejo de estados de la cita, que permite confirmar, cancelar, reprogramar y bloquear un día. Para garantizar la integridad se incorporaron los triggers que sincronizan el estado del calendario con el de la cita y que impiden duplicar una cita activa sobre el mismo bloque. Cada paso se probó antes de avanzar al siguiente, verificando que no se generaran cruces de horario ni reservas sobre bloques ocupados.

Iteración 3. Landing page y portal del cliente (una semana). La landing se desarrolló al final del Sprint, después de tener lista la agenda y las citas del sistema principal,

porque su lógica es prácticamente la misma y se reutilizó. En esta iteración se construyó la página pública con la información del consultorio, los servicios y los veterinarios, y el portal del cliente, donde el propietario consulta la disponibilidad y reserva o cancela sus citas reutilizando el calendario y las reglas ya validadas en la iteración anterior.

Plan de trabajo del Sprint Backlog:

- Registrar y consultar propietarios con manejo de estado.
- Registrar y consultar pacientes asociados a su propietario, con trigger de identificador.
- Definir la disponibilidad y generar el calendario mediante procedimiento almacenado.
- Reservar citas sobre bloques libres y manejar sus estados (confirmar, cancelar, reprogramar, bloquear día).
- Incorporar los triggers de sincronización de estado y de prevención de citas duplicadas.
- Desarrollar la landing pública y el portal del cliente reutilizando la lógica de agenda y citas.
- Probar cada iteración antes de avanzar a la siguiente.

Incremento obtenido: Propietarios y pacientes funcionando, agenda con citas centralizadas y validadas, y portal público con reserva de citas del cliente.

4.5.3 Sprint 4: 1 de junio al 22 de junio de 2026 (tres semanas)

Objetivo del Sprint: Desarrollar el historial clínico del paciente con sus módulos especializados y completar las reglas de integridad del sistema.

Elementos seleccionados del Product Backlog: PB-09, PB-10, PB-11, PB-12, PB-13.

Al igual que el Sprint anterior, este Sprint se organizó en iteraciones por módulo, ya que el historial clínico está compuesto por varios registros que se construyen sobre la consulta médica del paciente. Los procedimientos almacenados y triggers de identificación y de borrado seguro se desarrollaron dentro de la iteración del módulo correspondiente. A continuación, se describe cada iteración con su duración aproximada.

Iteración 1. Anamnesis y catamnesis (una semana). Se desarrolló el registro de la anamnesis y de la catamnesis, que son la base de la consulta médica y de la que dependen los demás registros clínicos. Se incorporaron los triggers que generan sus identificadores y la validación de que la catamnesis pertenezca al veterinario que atiende. La catamnesis funciona como el punto de partida al que se asocian la vacunación, las prescripciones, los exámenes, los tratamientos y las especialidades.

Iteración 2. Vacunación, prescripciones y exámenes (una semana). Sobre la catamnesis se construyeron el registro de vacunación, las prescripciones con sus medicamentos asociados y los pedidos y resultados de exámenes. Cada uno validó la existencia de su catálogo correspondiente (tipo de vacuna, medicamento o tipo de examen) y se incorporaron los triggers de identificación y los procedimientos de borrado seguro para no eliminar registros enlazados al historial.

Iteración 3. Tratamientos y especialidades clínicas (una semana). Se desarrolló el registro de tratamientos con sus seguimientos y los módulos de especialidad clínica priorizados (mamas, oncología, traumatología y odontología). En esta iteración final se revisaron de forma integral los procedimientos almacenados, los triggers y las reglas de integridad de todo el sistema, y se ejecutaron las pruebas funcionales de cierre sobre el historial clínico completo.

Plan de trabajo del Sprint Backlog:

- Registrar anamnesis y catamnesis con sus triggers de identificación.
- Registrar vacunación validando el tipo de vacuna y el medicamento.
- Registrar prescripciones y sus medicamentos asociados.
- Registrar pedidos y resultados de exámenes con borrado seguro.
- Registrar tratamientos y seguimientos.
- Incorporar los módulos de especialidad clínica priorizados.

- Revisar de forma integral procedimientos almacenados, triggers y reglas de integridad.
- Ejecutar las pruebas funcionales de cierre del historial clínico.

Incremento obtenido: Historial clínico integrado, reglas de base de datos revisadas y prototipo preparado para la entrega.

4.5.4 Resumen de planificación de Sprint Backlog

Tabla 4. Planificación de Sprints y Sprint Backlog del prototipo

Sprint	Fechas	Objetivo	PBIs	Plan de trabajo	Incremento esperado
Sprint 1	16 mar – 5 abr 2026 (3 semanas)	Definir reglas de negocio, requerimientos, modelo de datos e infraestructura de seguridad.	PB-01, PB-02	Consolidar reglas de negocio. Depurar requerimientos. Elaborar el modelo de datos. Crear la estructura de tablas. Montar el motor de autenticación JWT. Verificar la integridad del modelo.	Modelo de datos, reglas base y motor de seguridad listos.
Sprint 2	6 abr – 19 abr 2026 (2 semanas)	Configurar consultorio y veterinarios, crear clientes y acceso, y desarrollar CRUDs y catálogos.	PB-03, PB-04, PB-05	Parametrizar consultorio y agenda. Registrar veterinarios y horarios. Crear clientes y habilitar el acceso (login interno y de clientes). CRUD de catálogos (especies, razas, exámenes, vacunas, medicamentos, servicios, enunciados). SPs de eliminación segura y pruebas.	Configuración base, acceso operativo y catálogos disponibles.
Sprint 3	20 abr – 31 may 2026 (6 semanas, con iteraciones)	Desarrollar CRUD particulares, agenda y citas, y la landing/portal del cliente.	PB-06, PB-07, PB-08	Iteración 1 (2 sem): propietarios y pacientes. Iteración 2 (3 sem): disponibilidad, calendario, reserva y estados de cita. Iteración 3 (1 sem): landing y portal del cliente (reutiliza la lógica de citas). SPs y triggers de cada módulo. Probar cada iteración antes de avanzar.	Propietarios, pacientes, agenda con citas validadas y portal del cliente.
Sprint 4	1 jun – 22 jun 2026 (3 semanas, con iteraciones)	Desarrollar el historial clínico y completar las reglas de integridad.	PB-09 a PB-13	Iteración 1 (1 sem): anamnesis y catamnesis. Iteración 2 (1 sem): vacunación, prescripciones y exámenes. Iteración 3 (1 sem): tratamientos y especialidades. Revisión integral de SPs y triggers. Pruebas funcionales de cierre.	Historial clínico integrado y prototipo preparado para entrega.

Nota. Listado de la planificación de Sprints y Sprint Backlog. Elaborado por el autor Calvopiña Daniel.

En la anterior tabla 4 se hace un resumen de la planificación de los Sprints. A diferencia de la tabla del Product Backlog, aquí sí se muestra la relación entre fechas, elementos seleccionados y plan de trabajo, porque esta información corresponde al Sprint Backlog y no a la lista general del producto.

Con esta planificación, el desarrollo del prototipo mantiene una relación clara entre Scrum y el trabajo realizado. El Product Backlog conserva la visión general del producto, mientras que cada Sprint Backlog muestra qué elementos fueron seleccionados, qué tareas se planificaron y qué incremento se esperaba obtener al finalizar la iteración. De esta manera, la planificación deja de ser una simple tabla de módulos y se convierte en una explicación del proceso real de construcción del sistema.

4.6 Modelado de procesos

El modelado de procesos permitió transformar la información obtenida en el levantamiento de campo en una representación ordenada del funcionamiento del consultorio. En esta sección no se vuelve a describir la entrevista como técnica de investigación, sino que se toma lo encontrado en ella para explicar cómo fluye actualmente la atención y cómo se reorganiza con el prototipo propuesto.

Para mantener coherencia con el trabajo de campo, el proceso actual se construyó a partir de tres fuentes: la conversación con el médico veterinario, la observación directa durante la jornada de atención y los formularios físicos utilizados en el consultorio. Estos documentos se incluyen como anexos, ya que sirven como evidencia del punto de partida desde el cual se diseñaron los formularios digitales del sistema.

4.6.1 Proceso actual de atención en el consultorio

En el proceso actual el cual está reflejado de una manera resumida en la tabla 5, el primer contacto con el propietario suele realizarse por llamada telefónica, WhatsApp o llegada directa al consultorio. Cuando el propietario solicita atención, el veterinario o el personal de apoyo revisa manualmente la disponibilidad de horario y confirma la atención de acuerdo con la memoria, la libreta o los espacios disponibles en la jornada.

Una vez que el propietario llega con la mascota, el proceso continúa con la búsqueda de la ficha clínica física. Si el paciente ya ha sido atendido antes, se intenta localizar su carpeta por número de ficha o por datos del propietario. Cuando el paciente es nuevo, se toma una ficha en blanco, se asigna un número manualmente y se registran los datos básicos del propietario y de la mascota. Este paso puede volverse lento cuando el propietario no recuerda datos como la edad exacta de la mascota, antecedentes de vacunación o información clínica previa.

Durante la consulta, el veterinario alterna entre examinar al animal y completar manualmente los campos de la ficha. En consultas generales se registran signos vitales, observaciones clínicas, diagnóstico presuntivo, diagnóstico definitivo, pronóstico y tratamiento. Cuando el caso requiere seguimiento, se utilizan hojas adicionales, como la hoja de control de tratamiento o el formulario de catamnesis oncológica. En casos donde se necesita ubicar lesiones o hallazgos, se emplea la hoja de ubicación anatómica.

El problema principal de este proceso es que la información queda distribuida en varios documentos físicos. Aunque estos formularios están bien estructurados desde el punto de vista clínico, su consulta depende de la disponibilidad de la carpeta y de la lectura manual

de las anotaciones. Además, no existen alertas automáticas para vacunación, desparasitación, tratamientos pendientes o citas futuras.

Tabla 5. Proceso actual de atención veterinaria

Etapa	Actividad actual	Responsable	Limitación identificada
Solicitud de atención	El propietario contacta por WhatsApp, llamada o llega directamente al consultorio.	Propietario / veterinario	No existe una agenda centralizada ni validación automática de disponibilidad.
Búsqueda de ficha	Se localiza la carpeta física del paciente por número o por datos del propietario.	Veterinario / apoyo administrativo	La búsqueda puede tardar varios minutos y depende del orden físico del archivo.
Registro de paciente nuevo	Se llena una ficha en papel y se asigna un número manualmente.	Personal del consultorio	Existe riesgo de duplicidad o campos incompletos.
Consulta clínica	El veterinario registra signos, diagnóstico y tratamiento en la ficha.	Médico veterinario	El espacio del formulario puede ser insuficiente y el registro interrumpe el flujo de atención.
Seguimiento	Se utilizan hojas adicionales para tratamientos, oncología o ubicación anatómica.	Médico veterinario	Los documentos no quedan integrados en un expediente digital único.
Cierre de atención	El propietario recibe indicaciones verbales y, cuando aplica, carnet físico.	Médico veterinario	El seguimiento depende de la memoria del propietario o del documento físico.

Nota: Proceso actual de atención de la veterinaria. Elaborado por el autor Calvopiña Daniel.

4.6.2 Proceso propuesto con el sistema

El proceso propuesto de manera resumida que está en la tabla 6 donde se está reorganizando el flujo de trabajo alrededor de un expediente digital. El sistema no elimina la lógica clínica utilizada por el consultorio, sino que la traslada a formularios digitales conectados entre sí. De esta forma, la información del propietario, de la mascota, de las citas y de la consulta médica queda centralizada en una sola plataforma.

El nuevo proceso inicia con la parametrización del consultorio, la configuración de servicios, horarios de atención, veterinarios y agenda. A partir de esa base, el propietario puede solicitar una cita desde el portal o el personal interno puede registrarla desde el sistema administrativo. Antes de confirmar la cita, el sistema consulta la disponibilidad del calendario veterinario para evitar cruces de horario.

Cuando el paciente llega a consulta, el veterinario puede buscarlo por nombre, cédula del propietario o identificador del paciente. Si es un paciente nuevo, se crea su expediente digital y se relaciona con el propietario. Luego, durante la atención, se registra la anamnesis, catamnesis, diagnóstico, tratamiento, vacunación, prescripción o pedido de examen, según corresponda. La información queda asociada al paciente y puede consultarse posteriormente desde su historial clínico.

El proceso propuesto también contempla la presencia digital del consultorio mediante una landing pública. Esta página permite mostrar información general, servicios, datos de contacto, veterinarios, imágenes y anuncios. De esta manera, el consultorio deja de depender únicamente de recomendaciones personales y cuenta con un canal formal para darse a conocer y recibir solicitudes de atención.

Tabla 6. Proceso propuesto con el prototipo

Etapa	Actividad propuesta	Componente del sistema	Resultado esperado
Configuración inicial	Registrar consultorio, servicios, horarios, veterinarios y parámetros de agenda.	Sistema interno	Base operativa para que los módulos posteriores funcionen correctamente.
Visibilidad pública	Mostrar información del consultorio, servicios, veterinarios y anuncios.	Landing pública	Mayor alcance para nuevos propietarios y orientación antes de solicitar atención.
Registro de cliente	Crear cuenta del propietario y validar datos de contacto.	Portal cliente	Cliente registrado y habilitado para gestionar mascotas y citas.
Registro de paciente	Crear expediente digital de la mascota y asociarlo al propietario.	Sistema interno / portal	Paciente disponible para citas, historial clínico y seguimientos.
Agendamiento	Consultar disponibilidad y crear cita según veterinario, fecha y horario.	Agenda y citas	Disminución de cruces de horario y mejor control de atención.
Atención médica	Registrar anamnesis, catamnesis, diagnóstico, tratamiento y observaciones.	Historial clínico	Información clínica centralizada y consultable.
Seguimiento	Registrar vacunas, prescripciones, exámenes, tratamientos y controles.	Módulos clínicos	Trazabilidad de la evolución del paciente.

Nota: Proceso propuesto con el prototipo veterinaria. Elaborado por el autor Calvopiña Daniel.

4.6.3 Flujo de atención clínica propuesto

El flujo de atención clínica propuesto se puede evidenciar en la tabla 7 donde se organizó para que cada registro se relacione con el paciente y con la cita correspondiente. Esta estructura permite que el historial clínico no sea una colección de formularios separados, sino una secuencia ordenada de eventos médicos. El objetivo es que el veterinario pueda reconstruir la evolución del paciente sin revisar hojas físicas ni depender de anotaciones aisladas.

Tabla 7. Flujo de atención clínica propuesto

Paso	Actividad	Entrada	Salida
1	Buscar o registrar propietario.	Cédula/RUC, nombres y contacto.	Propietario identificado en el sistema.
2	Buscar o registrar paciente.	Nombre, especie, raza, sexo, color y fecha de nacimiento.	Paciente asociado al propietario.
3	Crear o confirmar cita.	Veterinario, fecha, hora y motivo.	Cita registrada con estado correspondiente.
4	Registrar anamnesis.	Antecedentes y datos informados por el propietario.	Antecedentes iniciales vinculados al paciente.
5	Registrar catamnesis.	Signos vitales, examen físico y observaciones.	Registro clínico de la consulta.
6	Registrar diagnóstico y plan médico.	Hallazgos clínicos.	Diagnóstico, pronóstico y tratamiento.
7	Agregar prescripción, vacunas, exámenes o seguimiento.	Indicaciones médicas.	Registros clínicos complementarios asociados.
8	Consultar historial en futuras visitas.	Paciente registrado.	Vista cronológica de atenciones previas.

Nota: Proceso propuesto para la atención de la veterinaria. Elaborado por el autor Calvopiña Daniel.

4.6.4 Flujo de agendamiento de citas

El agendamiento de citas se consideró uno de los procesos prioritarios ya que en el consultorio actual las solicitudes llegan por canales informales y pueden generar cruces de horarios, en el prototipo, la cita se apoya en el calendario del veterinario, el cual se genera según los horarios de atención y la duración configurada para cada consulta, este flujo se puede presenciar en la tabla 8, donde el flujo contempla dos formas de ingreso: desde el sistema interno, cuando el personal agenda una atención, o desde el portal del cliente, cuando

el propietario solicita una cita. En ambos casos, la disponibilidad debe consultarse antes de confirmar el turno. Una vez creada la cita, el calendario cambia de estado para evitar que el mismo bloque horario sea reservado.

Tabla 8. Flujo de agendamiento de citas

Paso	Actividad	Regla aplicada
1	Seleccionar veterinario y fecha de atención.	Solo se muestran veterinarios activos y con horarios configurados.
2	Consultar disponibilidad del calendario.	El sistema muestra bloques libres y oculta los ocupados, cancelados o bloqueados.
3	Seleccionar paciente y motivo de atención.	La cita debe estar asociada a una mascota existente.
4	Confirmar cita.	El sistema registra la cita y actualiza el estado del calendario.
5	Cancelar o reprogramar si es necesario.	El estado de la cita y del calendario debe mantenerse sincronizado.

Nota. Flujo de agendamiento de citas propuesto. Elaborado por el autor Calvopiña Daniel.

4.7 Desarrollo y validación funcional del prototipo

El desarrollo del prototipo no se documenta mediante la inclusión completa del código fuente, sino mediante la explicación de los módulos implementados, la forma en que se construyeron y la evidencia funcional obtenida durante las pruebas. En los casos donde resulta necesario mostrar lógica técnica específica, se mencionan procedimientos almacenados y triggers relevantes, explicando su finalidad sin incluir el código completo.

Debido a que el proyecto fue trabajado bajo un enfoque iterativo con Scrum, la validación funcional se realizó junto con el desarrollo de cada módulo. Por esta razón, en esta sección se integran la explicación del desarrollo, las interfaces del prototipo y las pruebas funcionales, en lugar de separarlas en apartados aislados. Esta decisión permite mostrar que cada incremento fue revisado conforme se construyó.

El desarrollo del prototipo siguió la planificación de sprints presentada en la sección 4.5. Cada Sprint agrupó los módulos relacionados, de modo que esta sección describe el trabajo realizado y su validación funcional organizados por Sprint. Para cada módulo se indica cómo se construyó y cómo se verificó su funcionamiento.

4.7.1 Sprint 1: base del sistema

El primer Sprint estableció la base sobre la que se construyó el resto del sistema. Comprende las reglas de negocio y el modelado de datos, y el módulo de autenticación que protege el acceso interno.

a) Reglas de negocio y modelado de datos

El primer paso del desarrollo fue convertir la información levantada en reglas de negocio y en un modelo de datos. A partir de las entrevistas, la observación y los documentos físicos del consultorio se definieron las entidades necesarias y la forma en que se relacionan entre sí.

El modelado se realizó como un modelo entidad-relación organizado alrededor del consultorio como entidad raíz, del que se desprenden los datos institucionales, los veterinarios, los servicios, los parámetros de agenda, los clientes, los pacientes, las citas y la información clínica. El modelo no copió los formularios físicos, sino que los transformó en entidades relacionadas para evitar fichas duplicadas y permitir consultar la información por módulos. Sobre este modelo se creó la estructura de tablas en PostgreSQL que sostiene todo el sistema. El diagrama entidad-relación completo se presenta en el Anexo 7.

b) Módulo de autenticación y seguridad

El módulo de autenticación fue el primer componente funcional necesario para proteger el acceso al sistema interno. Su propósito es permitir que solo usuarios autorizados ingresen al sistema administrativo del consultorio, este módulo nos permite validar el usuario y contraseña, generando una sesión y permite identificar el rol asociado al usuario manejado desde la propia base de datos.

En el backend, la autenticación interna se gestiona mediante un controlador de autenticación y un servicio encargado de validar credenciales y generar el token correspondiente, mientras que, en el frontend, la pantalla de login permite ingresar las credenciales y cuando la autenticación es correcta, redirige al usuario hacia el dashboard interno del sistema en la tabla 18 se puede visualizar las funcionalidades aplicadas en este módulo y una visualización en la figura 1 del módulo de login funcionando dentro del prototipo.

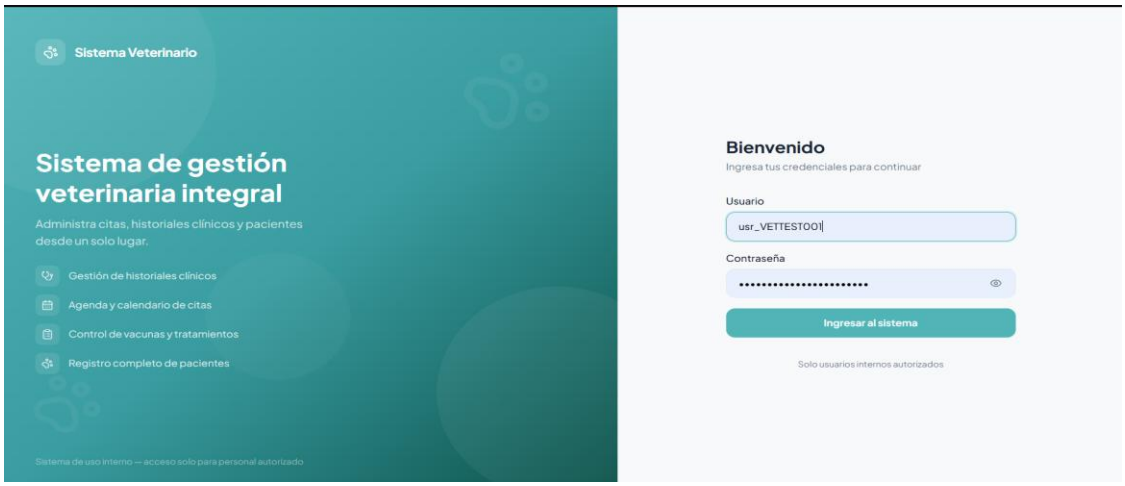
Tabla 9. Desarrollo y validación del módulo de autenticación

Funcionalidad	Cómo se implementó	Validación realizada	Resultado
Manejo de usuarios y roles en la base de datos	creación de roles determinados para cada usuario que se realice y creación de usuarios para poder conectarse al sistema gestionado por la base de datos	Creación de nuevos usuarios en la base de datos y comprobar su funcionalidad iniciando sesión.	Se puede ingresar a la base de datos mediante un pull de conexiones donde este usuario tiene accesos limitados.
Login interno	Pantalla de acceso conectada al endpoint de autenticación interna.	Ingresar usuario y contraseña válidos.	El sistema permite ingresar al dashboard.
Protección de rutas	Guardas de navegación que verifican si existe sesión activa.	Intentar acceder a rutas internas sin sesión.	El sistema redirige al login.
Diferenciación de usuario	Token con datos de usuario y rol.	Verificar acceso según tipo de usuario.	Solo usuarios internos ingresan al sistema administrativo.
Cierre de sesión	Limpieza de token y datos de sesión.	Cerrar sesión desde el sistema.	El usuario vuelve al login.

Nota. La tabla corresponde a las funcionalidades aplicadas en el módulo 4.7.1.2Elaborado por el autor Calvopiña Daniel.

Figura 1

Pantalla de inicio de sesión del sistema interno



Nota. La captura corresponde al módulo de autenticación descrito en la sección 4.7.1.2. Elaborado por el autor Calvopiña Daniel

4.7.2 Sprint 2: configuración del consultorio

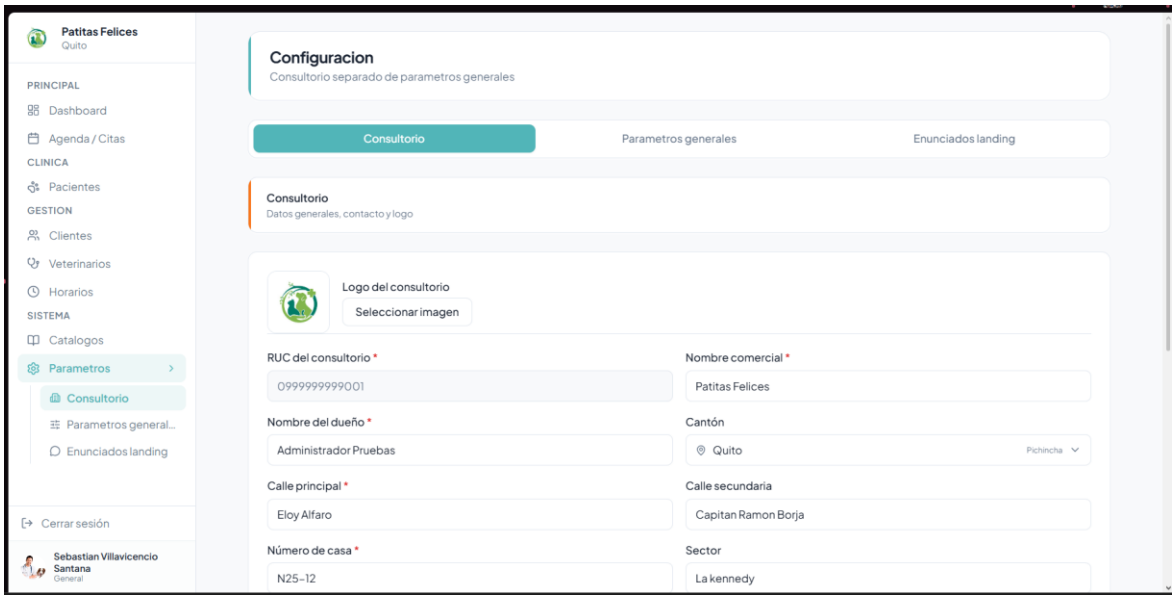
El segundo Sprint configuró la información base del consultorio, los veterinarios y los diversos catálogos y por último se habilitó el acceso operativo de los usuarios.

a) Módulo de parametrización del consultorio

El módulo de parametrización permite registrar la información básica del consultorio y configurar valores que serán utilizados por otros módulos, antes de trabajar con citas, servicios o portal público, fue necesario definir datos como nombre del consultorio, RUC, dirección, contactos, logo, misión, visión, horario de atención, código de area y duración de consulta estaos campos están separados en 3 pestañas diferentes de este módulo cada uno siento consultorio, parámetros generales y por ultimo enunciados landing esto se puede visualizar en la figura 2.

Figura 2

Módulo de parametrización del consultorio



Nota. La captura corresponde al módulo de parametrización descrito en la sección 4.7.2.1. Elaborado por el autor Calvopiña Daniel

Tabla 10. Desarrollo y validación del módulo de parametrización

Funcionalidad	Descripción	Validación
Datos del consultorio	Registro de RUC, nombre, dirección, teléfono, celular, correo y responsable.	Los datos se guardan y pueden consultarse nuevamente.
Logo institucional	Carga de imagen del consultorio en almacenamiento externo.	El logo aparece en la interfaz y puede consultarse mediante URL.
Textos institucionales	Registro de misión, visión y frase o refrán del consultorio.	La información queda disponible para el portal público.
Parámetros de agenda	Configuración de duración de consulta y horarios generales.	Los datos se utilizan para generar calendario de disponibilidad.
Parámetros de servicio	Registro de numero inicial de prescripción	Numero de la siguiente prescripción que se vaya a emitir.
Enunciados para landing	Registro de imágenes o anuncios visibles en la página pública.	Solo los enunciados activos se muestran al visitante.

Nota. Tabla en la cual se puede revisar las funcionalidades implementadas en el módulo de la sección 4.7.2.1 Elaborado por el autor Calvopiña Daniel.

Esta parametrización también se relaciona con la landing pública, ya que parte de la información registrada se utiliza para presentar el consultorio a nuevos clientes y también los parámetros de agenda permiten definir la duración estándar de las consultas y el

comportamiento inicial del calendario veterinario por último este apartado también nos dejara colocar los anuncios y definir las imágenes que se van a proyectar dentro de la página pública, las funcionalidades que se implementaron en este módulo se pueden revisar en la tabla 9 en la cual se explica la funcionalidad y la descripción de cada una de ellas.

b) Módulo de veterinario

Este constituye el módulo principal del segundo *sprint*, ya que está enfocado en la gestión de perfiles del personal veterinario. Su implementación permite que los profesionales accedan al sistema interno para administrar los registros y la atención de los pacientes de manera eficiente e intuitiva, adicionalmente, la información de estos perfiles se consume de forma dinámica para mostrar a los veterinarios activos en la *landing page* de la clínica, brindando mayor visibilidad al equipo médico las funcionalidades de este módulo se detallan en la tabla 20 y se visualizan el módulo funcionando de una manera visual en la figura 3.

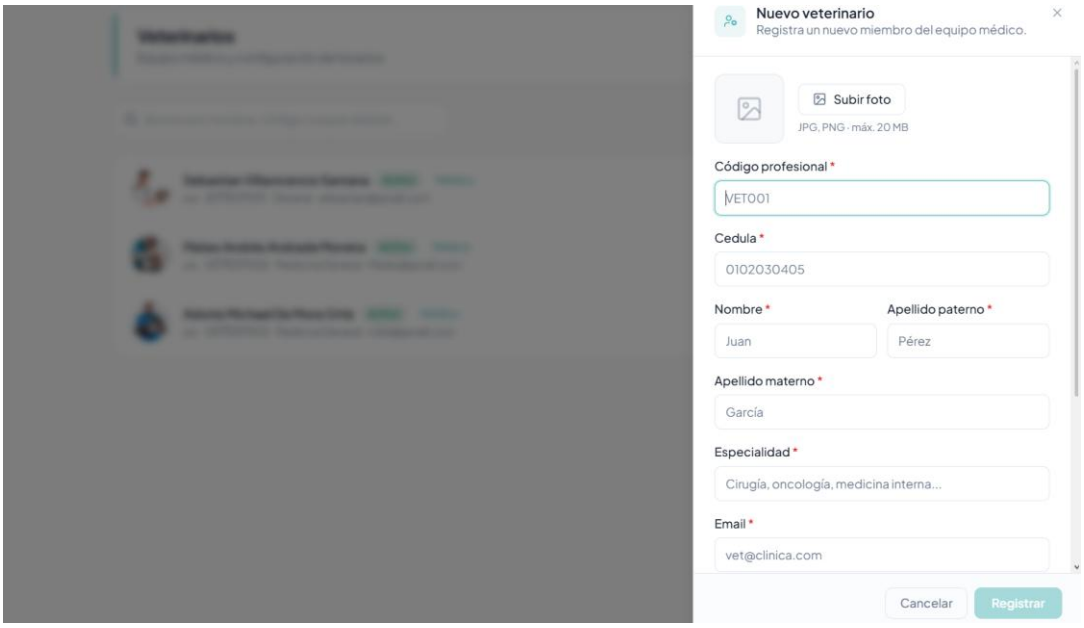
Tabla 11. Desarrollo y validación del módulo de Veterinarios

Funcionalidad	Cómo se desarrolló	Validación
Registro de Veterinarios	Formulario con numero de licencia profesional, datos personales, dirección, correo y contacto y la creación de su usuario en la base de datos junto a un rol de veterinario.	El Veterinario se guarda y se puede consultar desde el listado, además de iniciar sesión dentro de la plataforma
Actualización de Veterinarios	Edición de datos del veterinario.	Los cambios se reflejan al volver a consultar el registro.
Visualización de veterinarios	Se extracción de todos veterinarios registrados	Extracción de los datos

Nota. Lista de funcionalidades implementadas en la sección 4.7.2.2. Elaborado por el autor Calvopiña Daniel.

Figura 3

Módulo de veterinarios



Nota. La captura evidencia lo detallado en la sección 4.7.2.2. Elaborado por el autor Calvopiña Daniel.

c) Módulo de parametrización de catalogo

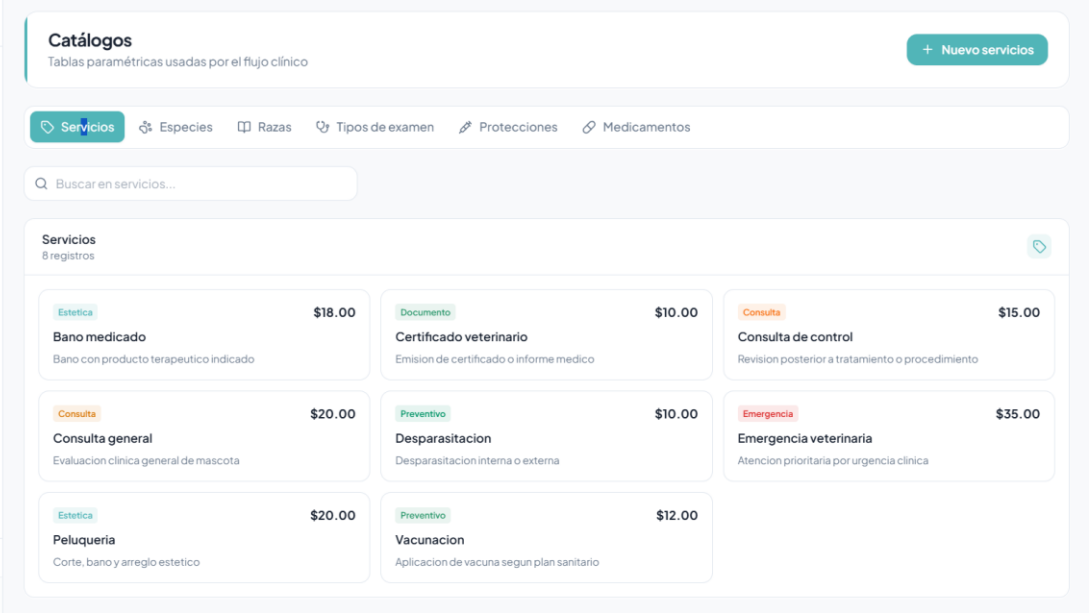
Este módulo se enfocó en el desarrollo CRUDs de datos donde en la tabla 21 se detalla las funcionalidades que tiene cada de las categorías para los catálogos base del sistema. La estructuración de estas entidades es un requisito fundamental, ya que proveerán la información necesaria para el funcionamiento de los módulos que se implementarán en fases posteriores. Los catálogos gestionados en esta etapa incluyen: especies, razas, servicios, tipos de exámenes, medicamentos y clasificaciones terapéuticas todas estas categorías se pueden evidenciar en pestañas en la figura 4 donde claramente se puede apreciar cada una de los catálogos anteriormente mencionados.

Tabla 12. Desarrollo y validación del módulo de catalogo

Funcionalidad	Cómo se desarrolló	Validación
Registro	Formulario creación de las categorías	El servicio se guarda dentro de las tablas.
Visualización	Se hizo una extracción de datos para su visualización	Visualización en los listados.
Actualización	Edición de datos de las diversas categorías.	Los cambios se reflejan al volver a consultar el registro.
Eliminación	Eliminación de servicios	Se eliminará siempre y cuando otras tablas no dependan del este catálogo para existir

Nota. Lista de funcionalidades que tiene cada una de las pestañas de las categorías dentro del módulo de catálogos. Elaborado por el autor.

Figura 4



Nota. La captura corresponde al módulo de catálogos descrito en la sección 4.7.2.3. Elaborado por el autor Calvopiña Daniel

4.7.3 Sprint 3: procesos principales y portal

El tercer Sprint desarrolló los procesos centrales de atención: el registro de clientes y pacientes, la agenda y las citas, y el portal público con el portal del cliente.

a) Módulo de clientes y pacientes

El módulo de clientes y pacientes constituye la base del expediente digital. El cliente representa al propietario o responsable de la mascota, mientras que el paciente representa al animal que recibe atención médica. Esta separación permite que un mismo propietario tenga una o varias mascotas asociadas.

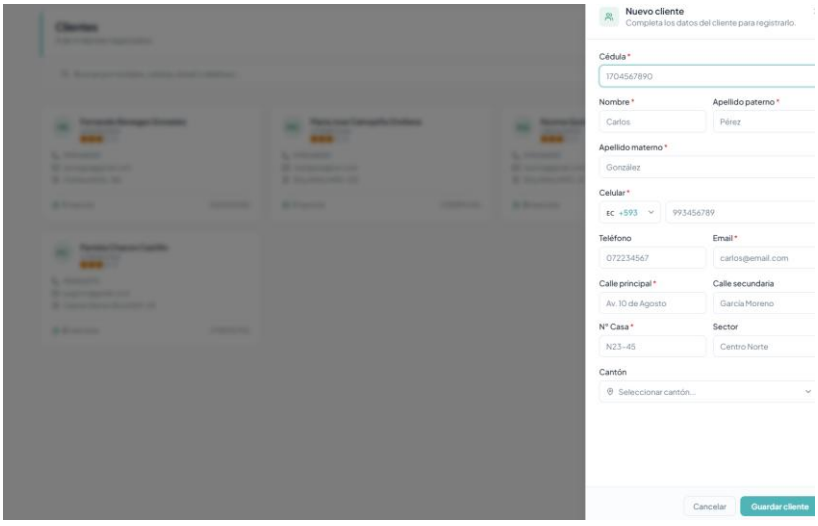
Tabla 13. Desarrollo y validación del módulo de clientes y pacientes

Funcionalidad	Cómo se desarrolló	Validación
Registro de cliente	Formulario con datos personales, dirección, correo y contacto.	El cliente se guarda y se puede consultar desde el listado.
Actualización de cliente	Edición de datos del propietario.	Los cambios se reflejan al volver a consultar el registro.
Registro de paciente	Formulario de mascota con nombre, sexo, color, especie, raza e imagen.	La mascota queda asociada al propietario.
Consulta por cliente	Visualización de mascotas asociadas a un propietario.	Se muestran solo los pacientes vinculados al cliente seleccionado.
Perfil de paciente	Vista con datos generales y acceso al historial clínico.	El veterinario puede acceder a la información médica del paciente.

Nota. Elaborado por el autor.

Figura 5

Registro y consulta de clientes

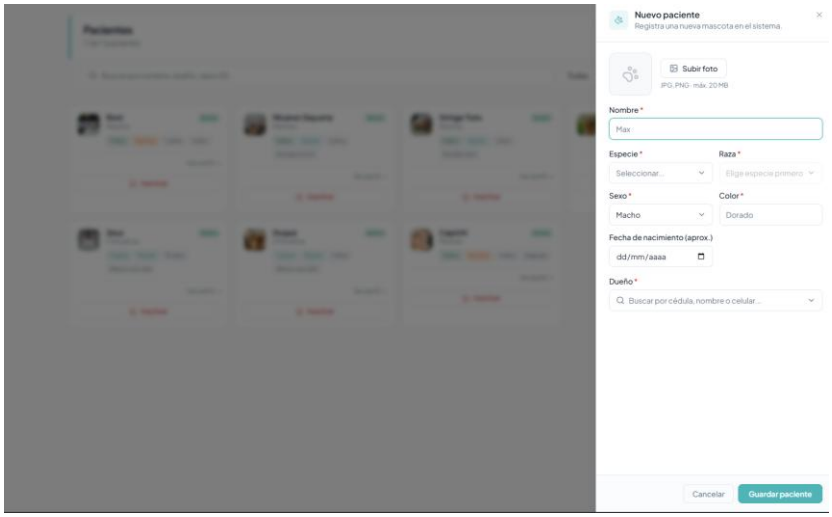


Nota. La captura corresponde al módulo de clientes descrito en la sección 4.7.3.1. Elaborado por el autor Calvopiña Daniel.

En la tabla 22 anteriormente mostrada se puede ver las funcionalidades de este módulo siendo parte el sistema interno donde el personal puede crear, consultar y actualizar propietarios y mascotas. En el portal del cliente, el propietario también puede registrar sus propias mascotas. En ambos casos, el sistema valida que la mascota quede asociada a un cliente existente y que los datos de especie y raza sean seleccionados desde catálogos previamente definidos en las figuras 5 y 6 se pueden visualizar las 2 partes del módulo con su correcto funcionamiento

Figura 6

Registro y consulta de pacientes



Nota. La captura corresponde al módulo de clientes descrito en la sección 4.7.3.1. Elaborado por el autor Calvopiña Daniel

b) Módulo de creación de horario, agenda de un veterinario y citas

El módulo de agenda fue uno de los más importantes del prototipo, debido a que responde directamente a los problemas de cruces de horario y solicitudes dispersas por

WhatsApp o llamadas. Para su desarrollo se configuraron horarios de veterinarios, generación de bloques de calendario, estados de disponibilidad y creación de citas.

El agendamiento funciona en 3 partes, la primera parte comienda con la parametrización de la duración de cada una de las citas que llegan al consultorio veterinario, la segunda parte comienza con la parametrización del horario que tiene cada uno de los veterinarios y por último es la generación del calendario del veterinario la primera parte se realizó en la parametrización de datos del consultorio por lo que este sprint lo que se realizo fue la creación de que cada uno de los veterinarios puede crear su horario y después puedan generar el calendario hasta fin de año.

Tabla 14. Desarrollo y validación del módulo de agenda y citas

Funcionalidad	Regla aplicada	Validación
Horario del veterinario	Se crea el horario del veterinario	Se muestra el horario de trabajo del veterinario
Generar calendario	Crear bloques según horario del veterinario y duración de consulta.	El sistema muestra disponibilidad por fecha.
Consultar disponibilidad	Mostrar solo horarios disponibles o según estado.	El usuario puede seleccionar un bloque libre.
Crear cita	Asociar paciente, veterinario, calendario y motivo.	La cita queda registrada y el bloque cambia de estado.
Confirmar cita	Cambiar estado de la cita a confirmada.	La cita aparece como confirmada en la agenda.
Cancelar cita	Actualizar cita y calendario según acción.	El bloque se libera o se marca según corresponda.
Bloquear día	Impedir uso de horarios de un veterinario en una fecha.	No se permite agendar en los bloques bloqueados.

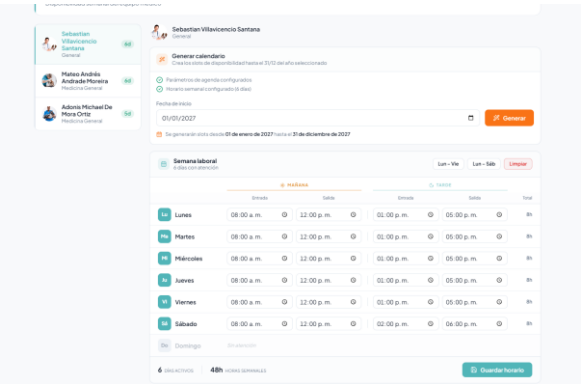
Nota. Lista de funcionalidades implementada en el módulo de creación de agendas, horario y creación de citas. Elaborado por el autor Calvopiña Daniel.

Después de estas 3 partes viene el apartado de las citas, para poder crear necesitamos tener un veterinario designado una mascota y seleccionar una parte del calendario del mismo veterinario que habíamos seleccionado.

En este módulo que es muy importante se añadieron funcionalidades importantes como la reprogramación de una cita o la cancelación de una cita , también otra cosa que se tiene en este

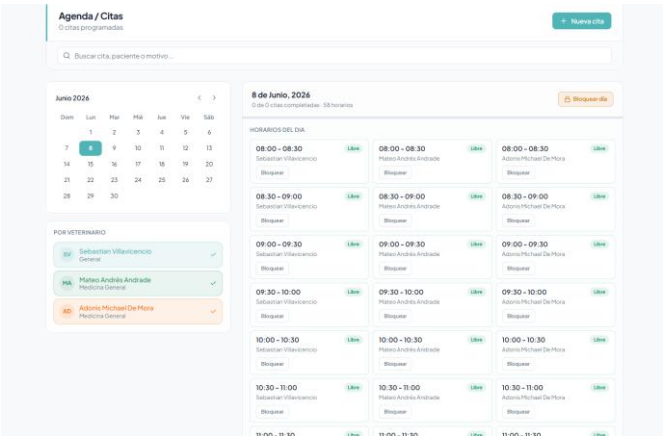
módulo es el bloqueo de un día por parte del veterinario por si llega a ocurrir un imprevisto y necesita ausentarse durante un periodo de tiempo y no puede atender en el consultorio, este flujo y funcionalidades se pueden revisar en la tabla 23, además en las figuras 7 y 8 se pueden presenciar las pantallas de la definición de un horario de un veterinario y el calendario de atención del mismo con su correcto funcionamiento.

Figura 7
Creación de horario de veterinario



Nota. La captura corresponde al módulo de horario, agenda y citas descrito en la sección. 4.7.3.2. Elaborado por el autor Calvopiña Daniel.

Figura 8
Agenda veterinaria y creación de citas



Nota. La captura corresponde al módulo de horario, agenda y citas descrito en la sección 4.7.3.2. Elaborado por el autor Calvopiña Daniel.

c) Módulo de portal público y portal del cliente

El portal público se desarrolló como respuesta a la baja visibilidad digital del consultorio. Durante el levantamiento de información se identificó que el veterinario era conocido principalmente por clientes antiguos o recomendaciones, por lo que se consideró necesario contar con una página web que funcione como carta de presentación.

La landing muestra información del consultorio, servicios, veterinarios, imágenes y medios de contacto, además, sirve como punto de entrada para que un propietario pueda registrarse, iniciar sesión o solicitar una cita. De esta manera, el sistema no solo mejora la gestión interna, sino que también apoya la forma en que el consultorio se da a conocer.

Tabla 15. Desarrollo y validación del portal público

Funcionalidad	Propósito	Resultado esperado
Visualización de información del consultorio	Mostrar nombre, datos de contacto, dirección y textos institucionales.	El visitante encuentra información básica sin llamar directamente.
Servicios publicados	Presentar los servicios que ofrece el consultorio.	El propietario puede conocer qué atenciones están disponibles.
Veterinarios visibles	Mostrar información de profesionales disponibles.	El visitante identifica quién atiende en el consultorio.
Imágenes y anuncios	Mostrar carrusel o comunicados administrados desde el sistema interno.	La página se mantiene actualizada sin modificar código.
Acceso a cliente	Permitir registro o inicio de sesión.	El propietario puede pasar de visitante a usuario registrado.
Registro de mascotas	Permite registrar a nuevas mascotas	El propietario permite
Toma de cita	Permite la toma de la cita para tu mascota	Se registra una cita

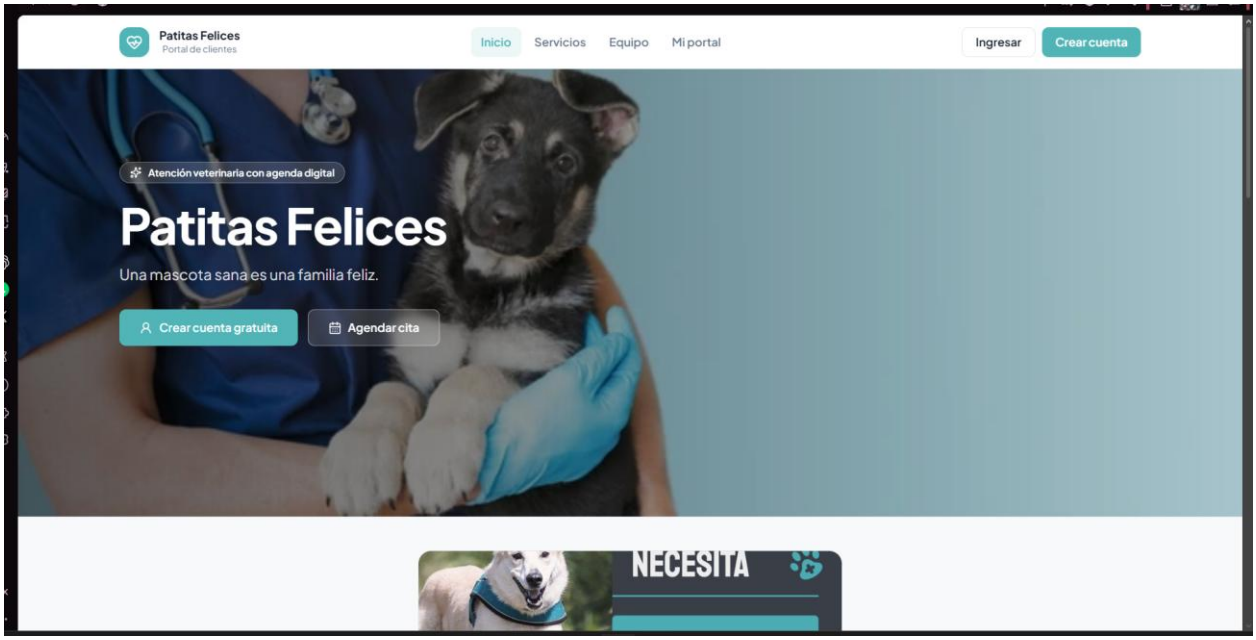
Nota. Listado de funcionalidades del portal público. Elaborado por el autor Calvopiña Daniel.

Una vez que el propietario inicia sesión, el portal del cliente le permite consultar y actualizar sus datos, registrar sus mascotas y, sobre todo, gestionar sus citas, además el cliente consulta la disponibilidad del veterinario y puede lograr una reserva de una cita sobre un bloque libre y puede cancelarla cuando lo necesita. Esta reserva reutiliza el mismo calendario y las mismas reglas de la agenda del sistema interno, del modo que una cita creada desde el portal queda registrada y sincronizada igual que una creada por el consultorio, dichas

funcionalidades las tenemos también en la tabla 24 explicado también su propósito y resultado esperado, además se puede ver una breve imagen introductoria de la página inicial del módulo en la imagen 9 de una manera resumida y funcional.

Figura 9

Página pública del consultorio



Nota. La captura corresponde al portal público descrito en la sección 4.7.3.3. Elaborado por el autor Calvopiña Daniel.

4.7.4 Sprint 4: historial clínico

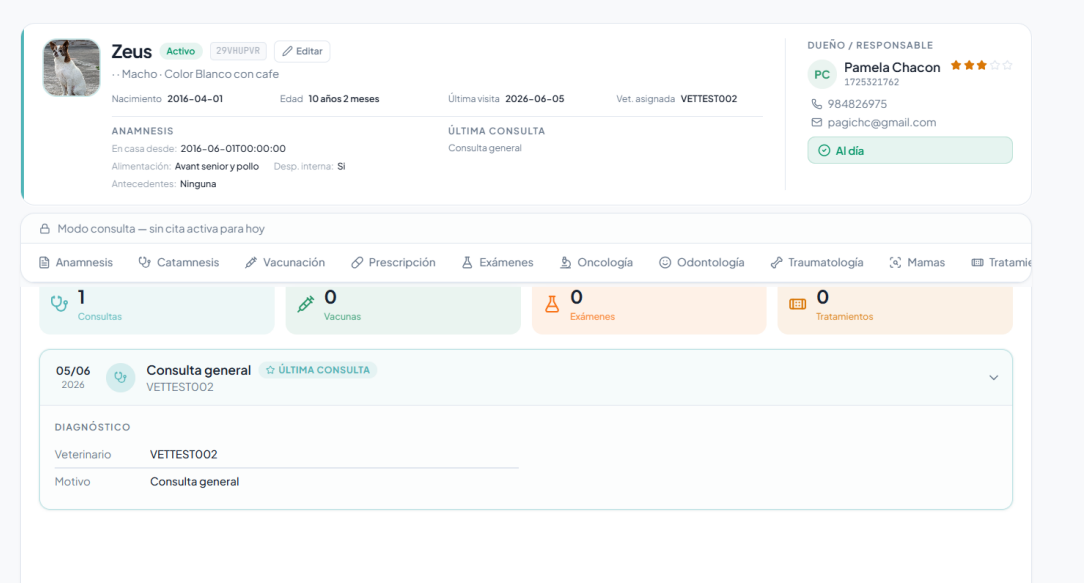
El cuarto Sprint completó la parte médica del sistema, con el historial clínico, las prescripciones, los exámenes, los tratamientos y los módulos de especialidad, para este módulo se añadió un método de control que únicamente cuando se tenga una cita se pueda atender al paciente caso contrario la información estará únicamente para su consulta, además otra medida de protección que se añadió fue que únicamente el veterinario que tiene la cita ese día puede crear y llevar a cabo la cita médica al paciente.

a) Módulo de historial clínico

El módulo de historial clínico permite consultar en una sola vista la información médica del paciente, esta funcionalidad reemplaza la búsqueda manual de fichas y facilita revisar antecedentes, consultas anteriores, vacunaciones, tratamientos, prescripciones, exámenes y especialidades clínicas.

Figura 10

Perfil clínico e historial del paciente



Nota. La captura corresponde al módulo de historial clínico descrito en la sección 4.7.4.1. Elaborado por el autor Calvopiña Daniel.

El historial se construye a partir de la relación entre paciente, cita, anamnesis y catamnesis. La anamnesis recoge antecedentes e información inicial aportada por el propietario, mientras que la catamnesis registra datos objetivos de la consulta, como signos vitales, examen físico, observaciones, diagnóstico, pronóstico y tratamiento si llegaran a surgir, cabe que aclarar que la vacunación y la desparasitación se encuentran en una sola

pestaña y la figura 10 tenemos la pantalla donde se muestran ya las distintas funcionalidades del apartado del historial clínico.

Tabla 16. Desarrollo y validación del módulo de historial clínico

Funcionalidad	Descripción	Validación
Consulta de historial	Vista general del paciente con registros clínicos asociados.	El veterinario puede revisar información anterior desde el perfil.
Registro de anamnesis	Formulario de antecedentes, vacunación previa, desparasitación y nutrición.	El registro queda asociado al paciente y cita.
Registro de catamnesis	Formulario de signos vitales, examen físico, diagnóstico y tratamiento.	La consulta queda disponible en el historial.
Registro de vacunación y desparasitaciones.	Control de vacunas aplicadas, tipo de vacuna y las desparasitaciones.	La vacunación y desparasitaciones aparecen en el historial preventivo.

Nota. Lista de funcionalidades del módulo de historia clínica. Elaborado por el autor Calvopiña Daniel.

b) Módulo de prescripciones, exámenes y tratamientos

Este módulo complementa la historia clínica y permite registrar las indicaciones médicas posteriores a la consulta dichas funcionalidades se encuentran de forma detalla en la tabla 26 y en las figuras 11, 12 y 13 se puede ver brevemente las funcionalidades explicadas a continuación. Las prescripciones se utilizan para registrar medicamentos, dosis, frecuencia y duración para un tratamiento a una enfermedad o padecimiento, mientras que los exámenes se dividen en 2 partes fundamentales la primera es la pedida de los exámenes donde el veterinario se encarga de indicar que examen tiene que realizarse al paciente y en qué condiciones mientras que por otra parte tenemos el registro de resultado, donde este es la entrega de los exámenes pedidos anteriormente junto con un análisis de parte del veterinario.

Los tratamientos y seguimientos se desarrollaron para representar aquellos casos en los que el paciente requiere varias sesiones de control, estas funcionalidades se basaron en la hoja física de seguimiento de tratamientos, donde el veterinario registra fecha, peso, temperatura,

medicamento administrado, medicamento prescrito para su correcto tratamiento y se tenga un historial sobre en cada uno de los tratamientos que se realizó y que se le inyecto al paciente.

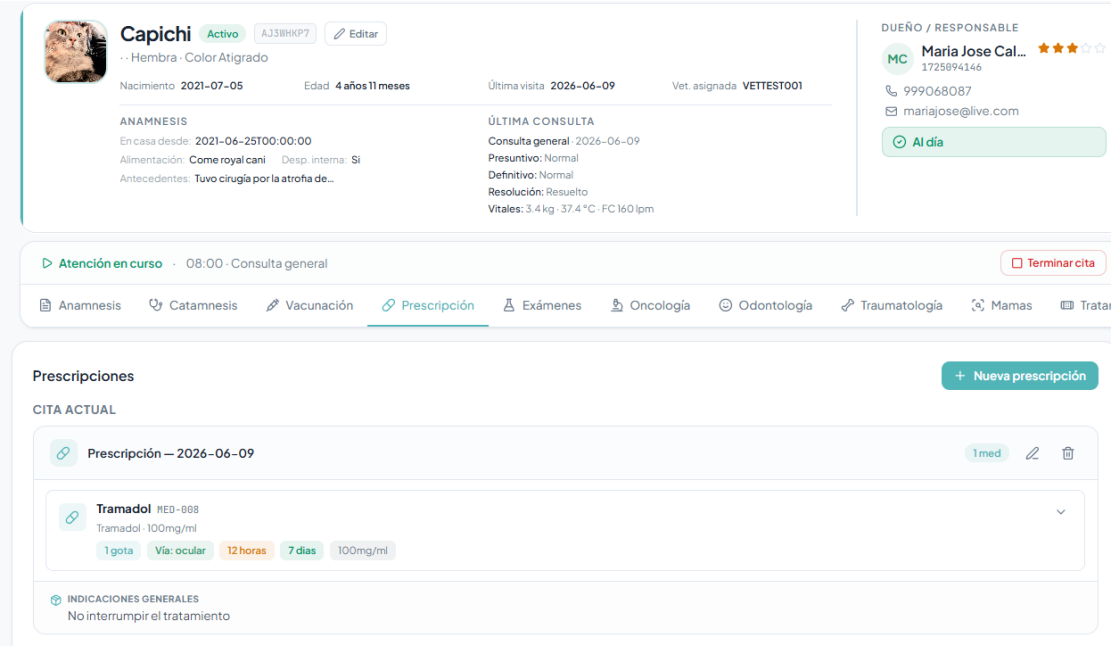
Tabla 17. Desarrollo y validación de prescripciones, exámenes y tratamientos

Funcionalidad	Descripción	Validación
Prescripción médica	Registro de indicaciones y medicamentos asociados a una consulta.	La prescripción queda vinculada a la catamnesis.
Medicamentos prescritos	Detalle de medicamento, dosis, frecuencia y duración.	El medicamento aparece dentro de la prescripción.
Pedido de examen	Solicitud de examen complementario por paciente y consulta.	El pedido queda visible en el historial.
Resultado de examen	Registro del resultado asociado a pedido o consulta.	El resultado puede consultarse desde el historial.
Tratamiento	Registro de tratamiento indicado para el paciente.	El tratamiento se asocia a la catamnesis.
Seguimiento de tratamiento	Registro de controles posteriores al tratamiento.	Cada sesión queda asociada al tratamiento correspondiente.

Nota. Listado de funcionalidades de prescripciones, exámenes y tratamientos. Elaborado por el autor Calvopiña Daniel.

Figura 11

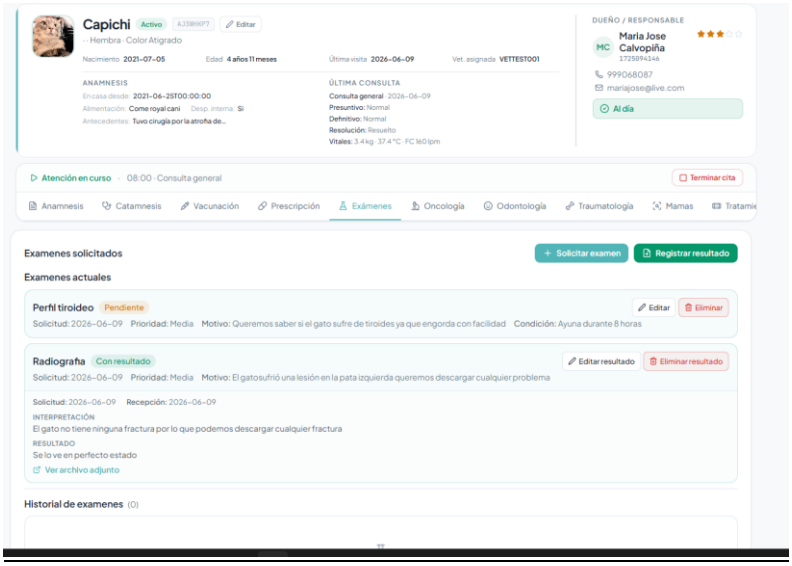
Prescripciones



Nota. La captura corresponde al módulo descrito en la sección 4.7.4.2. Elaborado por el autor Calvopiña Daniel.

Figura 12

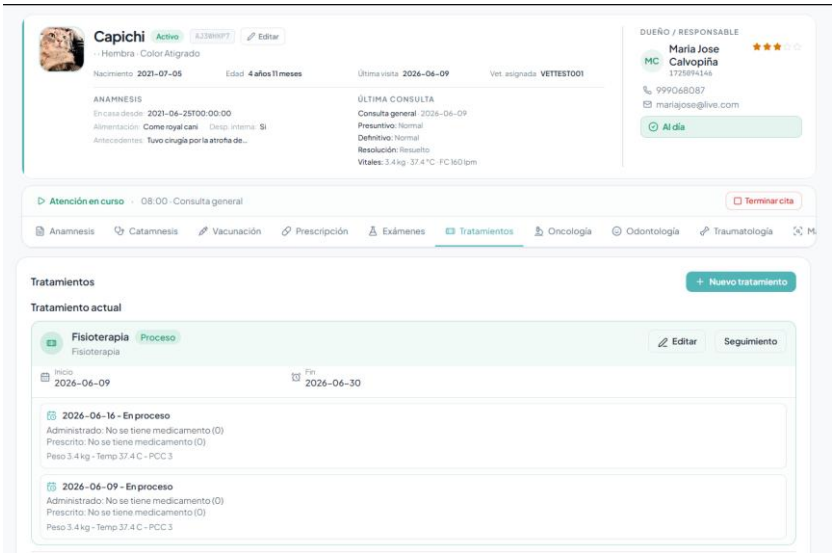
Exámenes



Nota. La captura corresponde al módulo descrito en la sección 4.7.4.2. Elaborado por el autor Calvopiña Daniel.

Figura 13

Tratamientos



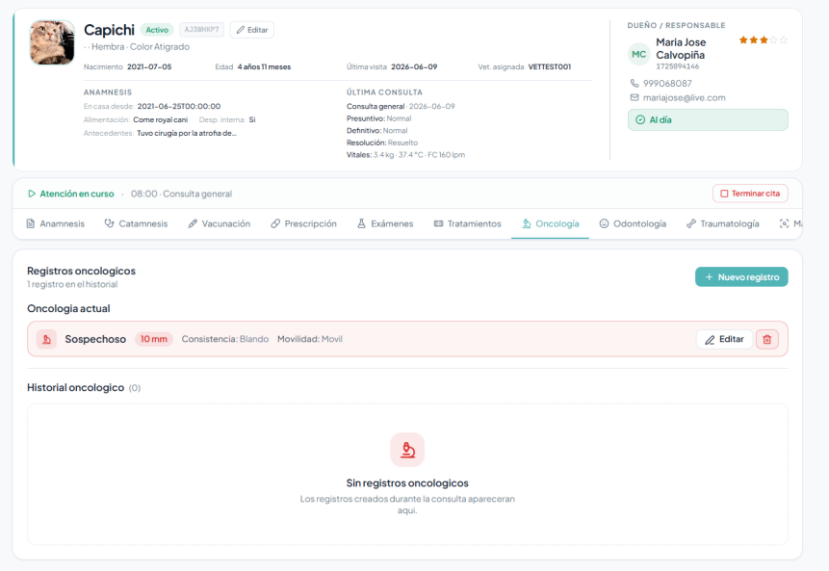
Nota. La captura corresponde al módulo descrito en la sección 4.7.4.2. Elaborado por el autor Calvopiña Daniel.

c) Módulos de especialidad médica

Los módulos de especialidad médica se incluyeron para digitalizar formularios específicos que el consultorio utiliza en casos particulares, entre estos se encuentran oncología, traumatología, odontología y mamas. Estos registros no reemplazan la catamnesis general, sino que la complementan cuando el caso clínico requiere mayor detalle o se encuentra un problema en concreto que puede llegar a tener el paciente y se tienen que registrar y controlar de una manera más fácil y sencilla para posteriormente si el problema llega a evolucionar saber cuál fue la condición inicial del problema, las funcionalidades anteriormente mencionas y especialidades se encuentran en la tabla 25 y las figuras 14,15, 16 y 17 la visualización de sus respectivas pantallas

Figura 14

Módulos de especialidad Oncológica



Nota. La captura corresponde a los módulos Oncológico descrito en la sección 4.7.4.3. Elaborado por el autor Calvopiña Daniel

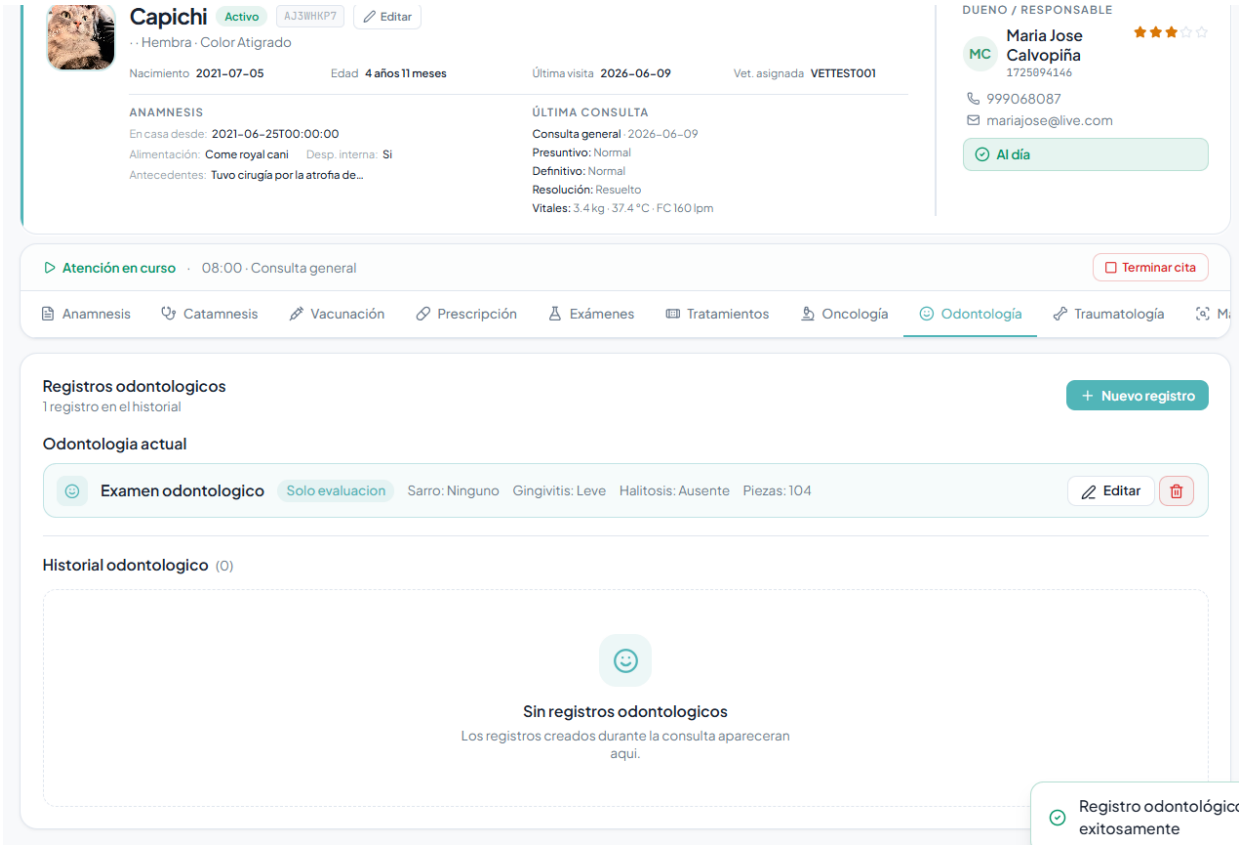
Tabla 18. Desarrollo y validación de especialidades clínicas

Especialidad	Uso dentro del sistema
Oncología	Registrar información de pacientes con diagnósticos neoplásicos o seguimiento oncológico.
Traumatología	Registrar lesiones, hallazgos físicos o imágenes relacionadas con traumatismos.
Odontología	Registrar hallazgos de dentición, piezas afectadas o procedimientos bucales.
Mamas	Registrar hallazgos en glándulas mamarias según ubicación y características.

Nota. Listado de especialidades y uso dentro del sistema. Elaborado por el autor Calvopiña Daniel.

Figura 15

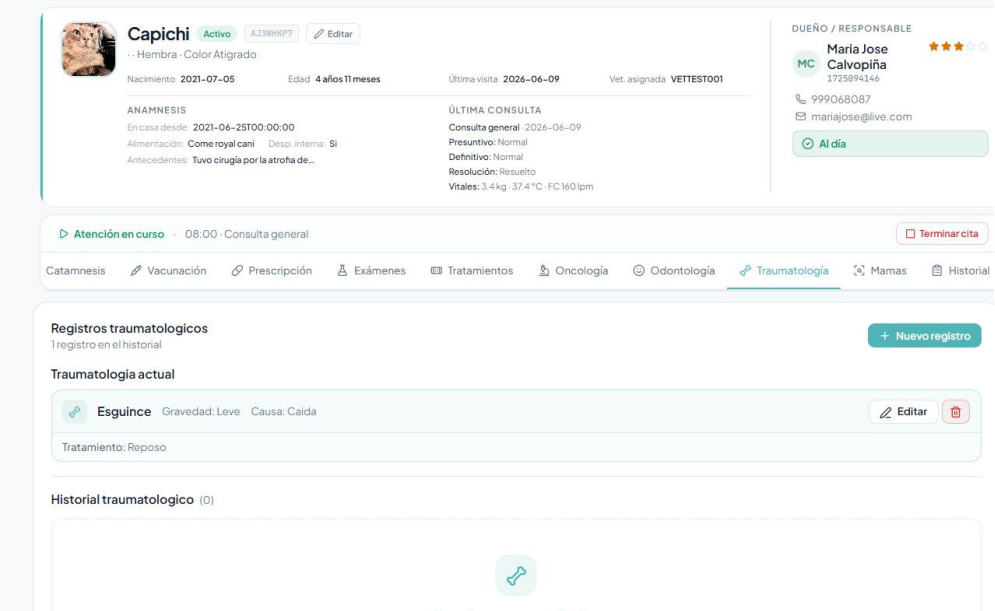
Módulos de especialidad Odontológica



Nota. La captura corresponde a los módulos de Odontológico en la sección 4.7.4.3. Elaborado por el autor Calvopiña Daniel.

Figura 16

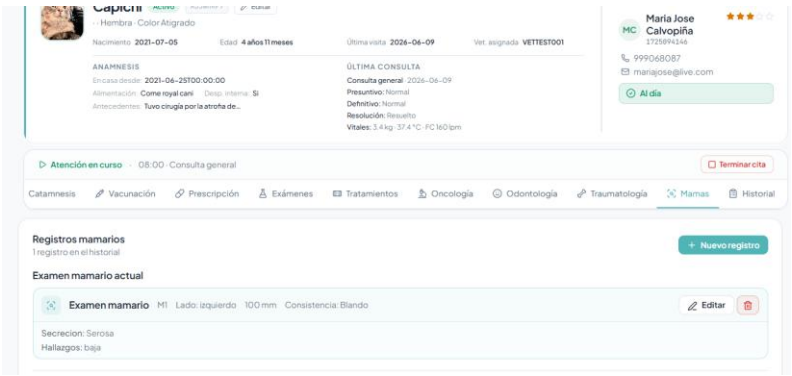
Módulos de especialidad Traumatología



Nota. La captura corresponde a los módulos de especialidad descritos en la sección 4.7.4.3. Elaborado por el autor Calvopiña Daniel.

Figura 17

Módulos de especialidad Mamas



Nota. La captura corresponde a los módulos de especialidad descritos en la sección 4.7.4.3. Elaborado por el autor Calvopiña Daniel.

4.7.5 Integración de procedimientos almacenados y triggers en el desarrollo

Durante el desarrollo se utilizaron procedimientos almacenados y triggers para reforzar reglas críticas. En lugar de depender únicamente de validaciones del frontend, ciertas reglas se ubicaron en la base de datos para garantizar que se cumplan incluso si la operación se ejecuta desde otro componente del sistema, por ejemplo, la generación de calendario y la sincronización con citas se apoyan en funciones y triggers para mantener la disponibilidad actualizada. De igual manera, los triggers de identificadores permiten que registros como pacientes, citas, anamnesis, catamnesis, vacunación, tratamientos, prescripciones y exámenes mantengan códigos ordenados. Las eliminaciones seguras evitan borrar catálogos o registros clínicos que ya se encuentran relacionados con otros datos en la tabla 26 tenemos algunos de los SPs implementados dentro de la base de datos para controlar distintos factores de duplicidad y aplicación de lógica.

Tabla 19. Validación funcional de reglas implementadas en base de datos

Regla	Mecanismo	Validación funcional
Generación automática de ID de paciente	trg_set_paciente_id	Al crear un paciente sin ID manual, el sistema asigna un código.
Sincronización de cita y calendario	trg_sync_calendario_estado	Al crear o actualizar cita, el bloque de calendario refleja el cambio.
Prevención de citas duplicadas	trg_prevent_duplicate_active_cita	No se permite registrar dos citas activas en el mismo horario.
Eliminación segura de catálogos	sp_delete_*_seguro	No se permite borrar elementos usados por pacientes, citas o registros clínicos.
Eliminación controlada de catamnesis	sp_delete_catamnesis_clinica	La eliminación considera dependencias clínicas y veterinario asociado.

Nota. Lista de reglas de SPs que están implementadas dentro de la base de datos Elaborado por el autor Calvopiña Daniel.

4.7.6 Cierre del desarrollo del prototipo

Como resultado del desarrollo, el prototipo integra la operación interna del consultorio con un portal público y de clientes. El sistema permite administrar datos institucionales, usuarios, clientes, pacientes, agenda, citas, catálogos, historial clínico, prescripciones,

exámenes, tratamientos y especialidades. Además, incorpora reglas de base de datos que refuerzan la integridad de la información y reducen problemas como citas duplicadas, registros sin identificador o eliminación incorrecta de datos relacionados.

El desarrollo también respondió al problema de visibilidad del consultorio mediante la landing pública, la cual permite mostrar servicios, veterinarios, anuncios y datos de contacto. Con ello, el prototipo no solo mejora el registro clínico y operativo, sino que también ofrece un medio para que nuevos propietarios conozcan el consultorio y puedan iniciar el proceso de atención de manera más ordenada.

En conjunto, el Capítulo 4 demuestra la relación entre el levantamiento de información, los requerimientos, el Product Backlog, la planificación de sprints, el modelado de procesos, el modelado de datos, la arquitectura y el desarrollo funcional. Esta secuencia permite evidenciar que el prototipo fue construido a partir de necesidades reales del consultorio y no únicamente desde una propuesta técnica aislada.

4.8 Arquitectura del sistema

La arquitectura del prototipo se organizó en cuatro proyectos principales: dos backends y dos frontends. Esta separación permite distinguir entre los procesos internos del consultorio y los procesos externos orientados a propietarios o visitantes de la página web. Además, permite que la lógica clínica y administrativa se mantenga separada de la presentación visual y de la experiencia del cliente.

El backend interno atiende los procesos clínicos y administrativos del consultorio. El backend del portal atiende la información pública, el registro de clientes y las citas solicitadas desde el exterior. El frontend interno permite al personal trabajar con clientes, pacientes, agenda, catálogos, parámetros e historial clínico. El frontend público muestra la landing del

consultorio y permite que el propietario interactúe con su perfil, mascotas y citas, además de tener 2 base de datos una encarga de la gestión de la información del sistema veterinaria mientras que la segunda se encarga de la información del portal público a clientes.

4.8.1 Arquitectura lógica

La arquitectura lógica separa las responsabilidades en capas. La capa de presentación está formada por los frontends desarrollados en React y TypeScript. La capa de servicios está compuesta por las APIs en ASP.NET Core. La capa de datos se apoya en PostgreSQL, donde se almacenan las entidades clínicas y operativas, además de esta estar distruida en 2 partes la primera base de datos encarga de la gestión de la información del sistema veterinaria mientras que la segunda se encarga de la información del portal público a clientes se opto por la distribución en este caso por el manejo de datos sensibles dentro del sistema veterinario que si optamos por tener una sola base de datos la información de la base de datos del sistema estaría predispuesta a sufrir ataques, evitando esto gracias a la distribución y colocando en la base de datos del portal únicamente la información necesaria para su funcionamiento y distribuyendo la información importante como es la de citas , logos, etc. Finalmente, el almacenamiento de archivos permite guardar imágenes de mascotas, logos, documentos o imágenes usadas por la página pública, en la tabla 27 se puede evidenciar los distintos componentes de tecnología y sus responsabilidades dentro del sistema.

Tabla 20. Componentes de la arquitectura lógica

Componente	Tecnología	Responsabilidad
Frontend interno	React, TypeScript, Vite	Permite al personal interno gestionar clientes, pacientes, agenda, catálogos, parámetros e historial clínico.
Frontend landing / portal	React, TypeScript, Vite	Muestra información pública del consultorio y permite registro, login, mascotas y citas de clientes.
Backend sistema veterinario	ASP.NET Core	Expone API interna para procesos clínicos y administrativos.
Backend portal veterinario	ASP.NET Core	Expone API pública y API autenticada para clientes.
Base de datos	PostgreSQL	Almacena datos del consultorio, usuarios, clientes, pacientes, agenda, citas e historial clínico.
Acceso a datos	Linq2DB / Npgsql	Permite consultar y modificar la base de datos desde los servicios de backend.
Almacenamiento de archivos	S3 compatible / bucket	Guarda imágenes, logos y archivos asociados al sistema.
Seguridad	JWT, roles y usuarios de base de datos	Controla sesiones internas y externas, diferenciando clientes de usuarios veterinarios.

Nota. Documentación técnica del proyecto. Elaborada por el Autor Calvopiña Daniel.

4.8.2 Arquitectura física

Tabla 21. Organización física de los proyectos

Proyecto	Tipo	Usuarios relacionados	Función principal
backend_sistema_veterinario	Backend API	Veterinarios, usuarios internos	Gestiona autenticación interna, pacientes, clientes, citas, agenda, historia clínica, catálogos, archivos y seguridad.
backend_portal_veterinario	Backend API	Clientes y visitantes	Gestiona landing pública, registro/login de cliente, mascotas, catálogos, disponibilidad y citas externas.
front_sistema_veterinario	Frontend interno	Usuarios internos	Presenta pantallas administrativas y clínicas del consultorio.
front_landing_veterinario	Frontend público/cliente	Visitantes y propietarios	Presenta información del consultorio y permite interacción del cliente.
PostgreSQL	Base de datos	Sistema completo	Guarda la información central del prototipo.
Bucket S3 compatible	Almacenamiento externo	Sistema completo	Almacena imágenes y archivos vinculados a registros.

Nota. Lista de Organización física del prototipo. Elaborado por el autor Calvopiña Daniel.

A nivel físico, el sistema se ejecuta como una solución web distribuida. Cada frontend se despliega como una aplicación independiente que consume una API mediante HTTP. Los backends se comunican con la base de datos PostgreSQL y con el servicio de almacenamiento de archivos. Esta estructura permite que el portal público pueda crecer sin mezclarse directamente con la operación interna del consultorio, en a tabla 13 se explica de una mejor manera la organización física de los proyectos colocando el tipo su relación con los usuarios y su función principal.

5 CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

a) El proyecto culminó exitosamente con el desarrollo de un prototipo funcional que cubre todo el ciclo de atención veterinaria. Esta solución, construida con dos backends en ASP.NET Core, dos frontends en React con TypeScript y dos bases de datos PostgreSQL, va mucho más allá de la simple digitalización de formularios. Logró integrar la administración del consultorio con la gestión médica del paciente, resolviendo de raíz los problemas iniciales de fragmentación del historial clínico, la pérdida de trazabilidad y la dependencia de registros manuales, garantizando un flujo de trabajo centralizado desde la solicitud de la cita hasta el control preventivo.

b) La combinación de entrevistas, observación directa de las jornadas de atención y el análisis de los documentos físicos permitió contrastar lo que se relataba con la práctica diaria. Gracias a este enfoque inmersivo, se identificaron cinco problemas centrales como la gestión informal de citas, el manejo ineficiente de la información clínica, la falta de seguimiento preventivo, la dispersión de papeles y la escasa presencia digital. Con base en estos hallazgos, se definieron requerimientos reales y precisos, evitando construir el sistema sobre simples suposiciones.

c) La organización de la arquitectura del sistema en cuatro proyectos separados y dividir la información en dos bases de datos resultó ser muy acertada. Esta estructura aísla los datos clínicos sensibles de la parte pública orientada a los clientes, reduciendo significativamente los riesgos de seguridad ante posibles vulnerabilidades. Además, el modelo de datos diseñado demostró ser robusto, ya que soportó el desarrollo de todos los módulos sin requerir cambios estructurales profundos. Esto confirma que el

levantamiento de información previo capturó de manera fiel las entidades del dominio veterinario, sus relaciones y reglas de negocio, las cuales quedaron bien blindadas mediante procedimientos almacenados y triggers.

d) El trabajo bajo la metodología Scrum con iteraciones organizadas en cuatro sprints permitió construir el sistema de forma ordenada y progresiva. Se comenzó sentando las bases del modelado de datos y la seguridad, para luego avanzar hacia la configuración del consultorio, la gestión de citas y el portal público, finalizando con el núcleo del sistema que es el historial clínico completo. Este enfoque iterativo facilitó la entrega de funcionalidades operativas al cierre de cada ciclo y permitió ajustar las prioridades del proyecto de manera flexible, demostrando ser la estrategia ideal para un desarrollo llevado a cabo por un solo programador con tiempos definidos.

e) La integración de la validación funcional a la par del desarrollo fue una estrategia clave para asegurar la calidad final. Cada módulo se probó a medida que se construía, culminando con validaciones exhaustivas sobre el flujo completo de la atención médica. Estas pruebas confirmaron el correcto funcionamiento de los registros, la prevención de choques de horarios en la agenda y la efectividad de las medidas de seguridad a nivel de base de datos y de roles de usuario. Se comprobó en la práctica que el sistema conserva y recupera la información de manera íntegra, eliminando definitivamente la duplicidad de fichas y la pérdida de seguimientos.

5.2 Recomendaciones

a) Se recomienda que, antes de llevar el sistema a producción, el levantamiento de información se amplíe a más de un consultorio veterinario, ya que este proyecto se trabajó sobre la realidad de un solo establecimiento, lo cual fue suficiente para el prototipo, pero validar los requerimientos en consultorios de distinto tamaño y especialidad permitiría confirmar qué funcionalidades aplican de forma general y cuáles responden únicamente al caso estudiado, además se recomienda mantener la combinación de entrevista, observación directa y análisis documental en cualquier ampliación futura, ya que durante este trabajo demostró ser muy útil para detectar necesidades que el usuario no menciona por considerarlas parte natural de su rutina.

b) Se recomienda conservar la separación entre la base de datos interna y la del portal público incluso si el sistema crece, ya que este aislamiento de la información clínica sensible es una medida de seguridad estructural y no solo una decisión de implementación, además de que para una operación real convendría complementar esta arquitectura con respaldos automáticos, cifrado de los datos en reposo y la posibilidad de distribuir la base de datos para garantizar la disponibilidad de la información ante un fallo del servidor principal.

c) Se recomienda dar continuidad al desarrollo incorporando las funcionalidades que quedaron fuera del alcance del prototipo y que el levantamiento de información mostró como valiosas para el consultorio, entre ellas el envío automático de recordatorios de vacunación, desparasitación y citas a los propietarios mediante correo o mensajería, ya que la falta de avisos oportunos fue uno de los problemas que más se repitió durante las entrevistas, además de la generación de reportes e indicadores que apoyen la toma de decisiones del consultorio y un módulo de facturación que complete el ciclo

administrativo de la atención, para estas ampliaciones se recomienda mantener el enfoque iterativo con Scrum, definiendo incrementos pequeños y verificables, ya que esta forma de trabajo resultó manejable y efectiva durante el proyecto.

d) Se recomienda que, como paso previo a un despliegue real, la validación funcional realizada en este proyecto se complemente con pruebas de aceptación ejecutadas directamente por el médico veterinario y su personal en condiciones reales de atención durante un período de tiempo, ya que esto permitiría evaluar la usabilidad del sistema bajo la presión del día a día y detectar ajustes de interfaz que las pruebas funcionales no llegan a revelar, de igual forma, se recomienda incorporar pruebas automatizadas sobre los servicios del backend, para que las futuras modificaciones puedan verificarse sin repetir manualmente todos los flujos, y realizar pruebas de seguridad específicas sobre el portal público, por ser la capa que queda expuesta a usuarios externos.
e) Finalmente, se recomienda este proyecto como base para futuros trabajos de titulación o de investigación aplicada, ya que existen varias líneas naturales de continuidad, como el desarrollo de una aplicación móvil para los propietarios que consuma los servicios del portal, el análisis de los datos del historial clínico acumulado para apoyar la medicina preventiva, o la generalización del sistema hacia un modelo multiconsultorio que permita atender varios establecimientos desde una misma plataforma. Cualquiera de estas extensiones puede apoyarse en la arquitectura, el modelo de datos y la documentación generados en este trabajo, lo que reduce bastante el esfuerzo inicial de análisis y diseño.

6 BIBLIOGRAFÍA

OWASP Foundation. (2021). OWASP Top 10 – 2021. https://owasp.org/Top10/

Pressman, R. S., & Maxim, B. R. (2019). Software engineering: A practitioner's approach (9th ed.). McGrawHill Education.

Radostits, O. M., Gay, C. C., Hinchcliff, K. W., & Constable, P. D. (2007). Veterinary medicine: A textbook of the diseases of cattle, horses, sheep, pigs and goats (10th ed.). Saunders Elsevier.

Shortliffe, E. H., & Cimino, J. J. (Eds.). (2014). Biomedical informatics: Computer applications in health care and biomedicine (4th ed.). Springer.

Weed, L. L. (1969). Medical records, medical education, and patient care: The problemoriented record as a basic tool. Case Western Reserve University Press.

WSAVA Vaccination Guidelines Group. (2015). WSAVA vaccination guidelines for the vaccination of dogs and cats [PDF]. https://wsava.org/wpcontent/uploads/2020/01/WSAVAVaccinationGuidelines2015.pdf

Ettinger, S. J., & Feldman, E. C. (Eds.). (2017). Textbook of veterinary internal medicine: Diseases of the dog and the cat (8th ed.). Elsevier.

Plumb, D. C. (2018). Plumb's veterinary drug handbook (9th ed.). WileyBlackwell.

Radostits, O. M., Gay, C. C., Hinchcliff, K. W., & Constable, P. D. (2007). Veterinary medicine: A textbook of the diseases of cattle, horses, sheep, pigs and goats (10th ed.). Saunders Elsevier.

WSAVA Vaccination Guidelines Group. (2015). WSAVA vaccination guidelines for the vaccination of dogs and cats [PDF]. https://wsava.org/wpcontent/uploads/2020/01/WSAVAVaccinationGuidelines2015.pdf

American Veterinary Medical Association Professional Liability Insurance Trust [AVMA PLIT]. (s. f.). Client management guidelines. Recuperado el 1 de enero de 2026, de https://www.avmaplit.com/educationcenter/library/clientmanagementguidelines/

GurolUrganci, I., de Jongh, T., VodopivecJamsek, V., Atun, R., & Car, J. (2013). Mobile phone messaging reminders for attendance at healthcare appointments. Cochrane Database of Systematic Reviews, 12, CD007458. https://doi.org/10.1002/14651858.CD007458.pub3

Schwaber, K., & Sutherland, J. (2020). The Scrum Guide. https://scrumguides.org/scrumguide.html

Shortliffe, E. H., & Cimino, J. J. (Eds.). (2014). Biomedical informatics: Computer applications in health care and biomedicine (4th ed.). Springer.

Sommerville, I. (2016). Software engineering (10th ed.). Pearson.

van der Aalst, W. M. P. (2016). Process mining: Data science in action (2nd ed.). Springer.

WSAVA Vaccination Guidelines Group. (2015). WSAVA vaccination guidelines for the vaccination of dogs and cats [PDF]. https://wsava.org/wpcontent/uploads/2020/01/WSAVAVaccinationGuidelines2015.pdf

American Veterinary Medical Association Professional Liability Insurance Trust [AVMA PLIT]. (s. f.). Client management guidelines. Recuperado el 1 de enero de 2026, de https://www.avmaplit.com/education-center/library/client-management-guidelines/

Schwaber, K., & Sutherland, J. (2020). The Scrum Guide. https://scrumguides.org/scrum-guide.html

Sommerville, I. (2016). Software engineering (10th ed.). Pearson.

van der Aalst, W. M. P. (2016). Process mining: Data science in action (2nd ed.). Springer.

WSAVA Vaccination Guidelines Group. (2015). WSAVA vaccination guidelines for the vaccination of dogs and cats [PDF]. https://wsava.org/wp-content/uploads/2020/01/WSAVA-Vaccination-Guidelines-2015.pdf

Bass, L., Clements, P., & Kazman, R. (2012). Software architecture in practice (3rd ed.). Addison-Wesley.

Codd, E. F. (1970). A relational model of data for large shared data banks. Communications of the ACM, 13(6), 377–387. https://doi.org/10.1145/362384.362685

Fette, I., & Melnikov, A. (2011). The WebSocket protocol (RFC 6455). Internet Engineering Task Force. https://www.rfc-editor.org/rfc/rfc6455

Fielding, R. T. (2000). Architectural styles and the design of network-based software architectures (Doctoral dissertation, University of California, Irvine). https://www.ics.uci.edu/~fielding/pubs/dissertation/fielding_dissertation.pdf

Fowler, M. (2002). Patterns of enterprise application architecture. Addison-Wesley.

Garrett, J. J. (2011). The elements of user experience: User-centered design for the web and beyond (2nd ed.). New Riders.

Pressman, R. S., & Maxim, B. R. (2019). Software engineering: A practitioner's approach (9th ed.). McGraw-Hill Education.

Schwaber, K., & Sutherland, J. (2020). The Scrum Guide. https://scrumguides.org/scrum-guide.html

Silberschatz, A., Korth, H. F., & Sudarshan, S. (2019). Database system concepts (7th ed.). McGraw-Hill Education.

Tanenbaum, A. S., & Bos, H. (2015). Modern operating systems (4th ed.). Pearson.

Microsoft. (s. f.). ASP.NET documentation. Recuperado el 1 de enero de 2026, de https://learn.microsoft.com/en-us/aspnet/core/

Microsoft. (s. f.). C# documentation. Recuperado el 1 de enero de 2026, de https://learn.microsoft.com/en-us/dotnet/csharp/

Microsoft. (s. f.). TypeScript documentation. Recuperado el 1 de enero de 2026, de https://www.typescriptlang.org/docs/

PostgreSQL Global Development Group. (s. f.). PostgreSQL documentation. Recuperado el 1 de enero de 2026, de https://www.postgresql.org/docs/

React. (s. f.). React documentation. Recuperado el 1 de enero de 2026, de https://react.dev/

Tailwind Labs. (s. f.). Tailwind CSS documentation. Recuperado el 1 de enero de 2026, de https://tailwindcss.com/docs/

Hernández Sampieri, R., Fernández Collado, C., & Baptista Lucio, P. (2014). *Metodología de la investigación* (6.ª ed.). McGraw-Hill Education.

Pressman, R. S., & Maxim, B. R. (2019). *Software engineering: A practitioner's approach* (9th ed.). McGraw-Hill Education.

Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide: The definitive guide to Scrum: The rules of the game*. https://scrumguides.org/scrum-guide.html

Sommerville, I. (2016). *Software engineering* (10th ed.). Pearson.

Hernández Sampieri, R., Fernández Collado, C., & Baptista Lucio, P. (2014). Metodología de la investigación (6.a ed.). McGraw-Hill Education.

Microsoft. (s. f.). ASP.NET documentation. Recuperado el 1 de enero de 2026, de https://learn.microsoft.com/en-us/aspnet/core/

Microsoft. (s. f.). TypeScript documentation. Recuperado el 1 de enero de 2026, de https://www.typescriptlang.org/docs/

PostgreSQL Global Development Group. (s. f.). PostgreSQL documentation. Recuperado el 1 de enero de 2026, de https://www.postgresql.org/docs/

Pressman, R. S., & Maxim, B. R. (2019). Software engineering: A practitioner's approach (9th ed.). McGraw-Hill Education.

React. (s. f.). React documentation. Recuperado el 1 de enero de 2026, de https://react.dev/

Schwaber, K., & Sutherland, J. (2020). The Scrum Guide. https://scrumguides.org/scrum-guide.html

Sommerville, I. (2016). Software engineering (10th ed.). Pearson.

Google. (s. f.). *Angular: The modern web developer's platform*. Recuperado de https://angular.io/

Ilin, I., et al. (s. f.). *linq2db - LINQ to Database*. Recuperado de https://linq2db.github.io/

Microsoft. (s. f.). *ASP.NET overview: Web framework for building great Web sites and Web applications*. Recuperado de https://dotnet.microsoft.com/apps/aspnet

Microsoft. (s. f.). *Entity Framework Core*. Recuperado de https://docs.microsoft.com/en-us/ef/core/

OpenJS Foundation. (s. f.). *Node.js*. Recuperado de https://nodejs.org/

Oracle. (s. f.). *MySQL*. Recuperado de https://www.mysql.com/

PostgreSQL Global Development Group. (s. f.). *About PostgreSQL*. Recuperado de https://www.postgresql.org/about/

React. (s. f.). *React – A JavaScript library for building user interfaces*. Recuperado de https://react.dev/

Hernández Sampieri, R., Fernández Collado, C., & Baptista Lucio, P. (2014). Metodología de la investigación (6.a ed.). McGraw-Hill Education.

Pressman, R. S., & Maxim, B. R. (2019). Software engineering: A practitioner's approach (9th ed.). McGraw-Hill Education.

Schwaber, K., & Sutherland, J. (2020). The Scrum Guide. https://scrumguides.org/scrum-guide.html

Sommerville, I. (2016). Software engineering (10th ed.). Pearson.

WSAVA Vaccination Guidelines Group. (2015). WSAVA vaccination guidelines for the vaccination of dogs and cats [PDF]. https://wsava.org/wp-content/uploads/2020/01/WSAVA-Vaccination-Guidelines-2015.pdf

React. (s. f.). React documentation. Recuperado el 1 de enero de 2026, de https://react.dev/

PostgreSQL Global Development Group. (s. f.). PostgreSQL documentation. Recuperado el 1 de enero de 2026, de https://www.postgresql.org/docs/

Microsoft. (s. f.). TypeScript documentation. Recuperado el 1 de enero de 2026, de https://www.typescriptlang.org/docs/

Microsoft. (s. f.). ASP.NET documentation. Recuperado el 1 de enero de 2026, de https://learn.microsoft.com/en-us/aspnet/core/

Fowler, M. (2002). Patterns of enterprise application architecture. Addison-Wesley.

Codd, E. F. (1970). A relational model of data for large shared data banks. Communications of the ACM, 13(6), 377–387. https://doi.org/10.1145/362384.362685

8 ANEXOS

8.1 Anexo 1. Guía de entrevista semiestructurada

Datos generales

Entrevistado: _____

Lugar: _____

Duración aproximada: 1 hora y media Modalidad: Presencial

Bloque 1: Rutina diaria y organización

1. ¿Cómo describiría una jornada de atención típica en el consultorio?

2. ¿Cuántas consultas atiende en promedio por día?

3. ¿Cuáles son los momentos del día que más tiempo le demandan fuera de la consulta en sí?

4. ¿Cuenta con personal de apoyo en el consultorio? ¿Cuál es su rol?

Bloque 2: Gestión de citas

5. ¿Cómo reciben y registran actualmente las solicitudes de cita?

6. ¿Qué ocurre cuando llega una solicitud fuera del horario de atención?

7. ¿Con qué frecuencia se producen conflictos de horario o citas cruzadas?

8. ¿Llevan algún registro histórico de las citas de cada paciente?

Bloque 3: Registro clínico

9. ¿Cómo se crea y archiva la ficha de un paciente nuevo?

10. ¿Qué tan frecuente es no encontrar la ficha de un paciente cuando se necesita?

11. ¿Ha tenido casos de fichas duplicadas para el mismo animal?

12. ¿Cómo maneja el seguimiento de tratamientos que requieren varias sesiones?

Bloque 4: Medicina preventiva

13. ¿Cómo controla los esquemas de vacunación y desparasitación de cada paciente?

14. ¿Tiene algún mecanismo para identificar pacientes con refuerzos vencidos?

15. ¿Es frecuente que los propietarios lleguen con el esquema preventivo atrasado?

16. ¿Qué ocurre cuando un propietario pierde el carnet de vacunación?

Bloque 5: Problemas frecuentes y necesidades

17. ¿Cuáles son los procesos que más problemas o errores le generan actualmente?

18. ¿Ha intentado antes implementar alguna solución digital? ¿Qué pasó?

19. ¿Qué funcionalidades consideraría indispensables en un sistema de gestión?

20. ¿Le interesaría que los propietarios pudieran solicitar citas sin llamarle directamente?

Observación: este anexo respalda los resultados presentados en la sección 4.1.1 y permite evidenciar cómo se obtuvo la información cualitativa usada para definir los requerimientos.

8.2 Anexo 2. Guía de observación directa

Datos generales

Lugar: _____

Modalidad: _____

Duración aproximada por visita: 1 hora 30 minutos a 2 horas

Aspectos para observar

1. Flujo completo de atención al paciente desde la llegada hasta la salida. Se registrará la secuencia de pasos, quién realiza cada uno y el tiempo aproximado de cada etapa.

2. Proceso de registro de una mascota nueva. Se observará cómo se asigna el número de ficha, cómo se recopilan los datos y qué información queda pendiente o incompleta.

3. Gestión de citas durante la jornada. Se registrará cómo se manejan las llegadas sin cita, los conflictos de horario y las solicitudes que llegan mientras el veterinario está en consulta.

4. Llenado de la ficha clínica durante la consulta. Se observará qué secciones completas el veterinario en cada momento y si el espacio disponible resulta suficiente.

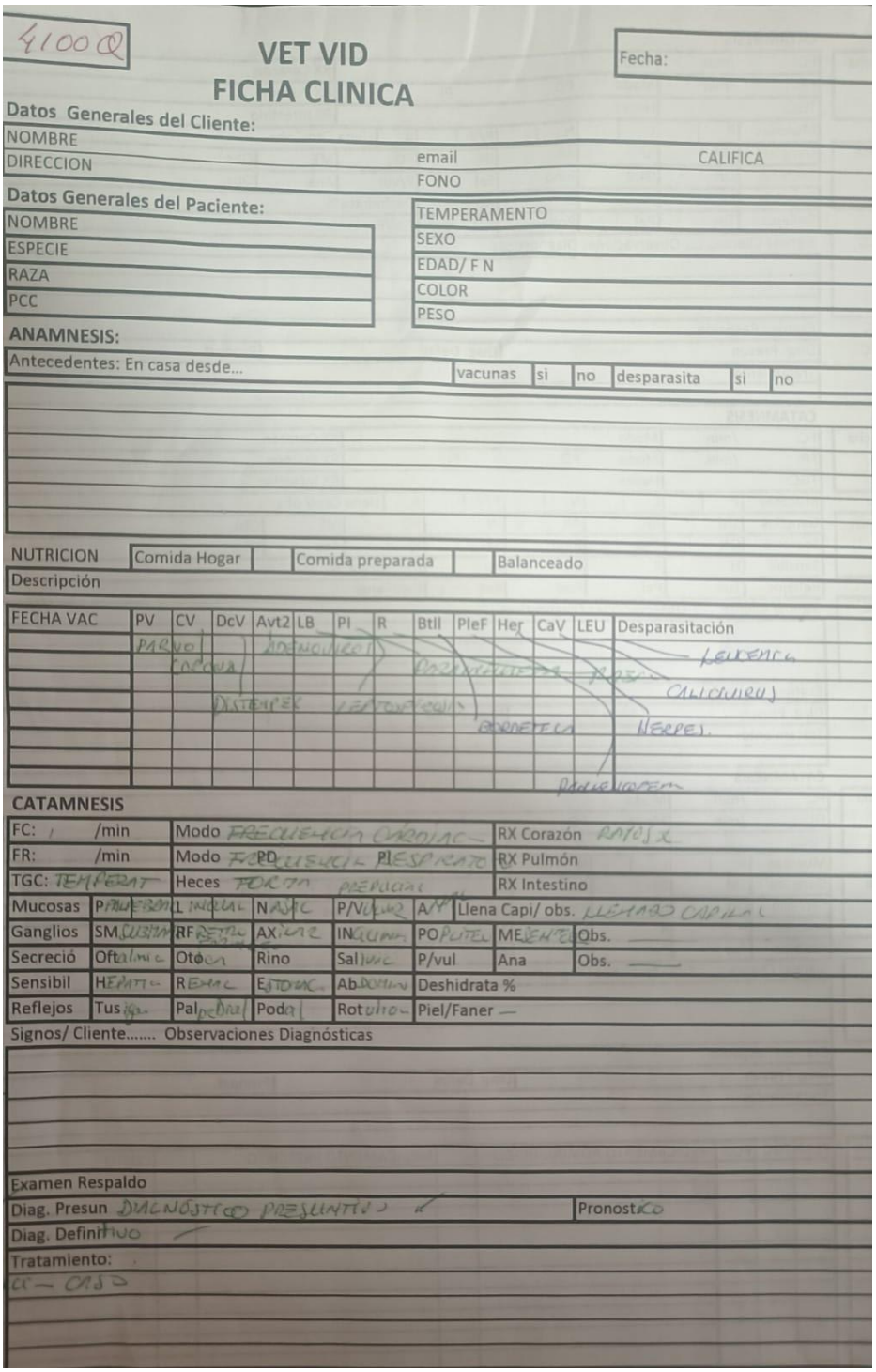
5. Control de vacunación y desparasitación. Se registrará si existe algún proceso activo de seguimiento o si el control ocurre únicamente cuando el propietario lo solicita.

6. Recuperación del historial clínico de un paciente recurrente. Se medirá el tiempo y el proceso necesario para acceder al historial de una consulta anterior.

7. Uso de formularios especializados. Se observará en qué situaciones el veterinario recurre a la hoja de ubicación anatómica y al formulario de catamnesis oncológica.

Observación: este anexo respalda la sección 4.1.2 y sirve como evidencia del proceso observado sin intervención del investigador.

8.3 Anexo 3. Ficha clínica general del consultorio



4100 Q

VET VID
FICHA CLINICA

Fecha:

Datos Generales del Cliente:

NOMBRE
DIRECCION — email — CALIFICA
FONO

Datos Generales del Paciente:

NOMBRE
ESPECIE
RAZA
PCC

TEMPERAMENTO
SEXO
EDAD/ F N
COLOR
PESO

ANAMNESIS:

Antecedentes: En casa desde... | vacunas | si | no | desparasita | si | no

NUTRICION | Comida Hogar | | Comida preparada | | Balanceado
Descripción

FECHA VAC	PV	CV	DcV	Avt2	LB	PI	R	Btll	PleF	Her	CaV	LEU	Desparasitación
	PARVO												
		CACHIA											
			DISTEMPER										

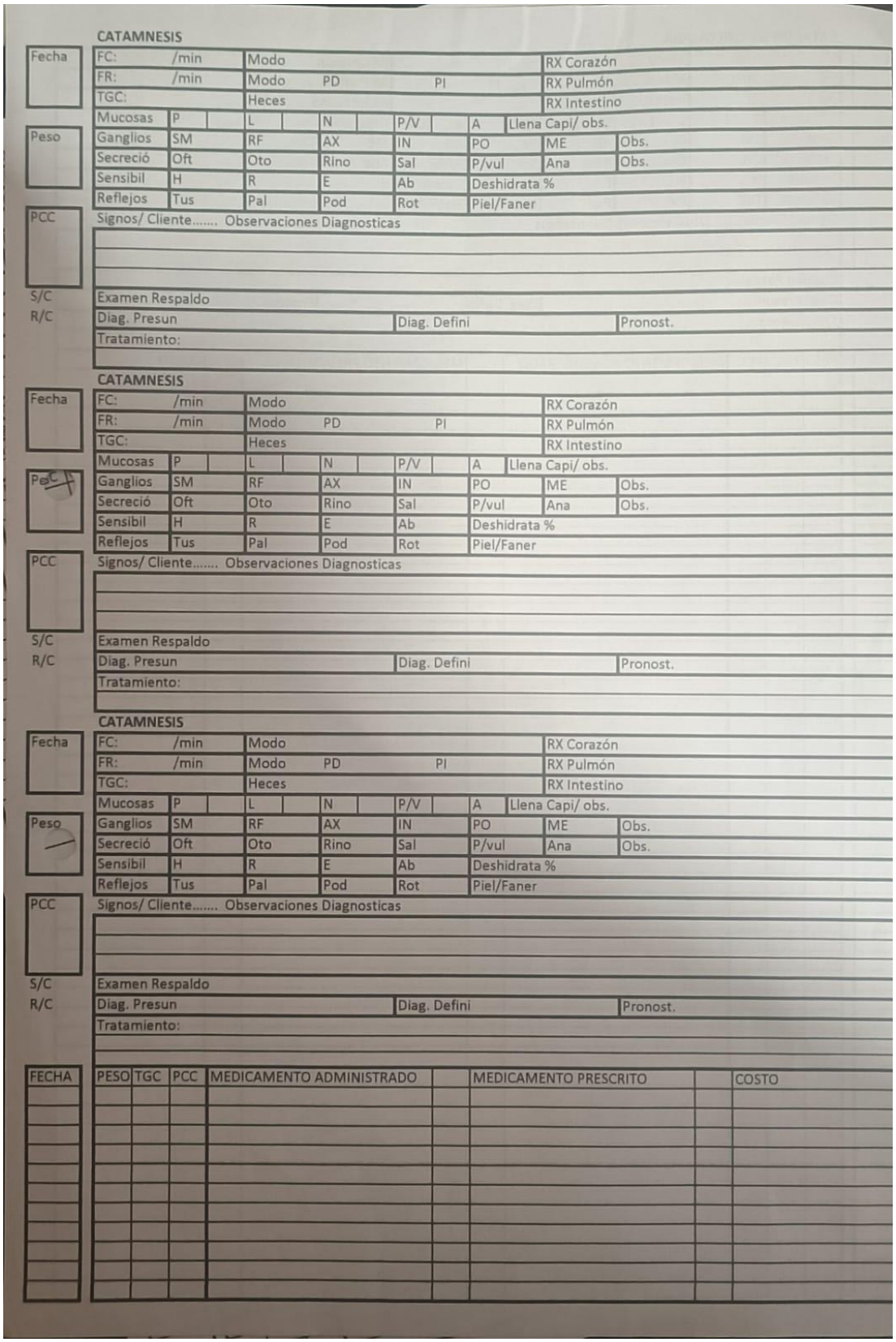
CATAMNESIS

FC: /min	Modo				RX Corazón		
FR: /min	Modo	PD		PI	RX Pulmón		
TGC:	Heces				RX Intestino		
Mucosas	P	L	NASC	P/V	AP	Llena Capi/ obs.	
Ganglios	SM	RF	AX	IN	PO	ME	Obs.
Secreció	Oft	Oto	Rino	Sal	P/vul	Ana	Obs.
Sensibil	H	R	E	Ab	Deshidrata %		
Reflejos	Tus	Pal	Pod	Rot	Piel/Faner		

Signos/ Cliente..... Observaciones Diagnósticas

Examen Respaldo
Diag. Presun — Pronostico
Diag. Definitivo
Tratamiento:

8.4 Anexo 4. Hoja de seguimiento de tratamientos



CATAMNESIS

Fecha	FC: /min	Modo					RX Corazón	
	FR: /min	Modo	PD		PI		RX Pulmón	
	TGC:	Heces					RX Intestino	
	Mucosas	P	L	N	P/V	A	Llena Capi/ obs.	
Peso	Ganglios	SM	RF	AX	IN	PO	ME	Obs.
	Secreció	Oft	Oto	Rino	Sal	P/vul	Ana	Obs.
	Sensibil	H	R	E	Ab	Deshidrata %		
	Reflejos	Tus	Pal	Pod	Rot	Piel/Faner		

PCC

Signos/ Cliente..... Observaciones Diagnosticas

S/C

Examen Respaldo

R/C

Diag. Presun	Diag. Defini	Pronost.

Tratamiento:

CATAMNESIS

Fecha	FC: /min	Modo					RX Corazón	
	FR: /min	Modo	PD		PI		RX Pulmón	
	TGC:	Heces					RX Intestino	
	Mucosas	P	L	N	P/V	A	Llena Capi/ obs.	
Peso	Ganglios	SM	RF	AX	IN	PO	ME	Obs.
	Secreció	Oft	Oto	Rino	Sal	P/vul	Ana	Obs.
	Sensibil	H	R	E	Ab	Deshidrata %		
	Reflejos	Tus	Pal	Pod	Rot	Piel/Faner		

PCC

Signos/ Cliente..... Observaciones Diagnosticas

S/C

Examen Respaldo

R/C

Diag. Presun	Diag. Defini	Pronost.

Tratamiento:

CATAMNESIS

Fecha	FC: /min	Modo					RX Corazón	
	FR: /min	Modo	PD		PI		RX Pulmón	
	TGC:	Heces					RX Intestino	
	Mucosas	P	L	N	P/V	A	Llena Capi/ obs.	
Peso	Ganglios	SM	RF	AX	IN	PO	ME	Obs.
	Secreció	Oft	Oto	Rino	Sal	P/vul	Ana	Obs.
	Sensibil	H	R	E	Ab	Deshidrata %		
	Reflejos	Tus	Pal	Pod	Rot	Piel/Faner		

PCC

Signos/ Cliente..... Observaciones Diagnosticas

S/C

Examen Respaldo

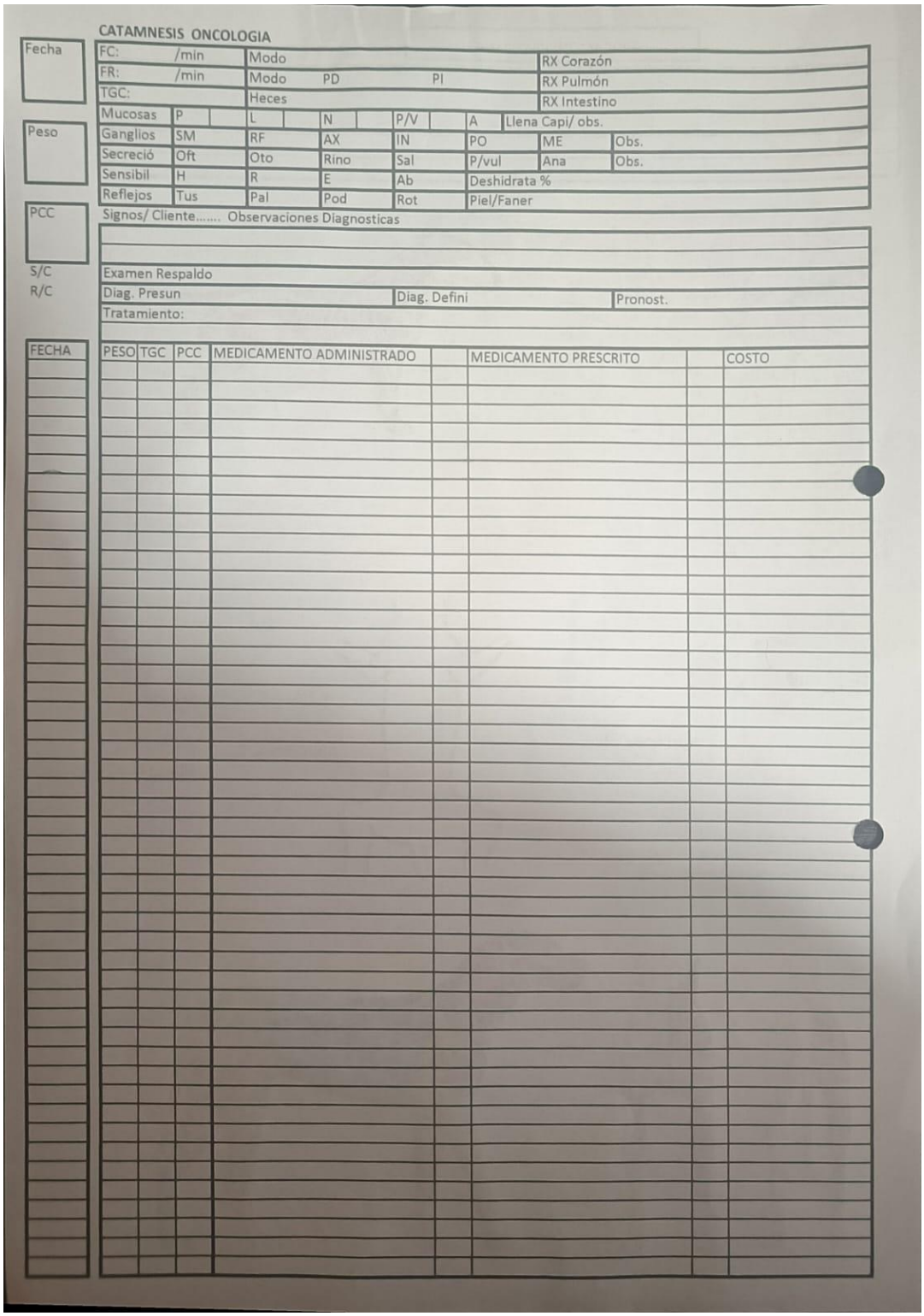
R/C

Diag. Presun	Diag. Defini	Pronost.

Tratamiento:

FECHA	PESO	TGC	PCC	MEDICAMENTO ADMINISTRADO		MEDICAMENTO PRESCRITO		COSTO

8.5 Anexo 5. Formulario de catamnesis oncológica



CATAMNESIS ONCOLOGIA

Fecha

Peso

PCC

S/C

R/C

FC: /min	Modo						RX Corazón	
FR: /min	Modo PD				PI		RX Pulmón	
TGC:	Heces						RX Intestino	
Mucosas	P	L	N	P/V	A	Llena Capi/ obs.		
Ganglios	SM	RF	AX	IN	PO	ME	Obs.	
Secreció	Oft	Oto	Rino	Sal	P/vul	Ana	Obs.	
Sensibil	H	R	E	Ab	Deshidrata %			
Reflejos	Tus	Pal	Pod	Rot	Piel/Faner			

Signos/ Cliente..... Observaciones Diagnosticas

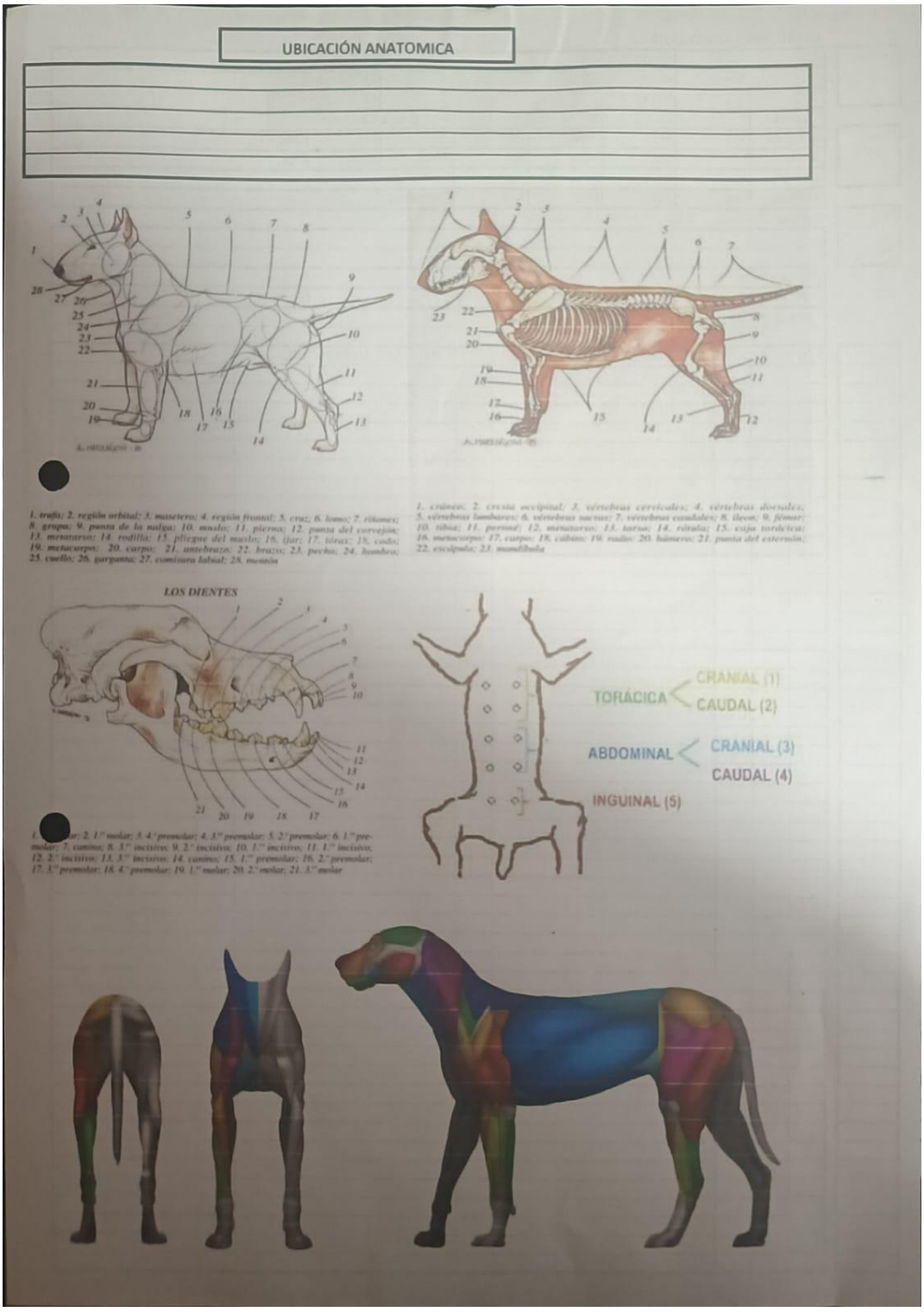
Examen Respaldo

Diag. Presun | Diag. Defini | Pronost.

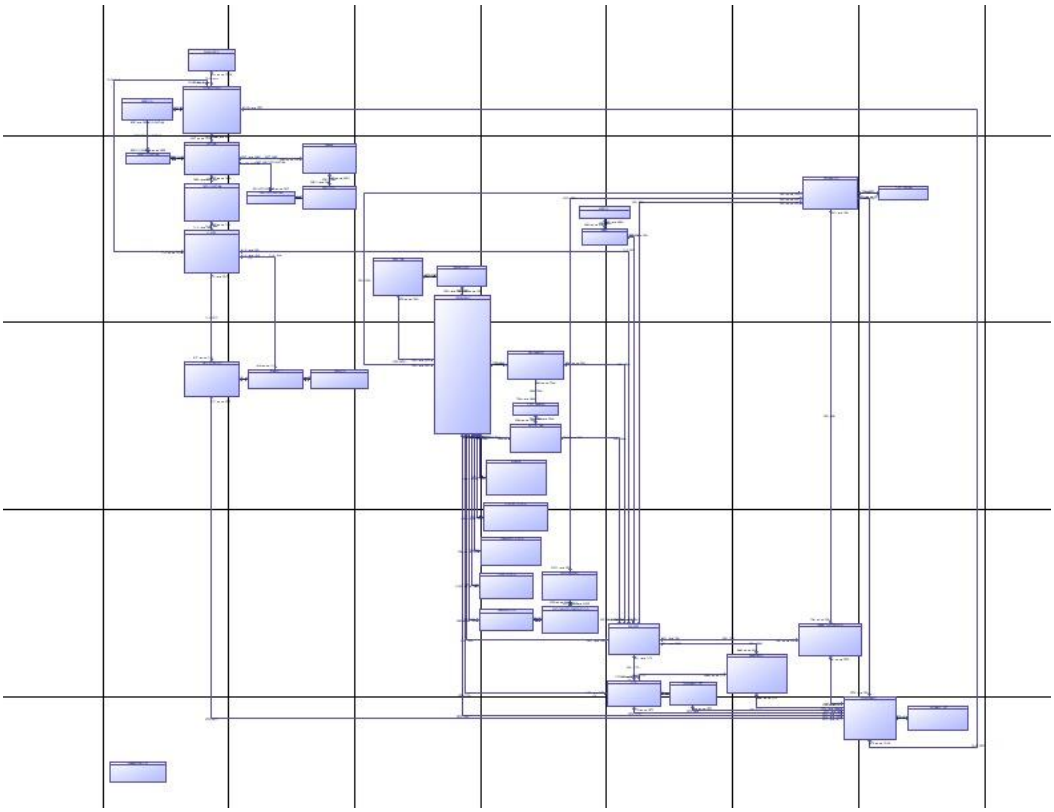
Tratamiento:

FECHA	PESO	TGC	PCC	MEDICAMENTO ADMINISTRADO		MEDICAMENTO PRESCRITO		COSTO

8.6 Anexo 6. Hoja de ubicación anatómica



8.7 Anexo 7. Modelo entidad-relación completo



8.8 Anexo 8. Historias de usuario del prototipo

8.8.1 Historia de usuario HU-01

Historia de Usuario	
Número: HU-01	**Usuario:** Médico veterinario
Nombre historia: Acceso interno y control de usuarios	
Prioridad en negocio: Alta	
Programador responsable: Daniel Calvopiña	
Descripción:	

- **Como:** médico veterinario que administra el consultorio.
- **Quiero/Necesito:** iniciar sesión y gestionar el acceso al sistema interno.
- **Para:** controlar quién puede acceder a la información clínica y administrativa del consultorio.

Observaciones: El usuario interno ingresa con credenciales válidas y el sistema restringe el acceso según corresponda.

8.8.2 Historia de usuario HU-02

Historia de Usuario	
Número: HU-02	**Usuario:** Médico veterinario
Nombre historia: Parametrización del consultorio	
Prioridad en negocio: Alta	
Programador responsable: Daniel Calvopiña	
Descripción:	

- **Como:** médico veterinario responsable del consultorio.
- **Quiero/Necesito:** registrar los datos del consultorio, servicios, horarios y parámetros de agenda.
- **Para:** que el sistema funcione con la configuración real del establecimiento.

Observaciones: El sistema guarda y actualiza la información del consultorio y la utiliza en la agenda y la página pública.

8.8.3 Historia de usuario HU-03

Historia de Usuario	
Número: HU-03	**Usuario:** Visitante público
Nombre historia: Consulta de página informativa	
Prioridad en negocio: Media alta	
Programador responsable: Daniel Calvopiña	
Descripción:	

- **Como:** visitante interesado en el consultorio.
- **Quiero/Necesito:** consultar una página pública con la información del consultorio.
- **Para:** conocer sus servicios, ubicación, horarios y medios de contacto antes de solicitar atención.

Observaciones: La landing muestra servicios, ubicación, horarios y contacto sin requerir autenticación.

8.8.4 Historia de usuario HU-04

Historia de Usuario	
Número: HU-04	**Usuario:** Cliente
Nombre historia: Registro e inicio de sesión en portal	
Prioridad en negocio: Media alta	
Programador responsable: Daniel Calvopiña	
Descripción:	

- **Como:** propietario de una mascota.
- **Quiero/Necesito:** registrarme e iniciar sesión desde el portal.
- **Para:** administrar mis datos, registrar mis mascotas y solicitar citas.

Observaciones: El cliente crea una cuenta, inicia sesión y accede a las funciones privadas del portal.

8.8.5 Historia de usuario HU-05

Historia de Usuario	
Número: HU-05	**Usuario:** Médico veterinario
Nombre historia: Registro de propietarios y pacientes	
Prioridad en negocio: Alta	
Programador responsable: Daniel Calvopiña	
Descripción:	

- **Como:** médico veterinario que atiende a los pacientes.
- **Quiero/Necesito:** registrar propietarios y mascotas.
- **Para:** mantener organizada la información básica y evitar la duplicidad de fichas físicas.

Observaciones: El sistema crea propietarios, asocia mascotas y permite consultar la relación cliente-paciente.

8.8.6 Historia de usuario HU-06

Historia de Usuario	
Número: HU-06	**Usuario:** Médico veterinario
Nombre historia: Agendamiento de citas	
Prioridad en negocio: Alta	
Programador responsable: Daniel Calvopiña	
Descripción:	

- **Como:** médico veterinario que organiza su jornada.
- **Quiero/Necesito:** consultar disponibilidad y agendar citas.
- **Para:** evitar cruces de horario y mantener una agenda centralizada.

Observaciones: El sistema valida horarios disponibles, registra citas y permite cambiar su estado.

8.8.7 Historia de usuario HU-07

Historia de Usuario	
Número: HU-07	**Usuario:** Médico veterinario
Nombre historia: Consulta del historial clínico	
Prioridad en negocio: Alta	
Programador responsable: Daniel Calvopiña	
Descripción:	

- **Como:** médico veterinario que atiende al paciente.
- **Quiero/Necesito:** consultar el historial clínico del paciente.
- **Para:** revisar antecedentes, consultas, tratamientos y controles anteriores.

Observaciones: El perfil del paciente muestra los registros clínicos asociados de forma ordenada y consultable.

8.8.8 Historia de usuario HU-08

Historia de Usuario	
Número: HU-08	**Usuario:** Médico veterinario
Nombre historia: Registro de consulta médica	
Prioridad en negocio: Alta	
Programador responsable: Daniel Calvopiña	
Descripción:	

- **Como:** médico veterinario durante la atención.
- **Quiero/Necesito:** registrar anamnesis, catamnesis, diagnóstico y pronóstico.
- **Para:** documentar correctamente la atención del paciente.

Observaciones: La consulta se registra y se asocia al paciente, la cita y el veterinario correspondiente.

8.8.9 Historia de usuario HU-09

Historia de Usuario	
Número: HU-09	**Usuario:** Médico veterinario
Nombre historia: Tratamientos, prescripciones y exámenes	
Prioridad en negocio: Media alta	
Programador responsable: Daniel Calvopiña	
Descripción:	

- **Como:** médico veterinario que da seguimiento al paciente.
- **Quiero/Necesito:** registrar tratamientos, prescripciones, pedidos y resultados de exámenes.
- **Para:** conservar la trazabilidad clínica del paciente.

Observaciones: Los tratamientos, prescripciones y exámenes quedan vinculados al historial clínico del paciente.