



PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR SEDE SANTO DOMINGO

Escuela de Hábitat, Infraestructura y Creatividad

APLICACIÓN WEB CON MACHINE LEARNING PARA LA GESTIÓN DE HISTORIAL
MÉDICO DE MASCOTAS EN EL HOSPITAL VETERINARIO SANTA MARTHA DEL
CANTÓN SANTO DOMINGO

TRABAJO DE TITULACIÓN

Previo a la obtención del título de Ingeniero en Tecnologías de la Información

Línea de investigación: Tecnologías de la información y la comunicación

Autoría:

Herrera Chancay Jordan Stalin

Paredes Romero Jenniffer Tahis

Dirección:

Ocampo Pazos Willian Javier, Mg.

Santo Domingo – Ecuador
Marzo, 2026



PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR SEDE SANTO DOMINGO

Escuela de Hábitat, Creatividad e Ingeniería

HOJA DE APROBACIÓN

APLICACIÓN WEB CON MACHINE LEARNING PARA LA GESTIÓN DE HISTORIAL
MÉDICO DE MASCOTAS EN EL HOSPITAL VETERINARIO SANTA MARTHA DEL
CANTÓN SANTO DOMINGO

Línea de investigación: Tecnologías de la información y la comunicación

Autoría:

Herrera Chancay Jordan Stalin

Paredes Romero Jenniffer Tahis

Revisado por:

Ocampo Pazos Willian Javier, Mg.
DIRECTOR DEL TRABAJO DE
INTEGRACIÓN CURRICULAR

Ulloa Meneses Luis Javier, Mg.
CALIFICADOR

Córdova Gálvez Rodolfo Sirilo, Mg.
CALIFICADOR

Ulloa Meneses Luis Javier, Mg.
COORDINADOR DE LA CARRERA DE
SISTEMAS DE LA INFORMACIÓN

Santo Domingo – Ecuador
Marzo, 2026

DECLARACIÓN DE AUTENTICIDAD Y RESPONSABILIDAD

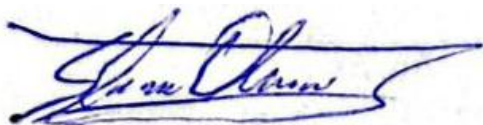
Nosotros, Herrera Chancay Jordan Stalin, portador de la cédula de ciudadanía 1722153887, y Paredes Romero Jenniffer Tahis, portadora de la cédula de ciudadanía 1316463387, declaramos que los resultados obtenidos en la investigación que presentamos como informe final, previo a la obtención del Título de Ingeniero en Tecnologías de la Información son absolutamente originales, auténticos y personales.

En tal virtud, declaramos que el contenido, las conclusiones y los efectos legales y académicos que se desprenden del trabajo propuesto de investigación y luego de la redacción de este documento son y serán de nuestra sola y exclusiva responsabilidad legal y académica.

Igualmente, declaramos que todo resultado académico que se desprenda de esta investigación y que se difunda tendrá como filiación la Pontificia Universidad Católica del Ecuador Sede Santo Domingo, reconociendo en las autorías al director del Trabajo de Integración Curricular y demás profesores que amerita.

Además, declaro que el presente trabajo, producto de las actividades académicas y de investigación, forma parte del capital intelectual de la Pontificia Universidad Católica del Ecuador, Sede Santo Domingo, de acuerdo con lo establecido en el artículo 16, literal j), de la Ley Orgánica de Educación Superior.

En tal razón, autorizo a la Pontificia Universidad Católica del Ecuador, Sede Santo Domingo, para que pueda hacer uso, con fines netamente académicos, del Trabajo de Integración Curricular, ya sea de forma impresa, digital y/o electrónica o por cualquier medio conocido o por conocerse, siendo el presente documento la constancia del consentimiento autorizado; y, para que sea ingresado al Sistema Nacional de Información de la Educación Superior del Ecuador para su conocimiento público, en cumplimiento del artículo 103 de la Ley Orgánica de Educación Superior.



Herrera Chancay Jordan Stalin
C.C. 1722153887



Paredes Romero Jenniffer Tahis
C.C. 1316463387

INFORME DE TRABAJO DE INTEGRACIÓN CURRICULAR ESCRITO

Mikel Ugando Peñate, PhD

Responsable de Investigación Formativa

Pontificia Universidad Católica del Ecuador Sede Santo Domingo

De mi consideración,

Por medio del presente informe en calidad de director del Trabajo de Titulación del Grado de Ingeniería en Tecnologías de la información titulado: APLICACIÓN WEB CON MACHINE LEARNING PARA LA GESTIÓN DE HISTORIAL MÉDICO DE MASCOTAS EN EL HOSPITAL VETERINARIO SANTA MARTHA DEL CANTÓN SANTO DOMINGO, realizado por los estudiantes: Herrera Chancay Jordan Stalin con cédula de ciudadanía 1722153887 y Paredes Romero Jenniffer Tahis con cédula de ciudadanía 1316463387, previo a la obtención del título de Ingeniero en Tecnologías de la Información, informo que el presente Trabajo de Integración Curricular escrito se encuentra finalizado conforme a la guía y al formato de la Sede vigente.

Además, certifico haber verificado la originalidad y autenticidad del trabajo de integración curricular por medio del programa anti plagio Turnitin, en respuesta a la normativa institucional vigente.

Santo Domingo, 20/03/2026.

Atentamente,



Mg. Ocampo Pazos Willian Javier

Profesor Titular Auxiliar II

RESUMEN

El presente trabajo de titulación tiene como finalidad fortalecer la gestión del historial médico de las mascotas del Hospital Veterinario Santa Martha, ubicado en el cantón Santo Domingo. La investigación se fundamenta en una base teórica creada a partir del análisis de diversas fuentes bibliográficas que contienen antecedentes de aplicaciones *web*, *machine learning* y la gestión de clínicas veterinaria. La metodología utilizada corresponde a un enfoque cuantitativo, con un diseño aplicado y de campo. Para la recolección de información se consideró la opinión de los dueños de las mascotas. Se implementó la aplicación *web* desarrollada con *Python*, usando *Vue.js* y *FastAPI*, para la base de datos se utilizó *MySQL* y la arquitectura la modelo vista controlador (MVC). Esta propuesta permite el registro, consulta y administración eficiente de los historiales clínicos de las mascotas, incorporando técnicas de *machine learning*, usando árbol de decisión como algoritmo determinado. El sistema propuesto contribuye a la optimización de los procesos médicos, mejorando la organización de los datos y la calidad del servicio veterinario brindado a los dueños de las mascotas.

Palabras clave: Historial Médico Veterinario, Aplicación *Web*, *Machine Learning*, Gestión Veterinaria.

ABSTRACT

This work aims to strengthen the management of pet medical records at the Santa Martha Veterinary Hospital, located in the Santo Domingo canton. The research is based on a theoretical framework developed from the analysis of various bibliographic sources containing background information on web applications, machine learning, and veterinary clinic management. The methodology employed is a quantitative approach with an applied and field-based design. The opinions of pet owners were considered for data collection. A web application was implemented using Python, Vue.js, and FastAPI. MySQL was used for the database, and the architecture was the Model-View-Controller (MVC). This proposal allows for the efficient registration, querying, and management of pet medical records, incorporating machine learning techniques using a decision tree as the chosen algorithm. The proposed system contributes to the optimization of medical processes, improving data organization and the quality of veterinary service provided to pet owners.

Keywords: Veterinary Medical History, Web Application, Machine Learning, Veterinary Management.

ÍNDICE DE CONTENIDOS

1. INTRODUCCIÓN	10
1.1. Antecedentes	10
1.2. Planteamiento y delimitación del problema.....	12
1.3. Preguntas de investigación.....	13
1.3.1. Pregunta general	13
1.3.2. Preguntas específicas.....	13
1.4. Justificación.....	14
1.5. Objetivos de investigación	16
1.5.1. Objetivo general.....	16
1.5.2. Objetivos específicos	16
2. REVISIÓN DE LA LITERATURA.....	17
2.1. Fundamentos Teóricos	17
2.1.1. Aplicación web	18
2.1.2. Machine learning.....	29
2.1.3. Historial médico	34
2.2. Predicción científica	39
3. METODOLOGÍA.....	40
3.1. Enfoque, alcance, diseño y tipo de investigación	40
3.2. Unidades de análisis.....	42
3.3. Técnicas e instrumentos de investigación	43
3.4. Técnicas de análisis de datos	43
3.5. Operacionalización de las variables	44
4. RESULTADOS	47
4.1. Resultados del primer objetivo específico.....	47
4.1.1. Aplicación de la encuesta a los clientes.....	47
4.1.2. Identificación de los procesos en la gestión de historiales médicos de mascotas.....	51

4.2.	Resultado del segundo objetivo específico.....	53
4.2.1.	Herramientas tecnológicas	53
4.3.	Resultado del tercer objetivo específico.....	58
4.3.1.	Nomenclatura y logotipo de la aplicación <i>web</i> con <i>machine learning</i>	58
4.3.2.	Marco de Trabajo Scrum	59
4.3.3.	Sprint 1	59
4.3.4.	Sprint 2.....	76
4.3.5.	Sprint 3.....	89
4.4.	Resultado general.....	109
4.4.1.	Implementación de aplicación web con machine learning.....	109
4.4.2.	Validación de la propuesta	112
4.4.3.	Validación de la hipótesis	117
5.	DISCUSIÓN.....	120
6.	CONCLUSIONES Y RECOMENDACIONES	122
6.1.	Conclusiones.....	122
6.2.	Recomendaciones	123
7.	REFERENCIAS	124
8.	ANEXOS.....	133

1. INTRODUCCIÓN

En la actualidad, la digitalización de los procesos administrativos y clínicos en el área de los animales, es una necesidad urgente para el fortalecimiento de la calidad del servicio, tanto a los dueños de mascotas, como a los especialistas en salud animal. La automatización de los registros médicos reduce probabilidades de error en la gestión de los recursos a usar. El manejo manual de datos médicos puede causar pérdidas de información, fallas en el seguimiento de tratamientos y una atención interrumpida. Frente a esta situación, el crear herramientas digitales aparece como una buena solución para cuidar y usar fácilmente los datos.

Por consecuencia, Sommerville (2011) explica que, el enfoque en desarrollo de software basado en servicios consiste en combinar y configurar distintos servicios para formar soluciones compuestas. Estas pueden integrarse con interfaces *web* o funcionar como componentes dentro de otros servicios, pueden ser diseñados específicamente para unas aplicaciones empresariales desarrollados dentro de una compañía u organización o provenir de terceros (p. 527).

1.1. Antecedentes

El uso de tecnologías de la información para una correcta administración de historiales médicos en las mascotas evita errores humanos, facilita el diagnóstico de enfermedades, permite compartir información actualizada de forma remota con la facilidad de acceder a los datos de manera más rápida.

A partir de este contexto, se planteó la problemática tratada por Zaninelli et al. (2007) en Italia, quienes desarrollaron un sistema de *software* de código abierto orientado a reducir los costos de implementación y mantenimiento en entornos veterinarios. Uno de los objetivos más resaltantes de esta aplicación fue su gran compatibilidad con otros centros veterinarios o centros de investigación sin generar costos adicionales. Los resultados de su implementación mostraron una valoración positiva por parte de los

profesionales involucrados. Según los datos recopilados el 71.4% de los veterinarios destacó la mejora en la disponibilidad de los datos clínicos útiles, el 78.58% señaló una mayor calidad en los procedimientos de diagnóstico y el 85.58% percibió una mejora significativa en la eficiencia de las actividades hospitalarias (p. 1).

De igual manera, en Arabia Saudita Hina et al. (2024), presentaron un proyecto que identifica las problemáticas que pueden atravesar los dueños de mascotas. Es por eso que, se creó "*Vet the Pet*" como solución a los problemas enfrentados, esta aplicación ayudó a obtener un mejor control de la salud del animal, anticipar enfermedades y a tratarlas. Además, la comunicación con los veterinarios fue más objetiva, ya que facilita informar comportamientos de la mascota, citas próximas y asesoramientos en tratamientos.

De manera similar, se plantea el problema abordado por Castro y Tejada (2020) en Perú, donde se evidenció una solución práctica para las deficiencias identificadas que tuvo el centro veterinario. Dichos problemas estaban relacionados a los procesos de atención, como la recepción de pacientes, gestión de citas, búsqueda de historiales, entre otros. Para solucionar esta problemática, se desarrolló una aplicación capaz de permitir dichos destronamientos, siguiendo los pasos estandarizados por la metodología RUP. Los resultados muestran que mediante la implementación del *software* se minimizó el tiempo de espera en el agendamiento de citas en un 91%, se mejoró el tiempo de atención y el tiempo de búsqueda se redujo a un 73% (pp.5, 20-21).

Por otra parte, Cedeño et al. (2021) estudiantes de la Universidad Estatal de Milagro en Guayas - Ecuador, desarrollaron una propuesta basada en una aplicación *web* diseñada de manera iterativa e incremental. Esta herramienta permitió alcanzar el objetivo de digitalizar y gestionar eficazmente los historiales médicos de mascotas y procesos administrativos en clínicas veterinarias evitando la redundancia, pérdida o ausencia de datos, manteniéndolos actualizados y contribuyendo al cuidado del medio ambiente

mediante la reducción de residuos físicos, maximizando rendimiento con rentabilidad (pp.111, 118-119).

1.2. Planteamiento y delimitación del problema

De los antecedentes analizados, se destacó que la incorporación de recursos tecnológicos en hospitales veterinarios es fundamental para mejorar la calidad de atención, tiempos de espera para consultas médicas, detección y tratamiento de enfermedades. Es decir, contribuyeron en el fortalecimiento de los procesos administrativos de las necesidades de los clientes. Los cuatro artículos revisados concretan la idea de que la mala gestión en las clínicas veterinarias a nivel mundial crea la necesidad de automatizar los procesos de salud de las mismas.

La gestión de historiales médicos de mascotas ha representado nuevos retos para los profesionales veterinarios, especialmente en darle un seguimiento adecuado a cada uno de sus pacientes. Una gestión deficiente de los tratamientos puede aumentar el riesgo de que estos no se lleven a cabo correctamente, afectando la salud del animal. Según afirma Cagri et al. (2024) en Turquía, la falta de mecanismos eficaces de notificación influye negativamente en el seguimiento clínico de los pacientes, lo que puede derivar en errores médicos. Aunque muchos de estos errores no generan efectos adversos evidentes, existe un número significativo que puede causar lesiones permanentes o incluso la muerte de los mismos (p. 2).

Según lo mencionado por Cagri et al. (2024) en Turquía, las quejas de los dueños de mascotas han revelado aspectos críticos en la calidad de atención que reciben sus mascotas. El estudio se realizó basado en el análisis de 127 expedientes de quejas recopiladas entre el 2012 y 2021, permitió identificar un total de 296 problemas específicos relacionados directamente con la atención veterinaria. Mediante una codificación estructurada de tres niveles, incluyendo dominios clínicos, de gestión y de relación, se logró clasificar y comprender la naturaleza de las inconformidades. De los

problemas registrados, un 62% corresponde al ámbito clínico, un 24% a la gestión administrativa y un 14 % al comportamiento de los veterinarios asistentes (p. 4).

Continuando con el mismo enfoque, conforme a los datos presentados por Haz et al. (2017), la implementación de un sistema para gestionar procesos clínicos en las veterinarias es esencial para disminuir el tiempo de respuesta y garantizar la información de las mascotas. La implementación del sistema *web* que presentaron en la clínica veterinaria “Mascotas” en Santa Elena – Ecuador, optimizó hasta el 80% del flujo de la información, se presentaron procesos claves como historiales médicos, inventarios y facturación (pp.26-30).

Por esa razón, se decidió visitar el Hospital Veterinario Santa Martha ubicado en Santo Domingo, mediante una reunión de planificación con el propietario, se evidenció que cuentan con un registro de carnet en formato físico de los pacientes, el mismo que provoca la desactualización de la información clínica. De igual forma existen casos de carnets no legibles por el uso o incidentes externos, lo que puede provocar riesgo de diagnósticos erróneos o administración de tratamiento no necesarios. Por otro lado, la ausencia de un control para agendar citas médicas provoca pérdidas de tiempo e insatisfacción de los clientes respecto al servicio (Anexo 10).

1.3. Preguntas de investigación

1.3.1. Pregunta general

¿Cómo fortalecer la gestión de historiales médicos de mascotas en el hospital veterinario Santa Martha del cantón Santo Domingo?

1.3.2. Preguntas específicas

- ¿Cuáles son los procesos que tiene el hospital veterinario respecto a la gestión de historiales médicos de mascotas?
- ¿Qué herramientas tecnológicas son necesarias para la creación de una aplicación *web* con *machine learning*?

- ¿Qué soluciones en base al desarrollo *web* se pueden implementar para la gestión de historial médicos de mascotas?

1.4. Justificación

El trabajo de investigación presentado tuvo como propósito demostrar los motivos necesarios para la implementación de un control sobre los historiales médicos en las veterinarias. Según la Constitución de la República del Ecuador (2008), en el artículo 71 reconoce los derechos de la naturaleza, lo que engloba la protección de animales (mascotas) como parte del medio ambiente (p.35). Además, menciona en el artículo 83 (numeral 6) que los ecuatorianos y ecuatorianas tienen el deber de preservar un ambiente sano, haciendo uso de la naturaleza de modo beneficioso para la misma (p.41).

Respecto a la inteligencia artificial la Asamblea Nacional del Ecuador propone la Ley Orgánica de regulación y promoción de la Inteligencia Artificial en Ecuador, con el objetivo de implementar un marco legal para la implementación y uso de la IA en el país. Además, busca impulsar avances médicos, mejorar procesos educativos y servicios públicos con enfoque en la dignidad y respeto de la información de los ecuatorianos (p.4). En el artículo 17 de la constitución ecuatoriana se afirma que el estado garantizará la creación y fortalecimiento de nuevas tecnologías para el aprovechamiento de redes y aplicaciones con innovaciones (p.15).

En este contexto, el desarrollo de la aplicación *web* con *machine learning* propuesta, se alineó con los principios antes mencionados, ya que busca aprovechar la innovación tecnológica para fortalecer la gestión de procesos de historiales médicos de mascotas. De la misma manera, este trabajo también se justifica con lo establecido por la Convención sobre la Diversidad Biológica (CBD), que es un tratado promovido por la Organización de Naciones Unidas (1993) y aprobado en Ecuador, en donde se establece que la diversidad biológica es base fundamental para la vida (p.6). En este sentido las veterinarias desarrollan un papel importante no solo en la atención médica, sino también en la protección de las especies y monitoreo de enfermedades.

Como complemento en la justificación de este trabajo, se tomó en cuenta que Ecuador afirmó su compromiso con La Organización de las Naciones Unidas para la Educación, Ciencia y la Cultura (UNESCO), en 2021 para la implementación de la inteligencia artificial que sirve para el uso y beneficio de los ecuatorianos respecto a las tecnologías en ámbitos de salud, comercio, educación e impactos ambientales (p.3).

Además, el Código Orgánico del Ambiente (2018) tiene como finalidad la protección del medio ambiente, la biodiversidad y los seres vivos que la habitan, incluyendo las mascotas. Aunque el COA está enfocado en la gestión ambiental y cuidado de los recursos ambientales también dedica importancia a los animales domesticados. El artículo 147 menciona las prohibiciones específicas que atenten contra la integridad del animal, como la donación, el abandono, adopción para experimentación de productos, destacando que para los casos mencionados tendrán sanciones de acorde al nivel del daño (pp. 45-46).

De igual manera, este trabajo tomó a consideración los objetivos establecidos por la Ley Orgánica de Bienestar Animal (2018), LOBA es una normativa propia de Ecuador, que regula y garantiza el cuidado de los animales, estableciendo mayor importancia en aquellos que conviven con los seres humanos. Esta ley establece que las mascotas, en su mayoría perros y gatos son seres que merecen un trato digno, condiciones adecuadas de salud, higiene y alimentación. Además, uno de los aspectos más importante de la LOBA es promover la tenencia responsable de mascotas, para ello esta ley establece medidas como la esterilización, vacunación y prevención de enfermedades. Asimismo, contempla la supervisión de centros veterinarios para que cumplan estándares éticos y técnicos, sancionando a quienes lo incumplan (pp.111-112).

En este sentido, se busca implementar mecanismos que permitan fortalecer la gestión del historial médico de mascotas, favoreciendo el seguimiento correcto del estado de salud de las mismas. De la misma manera, este proyecto se sustentó en base a la Ordenanza Municipal No. E-019-VQM, del GAD Municipal de Santo Domingo (2015),

donde se establece la tenencia responsable de perros u otros animales domésticos. En el artículo 3, se recalca que los dueños son responsables de mantenerlos en condiciones higiénicas y saludables, evitando que se produzca algún tipo de sufrimiento, es obligación del dueño mantener la vacunación necesaria y tratamientos que eviten enfermedades a las mascotas en la forma que considere la autoridad sanitaria (p. 3).

Por consiguiente, los documentos citados anteriormente promueven el respeto a los animales garantizando su cuidado y servicios de calidad respecto a su salud, razón por la que se presenta "*Medical Pets*" como ayuda necesaria en la gestión de la información de historiales médicos de mascotas de manera digital en el hospital veterinario Santa Martha, optimizando tiempo y recursos.

1.5. Objetivos de investigación

1.5.1. Objetivo general

Implementar una aplicación *web* con *machine learning* para el fortalecimiento de la gestión de historiales médicos de mascotas en el hospital veterinario Santa Martha del cantón Santo Domingo en el periodo 2025.

1.5.2. Objetivos específicos

- Identificar los procesos que tiene el hospital veterinario Santa Martha respecto a la gestión de historiales médicos de mascotas.
- Determinar herramientas tecnológicas que sean de utilidad para el desarrollo e implementación adecuado la aplicación *web* planteada.
- Desarrollar la aplicación *web* con *Machine learning* para la gestión de historiales médicos de mascotas en el hospital veterinario Santa Martha.

2. REVISIÓN DE LA LITERATURA

2.1. Fundamentos Teóricos

Para la realización de fundamentos teóricos se usaron esquemas para cada una de las variables mencionadas (independientes y dependiente) que definen de forma gráfica la estructura del presente Trabajo de Titulación de Grado. En palabras de Gómez (2012), cualquier campo científico de una investigación necesita de conocimientos y bases teóricas que expliquen de forma explícita el fenómeno de estudio, es decir el marco teórico permite conocer una descripción sustentada del problema (pág. 43).

Se explica en la Figura 1 la variable independiente denominada aplicación *web*, en la Figura 2 se detalla la variable independiente *machine learning* y la Figura 3 se centra en la variable dependiente llamada historial médico.

Figura 1. Fundamento teórico de la variable independiente aplicación *web*

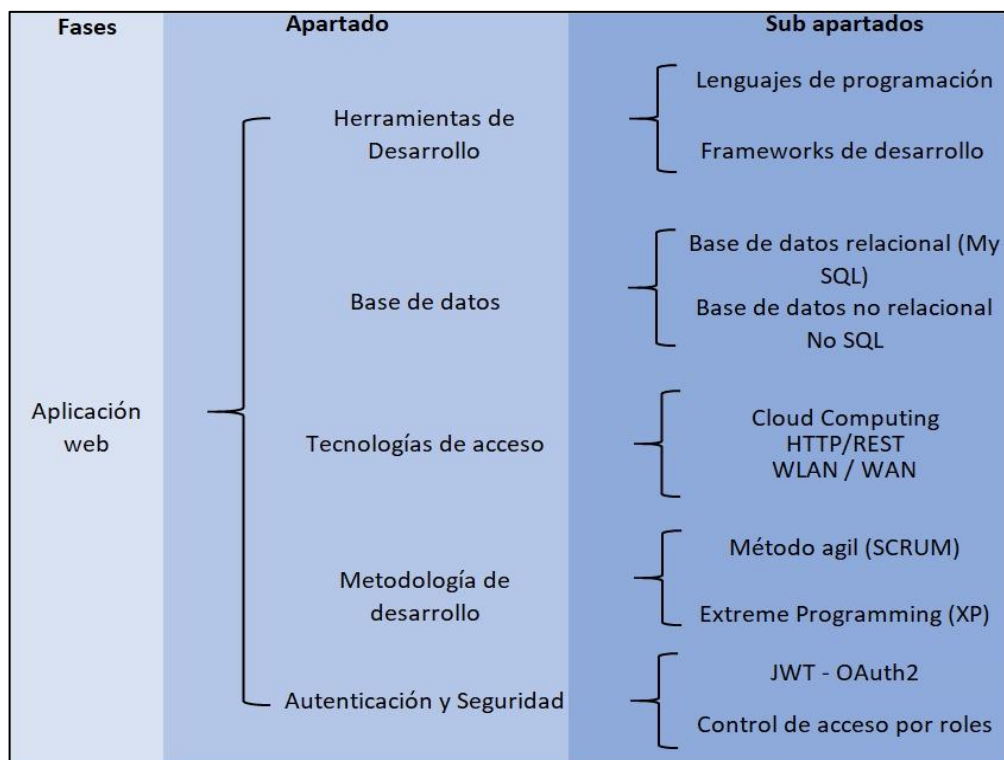


Figura 2. Fundamento teórico de la variable independiente *machine learning*

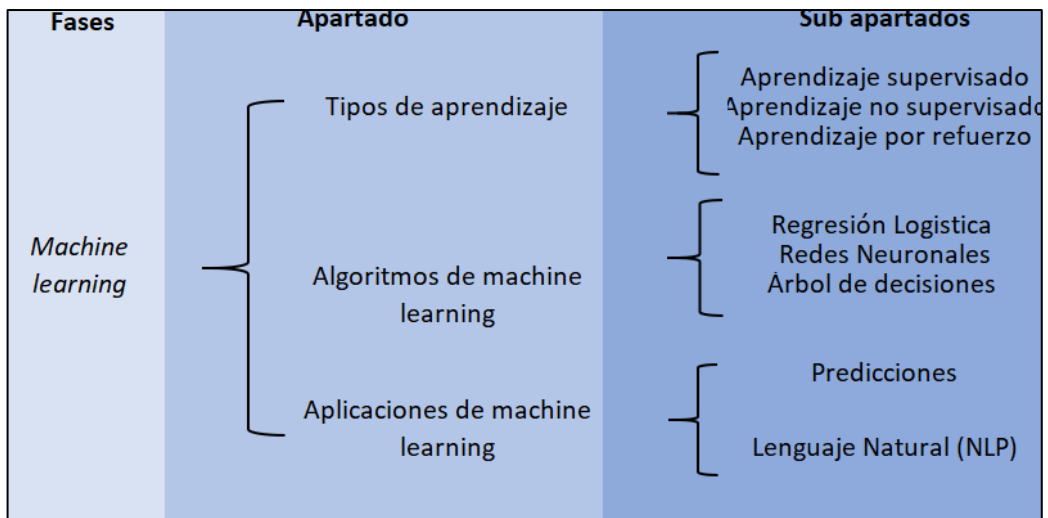
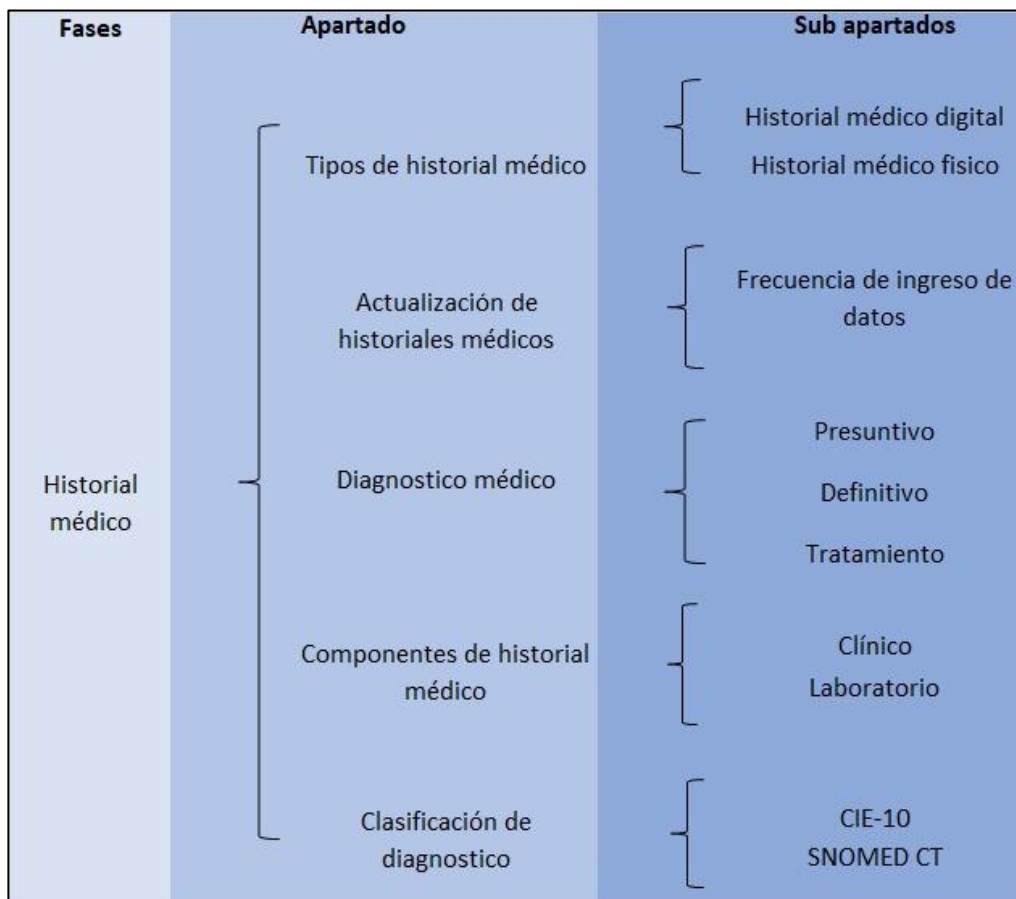


Figura 3. Fundamento teórico de la variable dependiente historial médico



2.1.1. Aplicación web

Según Molina et al. (2016), las aplicaciones *web* son herramientas que permiten realizar actividades a través de un dispositivo móvil que cuente con acceso a internet,

mejorando el tiempo empleado en cada solicitud, esto ha influido para la aceptación y usabilidad de los usuarios. Además, menciona que las aplicaciones *web* usan el formato base de *HTML* (Lenguaje de Hipertexto) para ejecutar las operaciones que más de un usuario desee de manera simultánea, siendo esto otra característica importante de las mismas.

Para Hernández-Rueda et al. (2022), una aplicación *web* es un sistema al que se ingresa con acceso a internet mediante un navegador y que funciona en un servidor *web*. Este tipo de sistema utiliza tecnologías para el desarrollo de software como *HTML*, *JavaScript* y *CSS* con el objetivo de crear interfaces de fácil entendimiento para el usuario. De manera que el acceso y relación con la aplicación sea a través de internet.

Amazon Web Services (AWS, 2023), presenta a las aplicaciones *web* como un *software* que se ejecuta mediante el navegador *web*, permitiendo el acceso de manera remota a las empresas para intercambiar su información. Menciona que las funciones más comunes de las páginas *web* como *e-commerce*, canales de noticias o mensajería instantánea conservan el diseño de las aplicaciones *web*. De igual manera recalca como beneficios la accesibilidad, el desarrollo eficiente, la escalabilidad y la simplicidad para el usuario de interactuar con las mismas.

Enfocándose en las tecnologías de desarrollo implementadas, se especifica *HTML*, conocido como Lenguaje de Marcado de Hipertexto y denominado como lenguaje estándar para crear páginas *web*. Esto hace que el archivo emitido sea más fácil de entender en comparación a un archivo de texto plano (*txt*), *HTML* sirve para modelar documentos con el fin de llegar a un navegador y poder ser interpretado de manera clara. Es decir, define la estructura de formularios que se presentarán en una página *web*, usando etiquetas para organizar el texto como menciona su definición (Luján, 2016).

2.1.1.1. Herramientas de desarrollo

De acuerdo con Pérez et al. (2021), el desarrollo de las aplicaciones *web* se conforma de dos partes fundamentales: *FrontEnd* que es la capa de presentación, hace que el usuario

pueda interactuar de forma comprensible con el sistema. Así mismo el *BackEnd* es la capa de acceso a los datos, se encarga de la manipulación de la información que será presentada en el *FrontEnd*, es decir el *BackEnd* no serviría de mucho si no existiera un *FrontEnd* que ayude al usuario a interpretar los servicios ofrecidos. Los desarrolladores deben tener noción teórica y práctica de la implementación de estas dos capas para el correcto funcionamiento de una aplicación web. Sin embargo, generalmente se establecen equipos de trabajos enfocados en cada parte mencionada para mejorar el rendimiento en el proceso de desarrollo del software (pp. 347-348).

2.1.1.2. Lenguajes de programación

En palabras de Camazón (2010), los lenguajes de programación son formas de crear aplicaciones ejecutables en un computador, intercambian datos entre la máquina y la lógica del programador, recalca que existen distintos tipos de lenguajes. Lenguajes de Marcas utilizados para modificar documentos, las marcas indican al programa como debe ser tratado su contenido, ejemplos de estos son *HTML*. Lenguajes de *Script* en navegador son lenguajes que sirven para indicar acciones que se realicen dentro de un programa, como por ejemplo validar formulario, este lenguaje se ejecuta cuando el usuario accede a la aplicación. Por último, los Lenguajes de *Script* en servidores se ejecutan cuando se realizan peticiones al servidor, uno de los lenguajes específicos para trabajar del lado del servidor es PHP (pp.9,17).

De la misma manera Trejos y Muñoz (2023), afirman que los lenguajes de programación son herramientas compuestas de instrucciones que una computadora puede ejecutar. Dentro de sus funciones principales está interpretar el proceso de las indicaciones del programador basándose en la lógica humana aplicándola en la lógica computacional. El desarrollador organiza las instrucciones respetando las normas de cada lenguaje, con el objetivo de definir una comunicación clara entre la máquina y los resultados esperados.

2.1.1.3. Frameworks de Desarrollo

Según lo expuesto por Avilés et al. (2020), los *frameworks* son herramientas tecnológicas que optimizan el tiempo y costo en el desarrollo de programas. Así mismo señala que la implementación de ellos mejora las etapas del desarrollo, facilita mayor rapidez para ejecutar las tareas, incluso mejorar los tiempos establecidos en los contratos, creando aplicaciones más organizadas y dando una mejor experiencia al usuario. Además, menciona que a menudo los desarrolladores optan por utilizar *framework backend* a través de librerías que añaden funcionalidades para mejorar habilidades y eficiencia en sus códigos, ejemplos de estos son: *Django (Python)*, *Laravel (PHP)*, *Spring Boot (Java)*, etc (p.4).

En palabras de Molina et al. (2016), los *frameworks* son instrumentos importantes en la creación de una aplicación *web*, porque facilitan el trabajo a los desarrolladores, permitiendo acelerar el proceso de creación a través de la reutilización de código. Además, fomentan el uso de buenas prácticas de ingeniería implementando patrones de diseño. Especifica también, que cuando se habla del desarrollo de aplicaciones siempre es importante garantizar y optimizar el trabajo, es decir los *frameworks* aparecen como una alternativa eficiente para la programación siempre y cuando se cumplan con las necesidades establecidas (pp.204-206).

Prieto et al. (2020), define los *frameworks* como un conjunto de clases para proporcionar diseños generalizados a cierto tipo de aplicaciones, asignando responsabilidad y estableciendo colaboración entre las clases que forman un modelo. Además, los *frameworks* tiene puntos positivos para el desarrollador, coincide con otros autores en la reutilización de código existente para proporcionar la funcionalidad específica rellenando los llamados “puntos calientes”.

2.1.1.4. Base de datos

Según Reinososa et al. (2012), las bases de datos es un conjunto organizado de datos estructurados y recopilados en un sistema informático que permite la gestión y acceso a la

misma. Menciona también que la gestión correcta de una base de datos tiene como objetivo final evitar la redundancia y accesos no autorizados, los datos ingresados se almacenan de manera masiva como un disco de información con derecho de permisos (p.3).

Oracle (2020), define a las bases de datos como una recopilación de información almacenada de forma digital en un *software*. Explica que para manejar una base de datos de manera eficiente debe estar controlada por un sistema gestor de base de datos (DBMS). Así funcionan de manera grupal los datos y el DBMS, de forma conjunta con las aplicaciones asociadas para modificar, actualizar, controlar y organizar fácilmente la información normalmente emplean lenguajes de consultas estructuradas conocido como *SQL*.

2.1.1.4.1. Base de datos relacional

Amazon Web Service (2024), afirma que las bases de datos relacionales estructuran la información en tablas, que están formadas por filas y columnas. Cada columna define un tipo específico de dato a diferencia de que cada fila contiene un registro individual. Así mismo a cada tabla se le asigna una clave primaria que actúa como identificador único para crear vínculos con otras tablas, a estas claves se las conoce como “Claves foráneas”. Es decir, gracias a la relación que existen entre las tablas es posible consultar y recuperar información con la instrucción *SQL*, lo que facilita una interacción sencilla con la base de datos.

Así mismo Suárez (2008), declara que la base de datos relacional es el modelo más utilizado en la actualidad para aplicaciones *web*, ya que permite establecer conexiones entre información almacenadas en las tablas, de allí proviene el nombre “Modelo relacional”. En el mismo aspecto manifiesta que las características principales de este modelo son que no existe la redundancia, ya que no puede haber dos tablas con el mismo identificador, de la misma manera resguardan la claridad e integridad de los datos con el registro de claves primarias, existiendo la relación entre tablas (padre e hijo).

2.1.1.4.2. Base de datos no relacional

Amazon Web Service (2024), explica que existen diversos tipos de base de datos no relacionales ya que cada tipo maneja y estructura los datos de manera distinta, especialmente al no requerir una estructura determinada. A diferencia de bases de datos relacionales estas permiten almacenar información sin seguir ningún tipo de esquemas. Los “Datos sin esquema”, son aquellos en el cual no es necesario cumplir el formato de tablas (filas y columnas) lo que genera mayor flexibilidad al momento de interactuar con la base de datos. Esta característica es ideal para sistemas empresariales que trabajan con grandes volúmenes de información y que no pueden parar su producción por un cambio en la base de datos, ya que poseen datos a tiempo real como aplicaciones *e-commerce* o contenidos multimedia.

En su investigación sobre la base de datos Chingo y López (2021), vinculan las bases de datos no relacionales con principios técnicos de ingeniería, resaltando la aplicación en el principio BASE. Este principio se basa en el Teorema CAP (Teorema de Brewer), el mismo que plantea las propiedades esenciales de: Consistencia, Disponibilidad y Tolerancia. La propiedad de consistencia asegura que los nodos del sistema reflejen los datos de manera simultánea. Por otra parte, la disponibilidad garantiza que el sistema siga funcionando adecuadamente en caso de que un nodo falle. Así también la tolerancia a particiones genera que el sistema pueda seguir operando de manera estable aun cuando haya pérdida de conexión entre los nodos (p.3).

2.1.1.5. Tecnologías de acceso

De acuerdo con Sommerville (2011), las tecnologías que permiten acceder a las aplicaciones *web* hacen posible que el usuario y el servidor interactúen usando varios caminos, tales como *HTTP/HTTPS*, *API REST* y sistemas de autenticación con *tokens*, lo cual ayuda a que la comunicación sea segura y rápida. Aparte de esto, usar plataformas como *Angular* o *React* hace que el usuario tenga una mejor experiencia, gracias a interfaces que son dinámicas y se adaptan bien. Para que estas tecnologías trabajen sin problemas, es clave tener una arquitectura bien pensada tanto en el lado del cliente como en el del servidor,

apoyada en servicios que controlen el acceso y también en *JWT (JSON Web Tokens)* para verificar quién es quién, lo que asegura que los datos estén protegidos durante las operaciones. Dicho de otro modo, al diseñar aplicaciones *web* hay que pensar no solamente en lo que hacen, sino también en qué tan seguras y adaptables son, pues estas tecnologías siempre están cambiando y son vitales para que el sistema funcione bien.

2.1.1.5.1. Cloud computing

De acuerdo a Mell y Grance (2011), miembros del Instituto Nacional de Estándares y Tecnología (NIST), declaran que la computación en la nube es como tener a disposición recursos de alta tecnología cuando los necesites, a través de la *web*. Se habla de guardar datos, usar servidores o apps, sin tener que manejarlos directamente, es un modelo que nos da acceso fácil y desde cualquier lugar a una o varias herramientas que se adaptan a lo que se necesite. Esta manera de trabajar baja los gastos, hace que sea más fácil crecer y ayuda a que las empresas innoven. Además, las deja enfocarse más en lo que se desee de manera comercial que en mantenimiento de equipos. Se puede usar de forma pública o privada, dependiendo de lo que cada empresa busque. Es decir, la nube cambia por completo cómo las empresas usan la tecnología para su beneficio.

Por otro lado, Armbrust et al. (2010) señalan que la computación en la nube es una transformación profunda en la forma en que se entregan los servicios tecnológicos. Plantean que introduce un esquema económico que se basa en el consumo efectivo de los recursos, como si se tratase de un servicio básico. Este enfoque impulsa la eficiencia energética y el aprovechamiento compartido de recursos a gran escala, lo cual es vital en entornos de negocios con necesidades de procesamientos avanzados (p.2).

Finalmente, Marinis y Briscoe (2009) resaltan que la nube permite a los usuarios ajustarse con rapidez a los cambios del mercado gracias a su flexibilidad. No obstante, también presenta retos considerables, como la protección de la información, la sujeción al proveedor y la adaptación a nuevos recursos. Sin embargo, las ventajas operativas y estratégicas siguen fomentando su aceptación a nivel mundial. Es decir, recalca que la

computación en la nube modifica tanto la tecnología como la manera en que se gestionan las empresas (p.473).

2.1.1.5.2. HTTP/REST

De acuerdo con Iglesias-Urkia et al. (2018) el protocolo *HTTP* se clasifica como un protocolo sin estado, es decir que la interacción que exista entre el cliente y el servidor, se generan de manera independiente, sin que el servidor conserve información sobre las peticiones realizadas. Asegura que *HTTP* es un protocolo utilizado para la transferencia de datos donde el servidor no tiene memoria, lo cual presenta una ventaja en el término de eficiencia, pero desventaja de la información en persistencia de datos que haya entre sesiones.

Por su parte Gourley y Totty (2002), mencionan que *HTTP* es el lenguaje del internet de manera global, ya que todas las aplicaciones a las que se accede de manera *web* se comunican a través de *HTTP*. Lo denominan “Mensajero Multimedia”, debido a que *HTTP* utiliza protocolos fiables de transmisión de datos, eso es bueno para un usuario que se preocupa por la integridad de su información, los servidores *HTTP* almacenan datos y los proporcionan cuando los clientes lo necesiten (pp. 2-4).

2.1.1.5.3. WLAN/WAN

Para Perry (2020), las redes *WLAN* conocidas como redes inalámbricas locales facilitan el acceso a internet sin necesidad de cables *ethernet* en sitios como la oficina, la universidad o una casa. Afirma que tener conexiones *WLAN* que no fallen es crucial para que los servicios web funcionen de manera correcta. Estas redes, al seguir normas como IEEE 802.11 dan velocidades buenas para usar programas que necesitan enviar datos al instante, aparte, ponerlas en marcha es fácil y barato, lo que las hace perfectas para negocios pequeños. Las aplicaciones web, al poderse usar desde navegadores sin instalar nada, se benefician de esta conexión inalámbrica. Igualmente, si se les añaden sistemas para verificar quién entra y para proteger los datos, las *WLAN* pueden ser sitios seguros para trabajar en

internet. Concreta en que las *WLAN* son una base fundamental para poder usar los servicios digitales de hoy en día (p.195)

En el contexto de las *WAN*, Perry (2020), aclaran que las *WAN* son esenciales para las compañías con sedes en varios sitios, ya que asegura que se pueda acceder desde fuera a los sistemas internos y a las plataformas web de la empresa. Menciona que estas redes usan tecnologías como *VPN*, *MPLS* o conexiones por satélite para que las conexiones grandes sean seguras y fiables. Declara que las aplicaciones *web* que dependen de servidores centrales o en la nube necesitan que la conexión *WAN* sea buena para que respondan rápido. Por eso, una *WAN* bien configurada puede mejorar mucho cómo la experiencia del usuario al final (p. 243). Es decir, tanto las *WLAN* como las *WAN* son piezas clave para que las aplicaciones *web* tengan siempre disponibilidad y buen rendimiento.

2.1.1.6. Metodologías de Desarrollo

Según Quitian et al. (2022), las metodologías de *software* de desarrollo pueden definirse en dos grandes enfoques: tradicionales y ágiles. Las metodologías tradicionales, como el esquema en cascada, destacan por su manera de actuar paso a paso, donde cada etapa del desarrollo debe acabarse antes de avanzar a la siguiente. Este patrón es práctico donde las peticiones son fijas y bien trazadas desde el comienzo, permitiendo un mayor dominio documental y vigilancia. No obstante, indaga en que su inflexibilidad restringe la adaptación a modificaciones inesperadas durante el avance. Así mismo, el modo tradicional es conveniente para ciertos proyectos, pero las metodologías ágiles mandan en casos dinámicos y mutables. Los autores expresan que estas metodologías hacen más fácil la detección pronta de fallos, aumentan la alegría del cliente y hacen más fuerte el trabajo en grupo por medio de juntas frecuentes y retroalimentación incesante. Este esquema se vuelve especialmente útil en proyectos novedosos, donde los requisitos pueden cambiar con el tiempo. Además, se adapta mejor a equipos de varias disciplinas y dispersos, gracias a su estructura cooperativa. Sin embargo, también trae retos como la necesidad de una gran disciplina por parte del equipo y una gestión correcta de los cambios. En este marco,

concluyen que escoger la metodología correcta depende de cosas como el tipo de producto, el ambiente del cliente y la madurez del equipo de desarrollo. Por esto, conocer y comparar ambas metodologías es clave para una toma de decisiones buena en proyectos de *software* (pp. 11-12).

2.1.1.6.1. SCRUM

De acuerdo con Schwaber y Sutherland (2020), los autores de la guía esencial de *Scrum*, este método impulsa la transparencia, la revisión y la capacidad de ajuste en todo momento, con figuras clave como el *Product Owner*, el *Scrum Master* y el grupo de desarrollo. Cada sprint culmina con una revisión, lo que favorece la mejora continua del producto y del equipo. Igualmente, *Scrum* pone énfasis en una comunicación fluida con las partes interesadas, facilitando la jerarquización de tareas a través de la lista de pendientes del producto. Este esquema resulta muy práctico en escenarios dinámicos donde las especificaciones no están claras desde el principio. Dada su naturaleza repetitiva, ayuda a minimizar peligros y a reaccionar ágilmente a las demandas del cliente (pp. 6-8).

2.1.1.6.2. Extreme Programming (XP)

Según Sánchez-Hernández et al. (2020), la metodología ágil llamada Programación Extrema (XP) reúne varias prácticas que se centran en cómo se diseña, se prueba y se entrega el *software* mientras se está desarrollando. Esta metodología se basa en las ideas principales del enfoque ágil y fomenta que el cliente y el equipo de desarrollo de *software* trabajen de la mano en todo momento. Algo que la distingue es que se hacen reuniones seguidas, donde cada vez que se avanza, se termina con un grupo particular de tareas ya hechas. Además, la forma en que se administra el trabajo de desarrolladores es dividiendo en periodos de trabajo que duran aproximadamente un mes, lo que ayuda a que se puedan dar resultados que sirven de verdad de forma constante y rápida.

2.1.1.7. Autenticación y seguridad

Chapple (2016), en su libro “*Access Control, Authentication and Public Key*” explica que garantizar la confidencialidad de los usuarios requiere mecanismos sólidos para el acceso y autenticación al sistema a través de contraseñas, biometrías y roles de usuarios. Expresa que el control de acceso se define como tecnologías utilizadas y las autenticaciones es el proceso mediante el cual el sistema verifica la identidad del usuario, de igual manera indaga que para proteger los datos sensibles de una empresa se debe mantener el uso de estos mecanismos manteniendo estructuras seguras y confiables en entornos corporativos.

2.1.1.7.1. JSON WEB Token (JWT)

De acuerdo con Gunawan y Rahmatulloh (2019), el uso de *JSON Web Token (JWT)* es un mecanismo fiable para la autenticación para aplicaciones web, esto mejora la comunicación entre sistemas de distintas plataformas. *JWT* es un *token* dividido en *header-payload-signatura*, que permite autenticar usuarios sin mantener sesiones abiertas. Los autores demuestran que mediante *JWT* es accesible implementar sistemas escalables y compatibles con aplicaciones *web* y móviles, es una solución eficaz para mantener la seguridad en el acceso de los usuarios. Sin embargo, Gunawan y Rahmatulloh (2019) recalcan que se debe reforzar el almacenamiento y algoritmos de cifrado para la protección de los clientes.

2.1.1.7.2. Acceso por roles

Refiriéndose a la seguridad de datos Chapple (2016), destaca el control de acceso por roles (*RBAC*) como una opción eficaz para administrar permisos dentro de una organización, ya que permite conceder privilegios a módulos en específico para cada usuario según sus funciones laborales. Además, fortalece la seguridad en accesos no autorizados, de este modo las organizaciones pueden llevar un control de los movimientos realizados en la plataforma encontrando responsables en caso de errores.

2.1.2. Machine learning

En el ámbito de la inteligencia artificial, Hinestroza (2018), plantea que el *machine learning* son herramientas digitales, cuyo objetivo es mejorar el análisis e interpretación de datos. Algunos sectores como la salud, servicios y transporte han encontrado en esta rama proveniente de la inteligencia artificial la solución a inconvenientes de manera eficaz. Por otra parte, esta materia promete agilizar procesos relacionados a la medicina, robótica y mecánica sin la necesidad de la intervención humana. (pp. 8-12). El *machine learning* tiene la capacidad de resolver situaciones de manera independiente, basándose de un análisis de los datos ingresados, es decir entre más información posea mejores resultados tendrá. Además, para un correcto análisis se usan algoritmos que se generan a partir de otros datos pero que justifiquen las necesidades (Juárez, G. 2017).

De acuerdo con Patil et al. (2023), el *machine learning* es una ayuda inteligente y eficiente para el mundo farmacéutico, ayuda a los científicos encontrar soluciones medicinales mucho más rápidas, comprendiéndose en pequeños detalles como las moléculas. De la misma manera es una fuente confiable para elegir la dosis correcta de un medicamento, emitir una predicción de posibles reacciones y es una red de seguridad que detecta rápidamente si existe algún problema con los medicamentos. Además, debido a su aprendizaje con experiencia ha podido llegar a personalizar los tratamientos de un paciente con el respaldo de la información proporcionada por el usuario.

2.1.2.1. Tipos de aprendizaje

Para Sandoval (2018), el *machine learning* es un campo de la *AI* que elabora algoritmos de predicción con la aptitud de aprender sin tener que programarlos previamente. Funciona bajo alimentación de información de algoritmos en gran escala para que entienda que decisión tomar en cada uno de los casos, existen distintos tipos de aprendizajes: aprendizaje supervisado, no supervisado y por refuerzo (p.37).

En base a las investigaciones de Rojas (2018), el aprendizaje automático (*machine learning*) tiene dos enfoques fundamentales. El aprendizaje supervisado que consiste en enseñarle a un algoritmo la manera de actuar en base a datos de los que ya conoce su respuesta, es decir identificar imágenes que ya están clasificadas con etiquetas dando resultados como “es” o “no es”. Por otro lado, el aprendizaje no supervisado funciona alimentando el algoritmo para análisis de datos a gran escala, básicamente puede elegir un conjunto de grupos con las mismas características y asociarlos a la clase que pertenecen de manera global. Existen varios ejemplos de aprendizaje automático, mencionando a *Deep learning* que utiliza cálculos y características abstractas de alto nivel (p.4).

2.1.2.1.1. Aprendizaje supervisado

Díaz (2021), sostiene que el *machine learning* engloba un conjunto de enfoques que se clasifican en distintas categorías, una de estas es el aprendizaje supervisado, el mismo que tiene bases en investigaciones realizadas en 1959 por Arthur Samuel. El autor recalca este tipo de aprendizaje como una estructura orientada a la supervisión y enseñanza, para permitir que la computadora aprenda a reaccionar al ingreso de cierto tipo de datos. Además, Díaz menciona que de las características más relevantes del aprendizaje supervisado es el uso de datos que ya fueron clasificados, es decir ya existen patrones y predicciones con ejemplos basados concretamente a una situación.

Por otra parte, Sandoval (2018), indaga en que el aprendizaje supervisado permite al desarrollador darle preguntas que se conocen como “características” y respuestas conocidas como “etiquetas”, de esta manera el algoritmo ensaya una posible predicción con fundamentos en lo practicado. Para este tipo de aprendizajes existen dos algoritmos de entrenamiento: clasificación en donde se espera que el algoritmo encuentre patrones que le permitan distinguir los datos ingresados y los clasifique de acorde al tipo de clase que pertenecen y en el algoritmo de regresión se espera un valor entero (número), es decir no busca clasificar en un grupo, al contrario, devuelve un valor específico (p.37).

2.1.2.1.2. Aprendizaje no supervisado

Sandoval (2018), concreta en la idea que el aprendizaje supervisado solo se le comparte características al algoritmo, no etiquetas. Es decir, se espera que reconozca las características dadas y de acorde a eso el algoritmo asuma los datos que pertenecen al mismo grupo. De acuerdo a la publicación realizada por IBM (2023), el aprendizaje no supervisado forma parte de una subcategoría en el campo del *machine learning*, esta categoría se caracteriza por el uso de datos no etiquetados. A diferencia del aprendizaje supervisado, este tipo de enfoque no cuenta con respuestas predeterminadas, por lo que el algoritmo debe descubrir de manera autónoma en base a las estructuras de datos que se ingresen. Además, es útil en situaciones en las que no se cuenta con información categorizada.

2.1.2.1.3. Aprendizaje por refuerzo

Kober y Peters (2013), enfocan al aprendizaje por refuerzo como una excelente opción para el mundo de la robótica, donde es difícil diseñar manualmente comportamientos esperados. En su investigación destacan la eficiencia del aprendizaje por refuerzos, ya que permite a los robots interactuar con el entorno, modelando movimientos forzando al robot a realizar tareas específicas.

De igual manera Rojas (2020), expone que el aprendizaje por refuerzo es una clase de aprendizaje en el que no hay que capacitar con datos clasificados, más bien el algoritmo aprende de manera autónoma con el sistema, donde no hay información clasificada, etiquetas o respuestas esperadas. Es decir, el aprendizaje por refuerzo exige al modelo reaccionar a sucesos inesperados, reforzándolos con problemas cotidianos (p.4).

2.1.2.2. Algoritmos de machine learning

Rojas (2020), afirma que existen técnicas que permiten al *machine learning* identificar instrucciones y hacer predicciones como, por ejemplo: Regresión lineal, árboles de

decisiones, redes neuronales y cadenas de *Markov*. Indaga también, en que el *ML* no es una auto programación al contrario es un proceso de estrategias, aprendizajes de datos y experiencia. El *ML* se pueden encontrar en aplicaciones de uso común como las redes sociales como: *Youtube*, esto se lo conoce como “Recomendaciones” (pp.587-588).

2.1.2.2.1. Regresión Logística

Becerra y Leguizamón (2024), definen la regresión logística como técnica estadística empleada en el ámbito del *machine learning* dentro del aprendizaje supervisado. Su principal objetivo es realizar tareas de clasificación, construyendo modelos que a partir de variables independientes estimen la probabilidad de que exista un fenómeno determinado. De igual manera presentan la estimación de probabilidad de acertar o fallar la predicción esperada, los autores afirman que, en términos probabilísticos un modelo de regresión logística correcto debería predecir que cuando los datos están alejados del objetivo la probabilidad de generar predicciones será menor (p.73-74).

Por otro parte, las tareas de regresión se enfocan en estimar la relación entre las variables de entrada y de salida de tipo continuo, un punto de partida común es la regresión lineal que se usa para determinar el valor de una variable en función de otra variable explicativa.

2.1.2.2.2. Redes Neuronales

Matich (2001), define a las redes neuronales como maneras de estimular características particulares de los seres humanos en un computador, como la capacidad de pensar, reaccionar o emitir acciones a distintos tipos de situaciones. De manera más clara; todos los procesos de un cuerpo humano se relacionan entre si con la actividad de las neuronas, aunque estas células pueden carecer de estructuras complejas, su verdadero poder es cuando se unifican y trabajan en conjunto, las redes neuronales buscan interpretar las mismas actividades, pero en máquinas computacionales. Además, Matich recalca que la capacidad de los seres humanos para el aprendizaje es la base en la que cual se desarrollan las redes neuronales, es decir adquirir la habilidad para resolver problemas (pp. 7-8).

2.1.2.2.3. Árbol de decisiones

De acuerdo a Rivero (2022), el árbol de decisiones es una estructura jerárquica similar a un diagrama de flujo, en ese formato los nodos inferiores figuran las características del conjunto de datos, mientras que las ramas representan las respuestas. Cada hoja del árbol figura una posible salida de decisión y el nodo principal recibe el nombre del nodo raíz, ya que cada división contiene un grupo de reglas y atributos que permiten recorrer una clase. El proceso de aprendizaje basado en árbol de decisiones consiste en dividir recursivamente la información según el valor del atributo, con el objetivo de aplicar datos externos que son utilizados para el entrenamiento, dando como resultado la clasificación según patrones ingresados (p.41).

2.1.2.3. Aplicaciones de Machine Learning

En la investigación realizada por Sandoval (2018), se declara que la utilización de los algoritmos de *machine learning* son necesarios para la predicción de posibles errores en el ámbito empresarial debido a la mayor disponibilidad de datos y tecnología. Estos elementos permiten que los algoritmos aprendan por si solos, sin necesidad de ser programados, lo que facilita su “autoconfiguración”, finaliza en que *ML* es usado en distintos ámbitos, pero su interés radica en su faceta predictiva (p.37).

2.1.2.3.1. Predicciones

Para Sandoval (2018), el machine learning es técnica que permite reconocer patrones en miles de datos individuales para asociarlos de manera predictiva, es una de las características más destacable de la inteligencia artificial. Además, son sistemas que interpretan de manera automática en base a la experiencia y datos ingresados, es decir no solo hace que tenga valores, si no que incluye lógica en tendencias. Esto permite realizar predicciones que se definen como consecuencias a futuro en distintos campos como la salud, contabilidad, incluso en el mundo empresarial.

2.1.2.3.2. Lenguaje Natural (NLP)

De acuerdo con Chaves y Zuleta (2014), el NLP en el área de la computación permite que el computador procese el texto por su definición al contario de tomarlo como archivo de máquina (binario). En la actualidad los sistemas que originan el procesamiento del lenguaje facilitan que texto no se procesa definidamente si no que se modifique el texto de manera lógica preservando características importantes, finalmente da como resultados la interpretación al lenguaje natural (p.3).

Hernández y Gómez (2013), presentan un panorama general sobre el Procesamiento de Lenguaje Natural en distintos ámbitos, destacan que el NLP permite transformar grandes volúmenes de textos en datos estructurados haciendo posible el análisis en áreas como recuperación o extracción de datos, generación de resúmenes, incluso análisis de sentimientos. Dado el crecimiento de información digital de las organizaciones es necesario contar con herramientas como NLP para comprender grandes volúmenes de datos, permitiendo categorizar documentos, clasificar categorías y etiquetar de manera automática contenidos (p.87).

2.1.3. Historial médico

Para Guzmán y Arias (2012), el historial médico es la relación entre la vida de una persona y los eventos que han ocurrido en ella. En la historia clínica existe información privada y real, pues el paciente sabe que si existe alguna distorsión de datos puede poner en peligro hasta su propia vida, además se registran datos familiares en caso de emergencia. Los autores indagan en que, de acuerdo a los conocimientos presentes, el doctor puede ubicar la dolencia del enfermo en ciertos veredictos médicos, asociando o descartando enfermedades, es importante saber que una buena gestión del historial médico permite llevar con claridad la recomendación de medicinas incluso controlar reacciones a los tratamientos recomendados (pp.15-17).

2.1.3.1. Gestión de historial médico

Escobar et al. (2019), en su investigación para la Revista Cubana de medicina explican la necesidad de un historial médico para un paciente, porque permiten emitir un diagnóstico preciso, evita en gran porcentaje riesgos médicos como las alergias, optimiza la gestión de tratamientos y es clave para la prevención de enfermedades. Aunque no nombran el uso de herramientas tecnológicas y se enfocan más en el ámbito medicinal, si aseguran que la actualización de los historiales médicos es base fundamental para tratar enfermedades y gestionar recetas médicas con menos probabilidades a fallos.

2.1.3.1.1. *Historial médico digital y físico*

Como bien destacan Haz et al. (2017), el uso de programas de gestión en las clínicas veterinarias es clave para hacer más ágil la atención y tener siempre a mano el historial médico de los animales. Esta modernización no solo agiliza la consulta de datos, sino que también optimiza la organización interna, minimiza errores en el diagnóstico y simplifica la programación de terapias y revisiones. Además, al guardar toda la información en un mismo lugar seguro, se mejora la constancia en el cuidado de las mascotas y se facilita la comunicación entre los veterinarios. En esta línea, tener un software de gestión hecho a medida para el sector veterinario ayuda mucho a mejorar el trabajo diario y la calidad del servicio. La adopción de estas herramientas digitales es también una respuesta a la necesidad, cada vez mayor, de los dueños de mascotas de recibir una atención más profesional, rápida y de confianza.

Según Cruz et al. (2012), el historial médico de un paciente es una relación entre datos actuales y posibles diagnósticos, se presenta en maneras digitales cuando se relacionan entre hospitales generalmente en la salud pública se usan estos métodos, sin embargo, la antigüedad de un historial médico físico sigue predominando en parroquias rurales. Además, aportar datos clínicos suficientes indica argumentos complementarios para indagar entre posibles enfermedades, exámenes o tratamientos (p.424).

2.1.3.2. Actualización de historiales médicos

Blaz et al. (2023), mencionan en su trabajo de investigación la creación de plataformas que proporcionen una herramienta digital respecto al cuidado y administración del historial clínico en centros veterinarios ubicados en Perú-Lima. La misma investigación que indaga sobre la importancia de la actualización de historiales médicos, ya que existen centros médicos que, si cuentan con un registro de ellos, pero de manera manual o en formato físico y herramientas como *Word* y *Excel*, donde la actualización de la información requiere un proceso más tardado, existiendo más posibilidades de errores, recalcando que este tipo de errores puede afectar directamente la salud de la mascota (p.23).

2.1.3.2.1. Frecuencia de ingreso de datos

Mantener al día la información en el historial clínico de una mascota es crucial para asegurar una atención veterinaria de calidad y sin interrupciones. En la práctica habitual, la actualización de estos datos tiende a ser irregular, limitándose a las visitas al veterinario, lo que a menudo resulta en historiales incompletos o que no reflejan la realidad actual del paciente. No obstante, al implementar soluciones digitales como las que ofrece Izipet, se fomenta un registro de datos continuo y ordenado. Esto permite a los veterinarios añadir información al instante después de cada revisión, tratamiento o procedimiento. Esta costumbre no solo hace que el historial sea más exacto, sino que también ayuda a tomar mejores decisiones clínicas y disminuye la posibilidad de errores por falta de datos recientes.

2.1.3.3. Diagnóstico Médico

La Organización Mundial de la Salud (2024), define a los diagnósticos médicos como como instrumentos significativos que permiten al personal sanitario de un hospital tratar enfermedades, los mismos que dan paso a la gestión de los medicamentos para tratamientos en los pacientes. Los servicios de diagnósticos son esenciales para la prevención de errores y crucial para la vigilancia de enfermedades que pueden ser transmisibles, por lo tanto, el diagnostico seguro garantiza los sistemas de salud del mundo.

De igual manera Rodríguez (2021), explica que los diagnósticos médicos son utilizados para la gestión de medicina en un paciente, la detección de enfermedades y la comunión entre el doctor y el personal de enfermería. Sin embargo, menciona que los diagnósticos médicos en cualquier ámbito aumentan su valor siempre y cuando sean vistos como ruta principal de información para tratar la salud de los pacientes.

2.1.3.3.1. Diagnóstico Presuntivo y Definitivo

Frómata et al. (2017), en su investigación Clínica Médica 2, aborda la importancia de mantener la diferencia entre un diagnóstico presuntivo y uno definitivo como parte fundamental de la gestión de un historial clínico. Aunque podrían ser intuitivos por sus nombres, el diagnóstico presuntivo se refiere a la idea inicial que el médico pueda tener de acorde a los síntomas que el paciente le indique, este diagnóstico orienta la toma de decisiones acerca de exámenes y medicamentos temporales. Por otro lado, el diagnóstico definitivo es el que se establece a partir de una valoración de pruebas para descartar enfermedades, las pruebas son las respuestas lógicas que requiere el médico para emitir una respuesta definitiva (pp.255,257).

2.1.3.3.2. Gestión de tratamientos

Según Fajardo y Robledo (2018), llevar un registro de los tratamientos dentro del historial clínico de una mascota ayuda a tener un control preciso y ordenado de las terapias que recibe. Esta sección del historial incluye información clave como el diagnóstico, el tipo de tratamiento utilizado, la dosis administrada, la frecuencia, la duración del tratamiento y las reacciones observadas. Al pasar este proceso a formato digital, se facilita la administración de medicamentos y se hace un mejor seguimiento de la evolución del animal, permitiendo configurar recordatorios para las próximas dosis, revisiones o valoraciones. Además, el acceso rápido a estos datos por diferentes veterinarios en una red conectada mejora la coordinación de la atención, la personalización del cuidado y evita repeticiones o equivocaciones en la medicación (pp.73,77).

2.1.3.4. Componentes de historial médico (clínico y laboratorio)

De acuerdo con Brun y Méndez (2017), la elaboración de un historial médico tanto clínico como de laboratorio garantiza una atención integral y segura al paciente. Cada componente como datos de identificación, datos familiares y sintomatologías permite al profesional de la salud un análisis lógico ante los padecimientos de la persona. El historial médico clínico por su parte se encarga de los padecimientos antiguos y actuales que ha tenido el paciente, tratamientos, operaciones y datos relevantes para una rápida atención. Sin embargo, el historial médico de laboratorio es el conjunto de evaluaciones que se ha realizado una persona para descartar enfermedades, este elemento no solo orienta la toma de decisiones del médico, sino que también es un respaldo legal y científico para la designación de un tratamiento y comunicación entre el personal médico (p.1200).

2.1.3.5. Clasificación de diagnóstico

Los sistemas de codificación para la medicina según Quispe et al. (2019), son técnicas métricas que permiten clasificar las sintomatologías de un paciente de forma ordenada ya que permiten traducir la información clínica en códigos estandarizados que son utilizados a nivel local e internacional. La codificación de diagnósticos permite el análisis y comparación entre datos médicos, lo que es esencial para la detección de pandemias y epidemias a tiempo. La clasificación de enfermedades mejora la calidad del historial médico basando la evidencia de un pronóstico en hechos reales y datos confiables.

2.1.3.5.1. CIE-10

La Organización Mundial de la Salud (1990), establece el sistema estandarizado CIE-10 que permite codificar enfermedades y síntomas mediante códigos alfanuméricos. Kin Wah Fung et al. (2024), en su investigación menciona que el principal objetivo de la Clasificación Internacional de Enfermedades en la 10ma revisión (CIE-10) es facilitar la identificación de datos médicos y epidemiológicos a nivel mundial, lo que aporta organización en la salud pública y privada. Además, contribuye la planificación de recursos médicos en los hospitales y garantiza la toma de decisiones médicas y administrativas.

2.1.3.5.2. SNOMED CT

De la misma forma Kin Wah Fung et al. (2024), definen a la Nomenclatura Sistemática de la Medicina (SNOMED CT) como una terminología utilizada en el área de la salud para presentar de forma estructurada los conceptos utilizados en la atención médica. SNOMED CT es reconocida por ser la clasificación de información clínica más completa a diferencia de CIE-10, que está orientada a clasificaciones estadísticas. SNOMED CT permite codificar la información médica de los pacientes con detalles semánticos, con diagnósticos, procedimientos. Esta terminología mejora la eficiencia en la comunicación entre sistemas de información en la salud, precisando historiales médicos electrónicos; es generalmente implementado en entornos con grandes volúmenes de datos clínicos. Además, ayuda al intercambio seguro entre organizaciones sanitarias (p.1635).

2.2. Predicción científica

H0: La aplicación *web* con *Machine Learning* no incide significativamente en el fortalecimiento del proceso de gestión de historiales médicos en el Hospital Veterinario Santa Martha, del cantón Santo Domingo.

H1: La aplicación *web* con *Machine Learning* incide significativamente en el fortalecimiento del proceso de gestión de historiales médicos en el Hospital Veterinario Santa Martha, del cantón Santo Domingo.

3. METODOLOGÍA

3.1. Enfoque, alcance, diseño y tipo de investigación

En este trabajo de titulación, se optó por utilizar un enfoque cuantitativo en la investigación. De acuerdo con Hernández-Sampieri y Mendoza-Torres (2018), este tipo de enfoque se define como un proceso riguroso, secuencial y dirigido a pruebas de hipótesis, diseñado para medir información mediante datos numéricos (p.6). Además, el enfoque cuantitativo posee etapas estructuradas para el análisis estadístico con el objetivo de generalizar resultados en muestras numéricas. A través de la utilización de este enfoque, se procura estudiar los datos obtenidos de la población de muestra estudiada y el procedimiento actual para la gestión de historiales médicos.

Para continuar con la misma línea, el alcance de una investigación determina el grado de profundidad con el que se estudia el objeto de análisis. En este sentido Hernández-Sampieri y Mendoza-Torres (2018) señala que, tradicionalmente existen cuatro tipos de alcance: exploratorio, descriptivo, correlacional y explicativo (p.105). Sin embargo, Supo (2025) presenta 2 niveles de alcance adicional a los planteados: predictivo y aplicativo, defiende también que cada nivel de investigación mencionado tiene un objetivo importante dentro del estudio, pero se mide según su alcance, finalidad o enfoque (p.129). En consecuencia, la presente investigación se desarrolló bajo un nivel de alcance aplicativo, ya que su objetivo principal es resolver problemáticas mediante la implementación de una herramienta, en este caso la aplicación *web* para la correcta gestión de historiales médicos de las mascotas.

De igual manera, en palabras de Bernal (2016), un diseño de investigación depende del tipo de estudio que se desea realizar y de la hipótesis que busca comprobar, el término “diseño” se utiliza mayormente en investigaciones experimentales, que tienen como objetivo demostrar que los cambios que se le hacen a una variable influyen en las demás variables dependientes. Además, el autor menciona tres categorías de diseños de experimentos para investigaciones: Preexperimentales, en donde el investigador genera un control limitado

sobre las variables independientes y dependientes, en este tipo de estudios no se realiza una asignación aleatoria de grupos, se define un único grupo para tomar medidas antes y después de la investigación. Por otro lado, los diseños cuasiexperimentales permiten al investigador tener control limitado sobre las variables externas, además los participantes pueden ser asignados a dos o más grupos de estudio de manera aleatoria, pero con las mismas características. Por último, el diseño experimental verdadero, exige sujetos participantes de dos o más grupos (experimental y de control) y se ejerce límites sobre todas las variables y los objetos de medición (pp. 194-196).

De la misma forma, Ramos (2021) señala que el diseño preexperimental se caracteriza por designar un único grupo de experimentación, en este diseño las variables deben ser medidas de la misma forma en dos momentos: antes y después de la intervención, mediante la aplicación de instrumentos de evaluación conocidos como *pretest* y *posttest* para medir los efectos producidos por la investigación (p.4). De acuerdo a lo mencionado el presente trabajo de titulación optó por utilizar el diseño preexperimental, ya que se trabajó con el mismo grupo de encuestados en las dos fases de investigación (*pretest* y *posttest*).

Por otro lado, Hernández-Sampieri y Mendoza-Torres (2018) establece que la investigación cumple dos propósitos fundamentales: la primera es producir conocimiento y teorías denominada investigación básica, y la segunda es resolver problemas conocida como investigación aplicada (p. xxxiii). Por lo cual, Lozada (2014) menciona que, la investigación aplicada permite generar los conceptos, características y requerimientos del usuario final y el proceso de materialización de los conceptos planteados para realizar el producto a entregar (p.35). Dentro del trabajo de titulación presentado, se optó por la utilización del tipo de investigación aplicada, cuyo objetivo es identificar problemáticas relacionadas con la gestión de historiales médicos. Con el fin de planificar una propuesta eficiente para el manejo de historias clínicas en el Hospital Veterinario Santa Martha, se integraron herramientas tecnológicas como *Machine Learning*.

Por último, se integró la investigación de campo, en la que se utilizaron técnicas de recolección de datos. Se visitó el Hospital Veterinario donde se obtuvo información importante por parte del Propietario sobre la gestión de los historiales médicos de mascotas y servicio brindado a los pacientes.

3.2. Unidades de análisis

Hernández et al. (2014), comparte que las técnicas estadísticas proporcionan herramientas confiables para la organización de una investigación considerando el tipo de muestreo a usar, basándose en los posibles encuestados (p.101). Además, Memon et al. (2025) expresa que el enfoque de muestreo por conveniencia permite seleccionar participantes que están accesibles durante el periodo de estudio, lo que beneficia al investigador cuando se tiene limitaciones en tiempo, recursos o acceso a las personas a encuestar. Aunque no permite generalizar los resultados es válido para estudios de investigación, es utilizado para analizar tendencias y deducir conclusiones sobre la percepción de la población (p.11).

En la reunión de planificación con el propietario informó que el Hospital Veterinario posee un registro de más de mil clientes aproximadamente. Por esa razón, en el trabajo de titulación presentando se consideró aplicar un método de estudio no probabilístico denominado muestreo por conveniencia, en el que se tomó una muestra de 50 clientes que asistieron al hospital en el tiempo que se realizó la encuesta, los cuales se contemplan en la Tabla 1.

Tabla 1. Muestra encuestada

Hospital Veterinario Santa Martha		
Categoría	Clasificación	Cantidad
Clientes	Hombres	22
	Mujeres	28
Total		50

3.3. Técnicas e instrumentos de investigación

Para la recolección de datos se utilizó como técnica a la encuesta, con el propósito de recaudar información confiable para ser analizada posteriormente. Para Hernández et al. (2014), la toma de datos para procedimientos sistemáticos es importante para la precisión de la investigación, el enfoque cuantitativo se destaca por su precisión en datos numéricos (p.4). Es así que se llegó a la conclusión de usar *Google Forms* como instrumento de investigación y encuestas con preguntas de opción múltiples a los clientes, con relación a la ruta de investigación planteada para el Hospital Veterinario.

3.4. Técnicas de análisis de datos

En palabras de Borrego (2008), la estadística descriptiva se encarga de organizar y clasificar las cifras recolectadas, para ello se utilizan tablas y representaciones gráficas de manera que la interpretación de los datos pueda analizarse de mejor manera. Además, en este tipo de estadística no se usa cálculos de probabilidad, si no, se encarga de generar conclusiones directamente de la información encontrada. Por otro lado, la autora menciona a la estadística inferencial como la técnica que permite tener conclusiones confiables acerca del fenómeno estudiado, su propósito es formular estimaciones acerca de una población a partir de los resultados recolectados de la muestra escogida (pp. 2-4).

Por lo antes mencionado, el análisis de la información tuvo comienzo con el uso de la estadística descriptiva para organizar, resumir y mostrar los resultados de manera comprensible. Para ello, se emplearon técnicas como la distribución de frecuencia relativas (porcentajes) y el análisis descriptivo mediante gráficos de barras, usando *Microsoft Excel* como herramienta de apoyo. Posteriormente, para procesar la información obtenida de las encuestas, se aplicó la estadística inferencial utilizando como técnica la regresión logística binaria, mediante el uso de la herramienta *IBM SPSS* que permitió la comparación de los resultados en la fase del *PreTest* y el *PosTest*.

3.5. Operacionalización de las variables

Tabla 2. Aplicación Web - Variable Independiente

Conceptualización	Dimensión	Indicadores	Preguntas	Herramientas
Hernández-Rueda et al. (2022), afirma que una aplicación <i>web</i> se define como un <i>software</i> al que se accede por medio de un navegador, destaca que las aplicaciones <i>web</i> ofrecen soluciones de manera rápida para la gestión de información permitiendo a los usuarios acceder desde cualquier dispositivo que cuente con internet. Según Tim Berners-Lee creador del <i>World Wide Web</i> , describe las aplicaciones <i>web</i> como programas que utilizan el protocolo <i>http</i> , para interactuar con los usuarios mediante navegadores <i>web</i> , permitiendo el acceso de manera remota a servicios o datos de los mismos.	Herramientas de Desarrollo	Lenguaje de programación	¿Considera que el manejo de aplicaciones <i>web</i> es una alternativa más fácil de acceder a la información de una mascota?	Reunión de planificación con el propietario
		Frameworks de desarrollo	¿Cree necesario que una aplicación <i>web</i> cuente con un diseño entendible?	Encuesta a clientes
			¿Estás de acuerdo en visualizar los datos médicos de su mascota desde un dispositivo con acceso a internet?	
	Base de datos	Base de datos relacional (My SQL)	¿Qué tan importante es para usted digitalizar el historial clínico de su mascota?	Encuesta a clientes
		NoSQL	¿Cómo se resguarda la información de los historiales médicos en el hospital veterinario?	Reunión de planificación con el propietario
	Tecnologías de acceso	Cloud Computing	¿Con qué frecuencia tiene acceso a internet?	Encuesta a clientes
		HTTP/REST	¿Con qué frecuencia utiliza navegadores <i>web</i> , como <i>Google Chrome</i> , <i>Safari</i> o <i>Firefox</i> ?	Encuesta a clientes
		WLAN / WAN	¿Está familiarizado con el uso de tecnologías de computación en la nube?	Reunión de planificación con el propietario
	Metodología de desarrollo	Método ágil (SCRUM)	¿Cuál es el procedimiento para agendar citas en el hospital?	Reunión de planificación con el propietario
		Extreme Programming (XP)	¿Qué tan importante es para usted que una aplicación <i>web</i> le permita obtener información de la salud de su mascota de manera organizada y segura?	Encuesta a clientes
	Autenticación y Seguridad	JWT - OAuth2	¿Con qué frecuencia está informado/a sobre citas, vacunas o tratamientos pendientes?	Reunión de planificación con el propietario
		Control de acceso por roles	¿Considera importante agilizar el proceso de registro de pacientes?	
			¿Considera importante que los usuarios tengan roles definidos?	Reunión de planificación con el propietario
			¿Con qué frecuencia actualiza la información médica de su mascota?	Encuesta a clientes

Tabla 3. Machine Learning- Variable Independiente

Conceptualización	Dimensión	Indicadores	Preguntas	Herramientas
El aprendizaje automático, o también conocido como <i>Machine Learning (ML)</i> , la podemos denominar como la ciencia que permite a los humanos diseñar algoritmos y enseñar a las computadoras aprender patrones de grandes cantidades de datos, para poder usar estos patrones para la toma de decisiones o predicciones automáticas (Pan, F. 2023).	Tipos de aprendizaje	Aprendizaje supervisado	¿Considera eficaz que una aplicación web tenga la capacidad de hacer posibles predicciones sobre la salud de su mascota basándose en el historial clínico de la misma?	Encuesta a clientes
		Aprendizaje no supervisado	¿Considera que hacer una relación entre síntomas y enfermedades podría ayudar a un diagnóstico temprano de la mascota?	Reunión de planificación con el propietario
		Aprendizaje por refuerzo		
	Algoritmos de <i>machine learning</i>	Regresión Logística Redes neuronales	¿Cuál es su opinión respecto al uso de redes neuronales en el diagnóstico temprano la mascota?	Reunión de planificación con el propietario
		Árbol de decisiones	¿El hospital cuenta con un proceso para predicción de enfermedades?	Reunión de planificación con el propietario
	Predicciones		¿Le gustaría poder predecir comportamientos o detectar anomalías en la salud de una mascota?	
	Aplicaciones de <i>Machine Learning</i>	Lenguaje Natural (NLP)	¿Qué nivel de confianza le genera el uso de inteligencia artificial para predicción de enfermedades?	Reunión de planificación con el propietario

Tabla 4. Historial Médico - Variable Dependiente

Conceptualización	Dimensión	Indicadores	Preguntas	Herramientas	
Según ACLAM <i>Medical Records Committee</i> et al (2007), las historias clínicas veterinarias son consideradas un elemento clave en un programa de atención veterinaria adecuada para animales utilizados en investigación, enseñanzas y pruebas.	Gestión de historial médico	Historial médico digital	¿El hospital cuenta con un sistema de gestión para historiales médicos de los pacientes?	Reunión de planificación con el propietario	
			¿Qué tan importante es para usted acceder con facilidad al historial médico de sus mascotas?	Encuesta a clientes	
		¿Considera seguro la administración del historial médico de su mascota?	Encuesta a clientes		
		Historial médico físico		¿Con qué frecuencia accede a información médica de su mascota sin ayuda de personal médico?	Encuesta a clientes
	¿Con qué frecuencia ha experimentado problemas para acceder al historial médico de su mascota?	Encuesta a clientes			
	Actualización de historiales médicos		Frecuencia de ingreso de datos	¿Cree que digitalizar los historiales médicos de la mascota mejora el servicio del hospital?	Reunión de planificación con el propietario
		¿Con qué frecuencia se actualizan los datos de los pacientes en la veterinaria?		Reunión de planificación con el propietario	
	¿Con qué frecuencia se siente involucrado/a activamente en el seguimiento de la salud de su mascota?	Encuesta a clientes			
	Diagnóstico Médico	Presuntivo	Definitivo	¿De qué forma administra los tratamientos de su mascota?	Encuesta a clientes
				¿Con qué frecuencia le resulta sencillo entender los diagnósticos y tratamientos que recibe su mascota?	Encuesta a clientes
Tratamiento		¿Considera importante llevar un control digital de los tratamientos de su mascota?	Encuesta a clientes		
		¿Qué tan fácil le resulta registrar posibles reacciones a los medicamentos suministrados?		Encuesta a clientes	
¿Considera necesario registrar información de posibles reacciones de los medicamentos suministrados a las mascotas?	Encuesta a clientes				
¿Le gustaría conocer el proceso o reacciones de medicamentos de los tratamientos asignados a las mascotas?		Reunión de planificación con el propietario			
Componentes de historial médico	Clínico Laboratorio	CIE-10	¿Le parece importante poder visualizar el historial de tratamientos que lleven las mascotas?	Reunión de planificación con el propietario	
			SNOMED CT	¿Le gustaría que un sistema cuente con la clasificación de diagnósticos?	Reunión de planificación con el propietario

4. RESULTADOS

4.1. Resultados del primer objetivo específico

Los instrumentos de investigación fueron validados por profesionales en el área de tecnologías de la información, los cuales se visualizan en la tabla 5.

Tabla 5. Profesionales en validación de instrumentos de recolección de datos.

Nombres	Área	Título Académico
Luis Javier Ulloa Meneses	<i>Machine Learning</i>	Mg. <i>Big Data y Data Science</i>
Jonathan Mario Moreno Rivera	Aplicación <i>web</i> y móvil	Mg. Ingeniería de <i>software</i> y sistemas informáticos
Rodolfo Sirilo Córdova Gálvez	Investigación	Mg. en Redes de Comunicaciones

4.1.1. Aplicación de la encuesta a los clientes

4.1.1.1. Presentación de los resultados de la encuesta a los clientes








En base a la recopilación de datos realizada a 50 clientes del hospital veterinario, mediante la aplicación de la encuesta en la fase del *PreTest*, se pudo identificar datos relevantes que sustentan la implementación de una aplicación *web* para la gestión del historial médico de mascotas.









Se pudo evidenciar que, el 78.30% considera entre “muy importante” e “importante” acceder con facilidad al historial médico de sus mascotas. Sin embargo, también se identifican debilidades en la gestión actual, el 63.00% administra los tratamientos de forma manual y solo el 6.50% lo hace digitalmente. Además, el 32.60% rara vez accede a la información médica de su mascota y el 39.10% frecuentemente ha experimentado problemas para acceder al historial médico. En cuanto al seguimiento de la salud de sus mascotas el 52.10% se siente involucrado entre “muy frecuente” y “frecuentemente” sin embargo, existe un 47.9% entre “ocasionalmente o raramente”. Finalmente, aunque el 37.00% califica como “normal” la facilidad para registrar reacciones a medicamentos, un 32.60% lo considera entre “difícil” y “muy difícil”, lo que se deduce a que los clientes consideran poco adecuado el registro de reacciones a tratamientos.

4.1.1.2. Resultados de encuesta dirigida a los clientes

En la tabla 6, se encuentran detallados los resultados obtenidos de la encuesta a los clientes del Hospital Veterinario Santa Martha.

Tabla 6. Resultados de la encuesta a los clientes del *pre test*

N°	Preguntas	Escala y %					Figura
		Muy frecuente	Frecuentemente	Ocasionalmente	Raramente	Nunca	
1	¿Con qué frecuencia tiene acceso a internet?	54,30%	39,10%	2,20%	4,30%	0,00%	
2	¿Qué dispositivo electrónico utiliza regularmente?	87,00%	4,30%	4,30%	4,30%	4,40%	
3	¿Con qué frecuencia utiliza navegadores web, como Google Chrome, Safari o Firefox?	37,00%	37,00%	15,20%	8,70%	2,20%	
4	¿Cree necesario que una aplicación web cuente con un diseño entendible?	39,10%	37,00%	17,00%	4,30%	2,20%	
5	¿Estás de acuerdo en visualizar los datos médicos de su mascota desde un dispositivo con acceso a internet?	37,00%	26,10%	21,70%	13,00%	2,20%	
6	¿Qué tan importante es para usted digitalizar el historial clínico de su mascota?	30,40%	41,30%	21,70%	2,20%	4,30%	
7	¿Qué tan importante es para usted que una aplicación web le permita obtener información de	39,10%	28,30%	15,20%	15,20%	2,20%	

la salud de su mascota de manera organizada y segura?							
8	¿Con qué frecuencia actualiza la información médica de su mascota?	Muy frecuente	Frecuentemente	Ocasionalmente	Raramente	Nunca	
		6,50%	28,30%	28,30%	17,40%	19,60%	
9	¿Con qué frecuencia está informado/a sobre citas, vacunas o tratamientos pendientes de su mascota?	Muy frecuente	Frecuentemente	Ocasionalmente	Raramente	Nunca	
		6,50%	30%	23,90%	28,30%	10,90%	
10	¿Considera eficaz que una aplicación web tenga la capacidad de hacer posibles predicciones sobre la salud de su mascota basándose en el historial clínico de la misma?	Completamente efectivo	Altamente efectivo	Parcialmente efectivo	Poco efectivo	Nada efectivo	
		28,30%	26,10%	21,70%	23,90%	0,00%	
11	¿Qué tan importante es para usted acceder con facilidad al historial médico de sus mascotas?	Muy importante	Importante	Moderadamente importante	Poco importante	No es importante	
		32,60%	45,70%	13,00%	8,70%	0,00%	
12	¿Considera seguro la administración del historial médico de su mascota?	Muy seguro	Seguro	Moderadamente seguro	Poco seguro	Totalmente inseguro	
		23,90%	47,80%	15,20%	10,90%	2,20%	
13	¿Con qué frecuencia accede a información médica de su mascota sin ayuda de personal médico?	Muy frecuente	Frecuentemente	Ocasionalmente	Raramente	Nunca	
		6,50%	20%	23,90%	39,10%	10,90%	
14	¿Con qué frecuencia ha experimentado problemas para acceder al historial médico de su mascota?	Muy frecuente	Frecuentemente	Ocasionalmente	Raramente	Nunca	
		11%	39,10%	21,70%	10,90%	17,40%	
15		Muy frecuente	Frecuentemente	Ocasionalmente	Raramente	Nunca	

	¿Con qué frecuencia le resulta sencillo entender los diagnósticos y tratamientos que recibe su mascota?	8,70%	23,90%	32,60%	26,10%	8,70%	
16	¿De qué forma administra los tratamientos de su mascota?	Manual 63%	Digital 6,50%	No administro 30,40%			
17	¿Con qué frecuencia se siente involucrado/a activamente en el seguimiento de la salud de su mascota?	Muy frecuente 13%	Frecuentemente 39,10%	Ocasionalmente 21,70%	Raramente 26,10%	Nunca 0,00%	
18	¿Considera importante llevar un control digital de los tratamientos de su mascota?	Muy importante 21,70%	Importante 47,80%	Moderadamente Importante 17,40%	Poco Importante 13%	No es importante 0,00%	
19	¿Qué tan fácil le resulta registrar posibles reacciones a los medicamentos suministrados?	Muy fácil 15,20%	Fácil 15,20%	Normal 37%	Difícil 28,30%	Muy difícil 4,30%	
20	¿Considera necesario registrar información de posibles reacciones de los medicamentos suministrados a las mascotas?	Muy necesario 23,90%	Necesario 45,70%	Medianamente necesario 10,90%	Poco necesario 19,60%	Innecesario 0,00%	

4.1.2. Identificación de los procesos en la gestión de historiales médicos de mascotas

Luego de la reunión de planificación con el propietario se pudo identificar los procesos: proceso de registro y verificación, proceso de evaluación médica, proceso de diagnóstico y tratamiento y proceso de actualización del historial médico, esta información consta en el anexo 10.

4.1.2.1. Diagrama BPD del proceso de gestión de historiales médicos de mascotas

De acuerdo con la *OMG (2011)*, el *BPMN 2.0 (Business Process Model and Notation)* es un estándar de modelado de procesos de negocio que representa de forma gráfica los flujos que son parte de un proceso organizacional, incluyendo los tipos de eventos desde su inicio a fin. Este modelo permite la relación entre el diseño y el análisis de los procesos de un negocio. Por tal motivo, *BPMN 2.0* fue utilizado de única manera en este proyecto para gestionar la planificación de los *sprints*, representar de manera visual los procesos de la empresa y servir de modelo para el desarrollo de una aplicación *web* con *machine learning* para la gestión de historial médico de mascotas en el Hospital Veterinario Santa Martha en el Cantón Santo Domingo. Cabe recalcar que *BBP* no fue utilizado como artefacto del enfoque de la investigación.

Figura 4. Diagrama BPD de gestión historial médicos de las mascotas

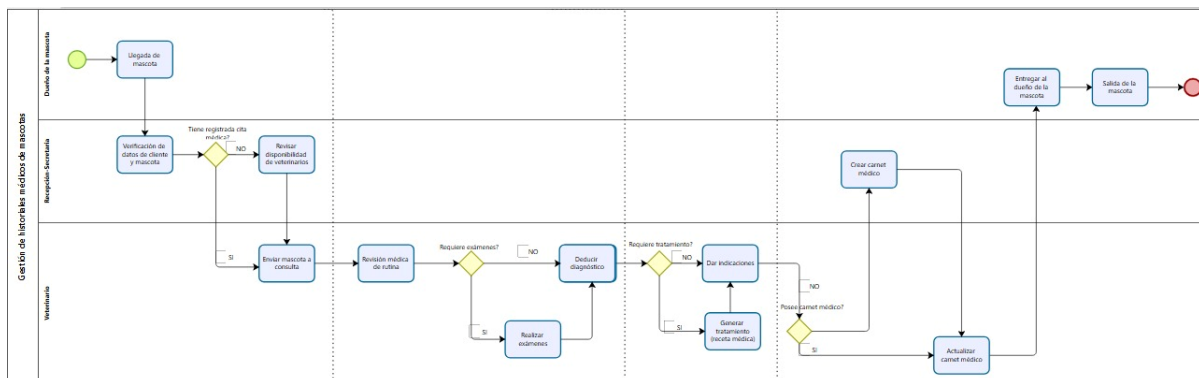


Figura 5. Proceso de registro y verificación

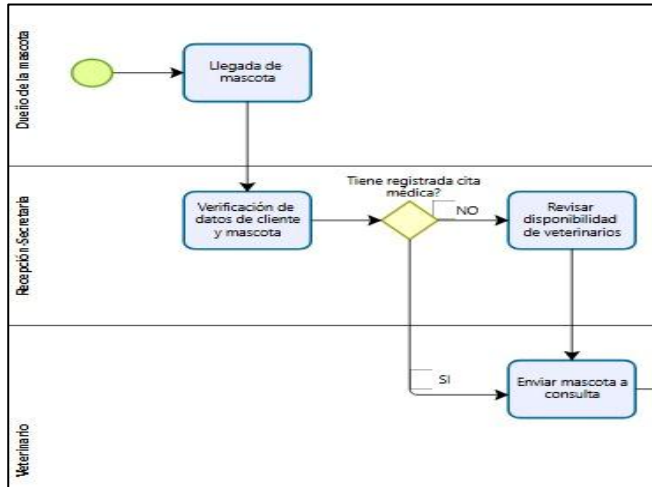


Figura 6. Proceso de evaluación médica

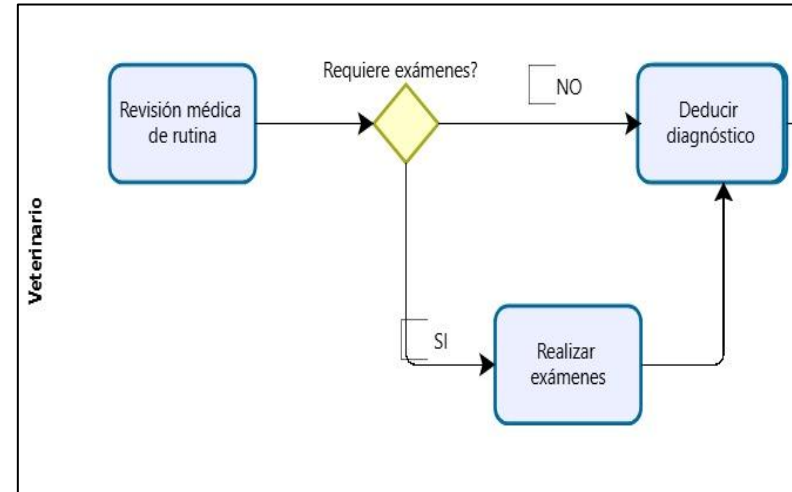


Figura 7. Proceso de diagnóstico y tratamiento

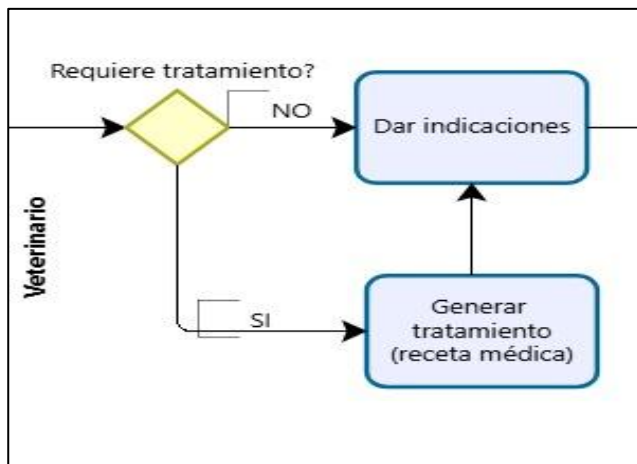
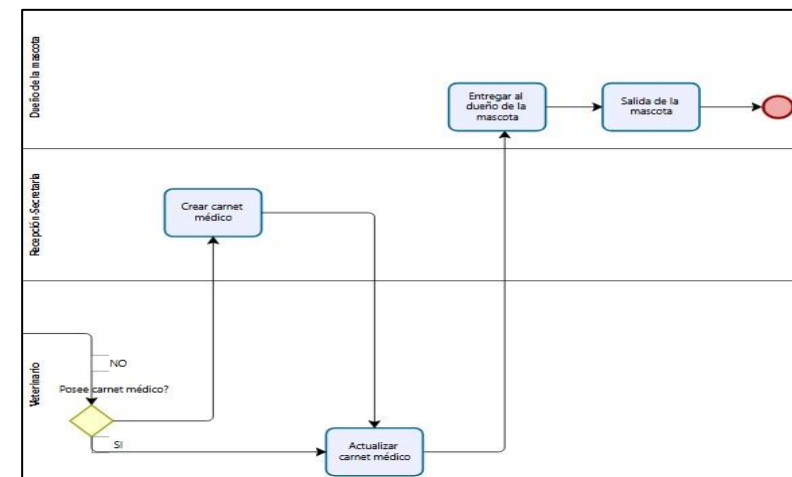


Figura 8. Proceso de Gestión y Actualización del Historial Médico



4.2. Resultado del segundo objetivo específico

Para el desarrollo de la aplicación *web* se utilizaron diversas tecnologías que, permitieron garantizar una arquitectura eficiente, se analizaron *frameworks* para el *backend* y el *frontend*, herramienta de gestión de base de datos, metodología de desarrollo, la arquitectura de la aplicación y los algoritmos de *machine learning*.

4.2.1. Herramientas tecnológicas

La elección adecuada de las herramientas tecnológicas representa la estructura de una aplicación para garantizar el resultado esperado.

4.2.1.1. Framework Frontend

Para establecer las tecnologías que se utilizaron, se realizó una comparación de herramientas. Para la selección del *framework* utilizado en el desarrollo del *frontend*, fue necesario realizar un análisis comparativo entre dos herramientas de desarrollo comúnmente utilizadas para *frontend*. En la tabla 7 se presentan las características de cada una de ellas.

Tabla 7. Características de Angular y Vue.js

Angular	Vue.js
<i>Framework</i> completo dependiente de <i>Google</i>	<i>Framework</i> progresivo para la construcción de interfaces y <i>SPA</i>
Es necesario conocer <i>TypeScript</i> y el ecosistema	Sintaxis simple y documentación clara
Basada en componentes, modulo e inyección de dependencias	Basada en componentes con enfoque progresivo
Rendimiento alto, gracias al <i>Virtual DOM</i> y renderización eficiente	Rendimiento alto, optimizado mediante <i>Virtual DOM</i> y reactividad
Incluye bastantes funcionalidades integradas	Permite incorporar librerías según se necesiten
Ideal para proyectos complejos o integraciones múltiples	Excelente para proyectos que requieren rápida implementación
Arquitectura completa, pero menos intuitiva	Arquitectura amigable para nuevos desarrolladores

Nota: Documentación oficial de: ^aAngular (2024) y ^bVue.js (2024).

Finalmente, para el *frontend* se optó por *Vue.js*, por su fácil aprendizaje y su eficiencia en la construcción de interfaces dinámicas e interactivas. Además, el enfoque progresivo con el que cuenta permitió mantener un código limpio y escalable que se alinean con los objetivos del proyecto.

4.2.1.2. Framework Backend

Con el propósito de mejorar la etapa de desarrollo, se analizaron opciones de *framework* para el *backend*, ya que este incluye la lógica empresarial y funcional de la aplicación. Además, se optó por seleccionar un criterio funcional en el desarrollo *web*, que facilita la implementación y programación de un sistema sólido, seguro y con capacidad de escalar para cumplir las exigencias del usuario. Se evaluaron dos alternativas muy utilizadas dentro de la comunidad de *Python*, que observan en la tabla 8.

Tabla 8. Características de Django y FastAPI

Django	FastAPI
Completo y estructurado	Moderno y asíncronico
Para su aprendizaje es necesario conocer su estructura interna	No requiere conocimientos previos
Menor rendimiento por sobrecarga interna	Alto rendimiento gracias a <i>ASGI</i> y <i>asincronía</i>
Documentación extensa pero detallada	Documentación automática (<i>OpenAPI</i>)
Ideal para aplicaciones empresariales que necesiten múltiples módulos	Enfocada a <i>APIs</i> modernas y aplicaciones rápidas
Comunidad amplia, incluye <i>ORM</i> y panel administrativo	Comunidad orientada en <i>APIs</i>

Nota: Documentación oficial de: ^aDjango (2024) y ^bFastAPI (2024)

Después de realizar el estudio comparativo, se decidió por utilizar *FastAPI* para la programación, ya que es un *framework* de alto rendimiento y actual, adecuado para crear *APIs* usando *Python*. Además, es un instrumento versátil que posee una curva de aprendizaje moderada, fácil y funcional con un rendimiento bastante alto para un *framework* con un marco de trabajo reciente.

4.2.1.3. Base de datos

Para el almacenamiento de información de la aplicación, se evaluaron sistemas gestores de base de datos relacionales de código abierto, puesto que estos ayudaron a gestionar, crear y administrar datos entre la interfaz del usuario y la base de datos. Además, garantizan la seguridad e integridad de información en la aplicación, se investigaron dos opciones bastante utilizadas: *MySQL* y *PostgreSQL*, las dos ofrecen compatibilidad con *Python*, lo que las hace ideal para el proyecto, sus características se visualizan en la tabla 9.

Tabla 9. Características de MySQL y PostgreSQL

MySQL	PostgreSQL
Licencia Pública General (código abierto)	<i>PostgreSQL License</i> (código abierto)
Rendimiento óptimo en operaciones de lectura y consultas rápidas	Útil en operaciones complejas y de alto volumen
Soporte seguro en servidores <i>web</i> y entornos de <i>hosting</i>	Compatible con estándares <i>SQL</i> avanzados y funciones analíticas
Buena para aplicaciones que posee un tráfico controlado	Adecuada para entornos con volúmenes de información bastantes altos
Instalación y administración sencilla	Configuración más avanzada pero flexible
Escalabilidad rápida	Escalabilidad Rápida
Comunidad amplia y con bastante documentación	Comunidad amplia pero más técnica
Excelente mediante librerías como <i>PyMySQL</i> o <i>SQLAlchemy</i>	Total, compatibilidad con librerías <i>ORM</i> modernas
Protección de datos con monitorización	Modelo comparativo con actualizaciones de seguridad

Nota: Documentación oficial de: ^aOracle (2020) y ^bPostgreSQL (2026).

Se seleccionó *MySQL* como sistema gestor de la base de datos por su fácil administración, buen rendimiento en consultas y excelente compatibilidad con servicios *web*. De igual manera su eficiencia da una opción práctica para la implementación de la aplicación con menos probabilidad de errores, estabilidad y escalabilidad a largo plazo.

4.2.1.4. Metodología de Desarrollo

Para la organización en el proceso de desarrollo del proyecto, se analizaron dos metodologías bastante conocidas: *Scrum* y *Extreme programming*. Las dos ayudan a mejorar la productividad y comunicación de los equipos dentro de la programación, pero compiten entre los enfoques técnicos con que trabajan cada una. Además, usando estas herramientas hay más posibilidad de entregar un resultado con excelente calidad, se visualiza en la tabla 10.

Tabla 10. Características de Scrum y Extreme Programming

Scrum	Extreme Programming (XP)
Gestión de proyectos mediante <i>sprints</i>	Enfoque dedicado en la calidad del código
Sprints de 1 a 4 semanas	Avances en periodos cortos de 1 a 3 semanas
Roles: <i>Product Owner</i> , <i>Scrum Master</i> y Equipo de desarrollo	Roles: Desarrolladores, Clientes y <i>Coach</i>
Guiada en <i>backlog</i> y priorización de tareas	Guiada en historias de usuario y pruebas frecuentes
Reuniones programadas y revisiones de <i>sprint</i>	Comunicación constante con el cliente
Revisión al final de cada <i>sprint</i>	Pruebas unitarias de historias avanzadas
Herramientas usadas: <i>Jira</i> , <i>Trello</i> , <i>ClickUp</i>	Herramientas usadas: <i>Git</i> , <i>Test-Driven Development</i> , <i>Pair Programming</i>
Enfoque iterativo e incremental para optimizar la previsibilidad y controlar el riesgo	Adaptabilidad alta, con planificación técnica
Documentación ligera, específica en cada funcionalidad	Prioriza el código legible y probado

Nota: ^aSchwaber y Sutherland (2020), ^bSánchez-Hernández et al. (2020)

Como resultado del análisis, se escogió el marco de trabajo *Scrum* por su enfoque colaborativo en proyectos, lo que ayudó a controlar el avance mediante *sprints*. De igual manera se utilizó *Jira* para la gestión de las historias de usuario, asignar tareas y mantener la organización durante el desarrollo del proyecto.

4.2.1.5. Algoritmos de Machine Learning

Para la elección de algoritmo de *machine learning* del sistema, se tomó a consideración dos algoritmos de aprendizaje: redes neuronales y árboles de decisiones, que permitieron clasificar los datos según lo deseado, se visualiza en la tabla 11.

Tabla 11. Características de los algoritmos de Machine Learning

Redes Neuronales	Árboles de Decisión
Basadas en el funcionamiento del cerebro humano	Basados en la lógica de decisiones jerárquicas
Múltiples capas de neuronas conectadas entre sí	Nodos que representan decisiones y ramas con resultados
Requiere gran cantidad de datos y procesamiento	Más sencillo de interpretar y entrenar
Supervisado y no supervisado	Supervisado
Precisión alta en problemas no lineales y complejos	Precisión alta en problemas con relaciones más simples
Más lenta, requiere <i>hardware</i> especializado (<i>GPU</i>), pero es fácil conseguirlo	Más rápida y eficiente en <i>CPU</i>
Tolerancia a fallos, porque no dependen del ruido de los patrones de estudio	Riesgo moderado, controlable mediante poda (<i>pruning</i>)
Reconocimiento de patrones, imágenes, voz, predicciones avanzadas	Clasificación de datos, toma de decisiones y análisis descriptivo

Nota: ^a Serna (2017), ^b Barrientos et al. (2009)

Después del análisis, se optó por utilizar árboles de decisiones como algoritmo de aprendizaje automatizado, los cuales ayudan al procesamiento eficiente de los datos de manera visual, lo que optimiza la capacidad de aprendizaje mediante mecanismos supervisados en tareas complejas y de predicción.

4.2.1.6. Arquitectura de software

Es importante definir la estructura del proyecto para verificar la relación con sus componentes y como se fusiona el flujo de trabajo de los datos y la lógica comercial de la aplicación. Para el presente trabajo de titulación, se analizaron dos arquitecturas comunes en

el desarrollo de sistemas: Modelo-Vista-Controlador (MVC) y Modelo-Vista-Modelo de Vista (MVVM). En la tabla 12 se especifica cada una de sus características.

Tabla 12. Características de arquitectura de software

Características	Modelo–Vista–Controlador (MVC)	Modelo–Vista–Modelo de Vista (MVVM)
Estructura general	Divide el sistema en Modelo, Vista y Controlador	Añade un “Modelo de Vista” entre la vista y el modelo
Separación de responsabilidades	Separación de responsabilidad porque facilita el mantenimiento	Mejora la vinculación entre datos y vistas
Comunicación de datos	La vista se actualiza a través del controlador	Enlace bidireccional entre vista y modelo de vista
Complejidad	Complejidad Moderada	Complejidad Moderada
Mantenibilidad	Fácil de mantener en sistemas medianos para presentar datos	Ideal para interfaces interactivas

Nota: ^aFernández y Díaz (2012), ^bChikkanayakanahalli (2024).

Como resultado de la investigación entre las arquitecturas mencionadas, se optó por utilizar el Modelo Vista Controlador (MVC) como la base del producto, porque facilitó organizar las funciones correspondientes.

4.3. Resultado del tercer objetivo específico

4.3.1. Nomenclatura y logotipo de la aplicación *web* con *machine learning*

Durante el proceso para el desarrollo de la aplicación *web*, se decidió usar la nomenclatura “*Medical Pets*”, que se originó por la unión de las palabras “*Medical*” que proviene del inglés con relación a la medicina y “*Pets*” también proveniente del inglés que significa mascotas, el logotipo se muestra en la figura 9.

Figura 9. Logotipo de la aplicación



4.3.2. Marco de Trabajo Scrum

En el desarrollo de la aplicación con *machine learning* para el Hospital Veterinario Santa Martha, se utilizó el marco de trabajo *Scrum*, esta metodología fue implementada por su facilidad en el control de desarrollo ya que permite gestionar el avance de los *sprints* y tareas que contienen las historias. Según Schwaber y Sutherland (2020), *Scrum* es un marco ágil que permite gestionar, flexibilizar y retroalimentar los avances de la programación, se usa para productos complejos ya que reduce riesgos debido a su buena organización (p.3).

4.3.3. Sprint 1

4.3.3.1. Planificación del sprint 1 - (Sprint Planning)

Para la fase inicial del proyecto se desarrolló la planificación de tareas, lo cual tiene como objetivo el progreso eficiente del desarrollo de la aplicación *web*. En relación con el marco de trabajo *Scrum* se dividieron las historias en *sprints*, relacionando la importancia con el tiempo dedicado a cada una, sumando la prioridad y el requerimiento del cliente.

4.3.3.1.1. Roles

En el marco ágil *Scrum*, se define por roles que se detallan en la tabla 13, siendo estos: *Product Owner*, *Developers* y *Scrum Master*. Según Tymkiw et al. (2020), el *Product Owner* es quien marca los objetivos y toma la decisión final del producto a recibir. Por otro lado, el *Scrum Master* es el encargado de verificar que los requerimientos del cliente sean respetados haciendo que el equipo se acople al marco de trabajo establecido. Finalmente están los *developers*, que son los desarrolladores y quienes se encargan de materializar el proyecto, su tarea termina una vez entregado el producto final (p.83).

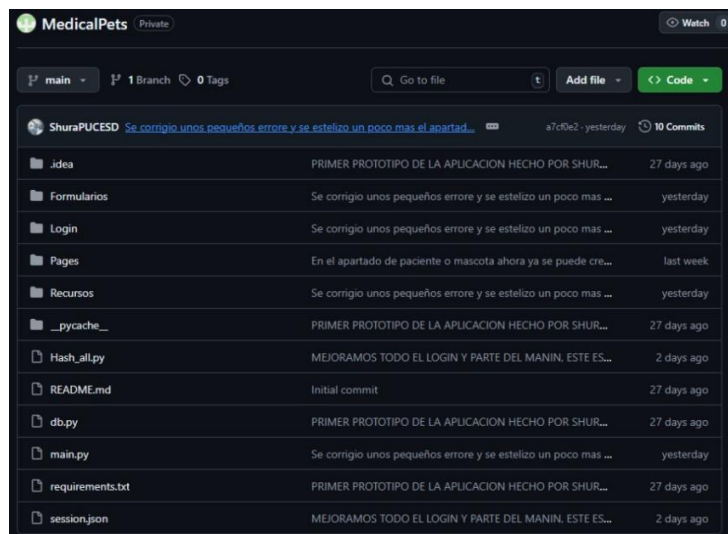
Tabla 13. Distribución de roles

Personas	Roles	Áreas
Dr. Patricio Intriago	<i>Product Owner</i> (<i>Propietario del producto</i>)	Gerente Propietario de Hospital Veterinario Santa Martha
Mg. Willian Ocampo	<i>Scrum Master</i>	Docente de la PUCESD
Jordan Herrera Jenniffer Paredes	<i>Developers</i>	Equipo de desarrollo, diseño y testeo

4.3.3.1.2. Control de versiones

Para controlar las versiones de la aplicación, se implementó la herramienta *GitHub*, esto permitió trabajar de manera conjunta en el desarrollo, de igual manera facilitó la gestión del código fuente y fue base en la seguridad para el control de errores en las nuevas modificaciones que se realizaban (figura 10).

Figura 10. Repositorio del proyecto en GitHub



4.3.3.1.3. Product Backlog

El *product backlog* es importante en el proceso de desarrollo, porque representa el conjunto de requerimientos para el resultado esperado del producto final. Para lograr establecer el *product backlog* se realizó una reunión con el equipo de desarrollo y el cliente estableciendo las necesidades de la aplicación (tabla 14).

Tabla 14. *Product Backlog*

N°	Historias	Estimación	Prioridad en negocio	Riesgo de desarrollo
1	Inicio de Sesión (<i>Login</i>)	1	100	Bajo
2	Cierre de sesión	1	100	Bajo
3	Registro de usuarios	3	99	Alto
4	Edición de usuarios	2	99	Alto
5	Eliminación de usuarios	2	99	Alto
6	Registro de pacientes (mascotas)	3	98	Alto
7	Edición de pacientes (mascotas)	2	98	Alto
8	Eliminación de pacientes (mascotas)	2	96	Alto
9	Creación de historial médico	4	95	Muy Alto
10	Edición de historial médico	2	95	Alto

11	Eliminación de historial médico	2	95	Alto
12	Visualización de historial médico	1	90	Bajo
13	Agendamiento de citas	4	90	Muy Alto
14	Eliminación de citas	2	89	Alto
15	Visualización de citas	1	89	Bajo
16	Registro de tratamientos	2	88	Alto
17	Edición de tratamientos	2	88	Alto
18	Finalización de tratamientos	2	88	Alto
19	Visualización de tratamientos	1	88	Bajo
20	Creación de notas	2	88	Alto
21	Edición de notas	2	87	Alto
22	Eliminación de notas	2	87	Alto
23	Análisis de historial médico	8	87	Muy Alto
24	Predicción de enfermedades	8	87	Muy Alto

Tabla 15. *Product Backlog* Técnico

N.	Historias	Estimación	Prioridad negocio	Riesgo de desarrollo
1	HT1-Seguridad e integridad de los datos de la aplicación	8	100	ALTO

4.3.3.1.4. Estimación

Para la valoración del proyecto, se utilizó la métrica de evaluación “días ideales” en la cual se indica el tiempo en realizar una tarea si no existieran interrupciones. Según *Atlassian* que es una empresa especializada en herramientas para la gestión de proyectos (2024), los días ideales son un concepto que ayuda al equipo de desarrollo a marcar una línea base en la productividad del equipo (tabla 16).

Tabla 16. Calendario de desarrollo

Calendario de trabajo			
Horas	Días	Semanas	Meses
1	5	12	3

4.3.3.1.5. Escenarios de prueba

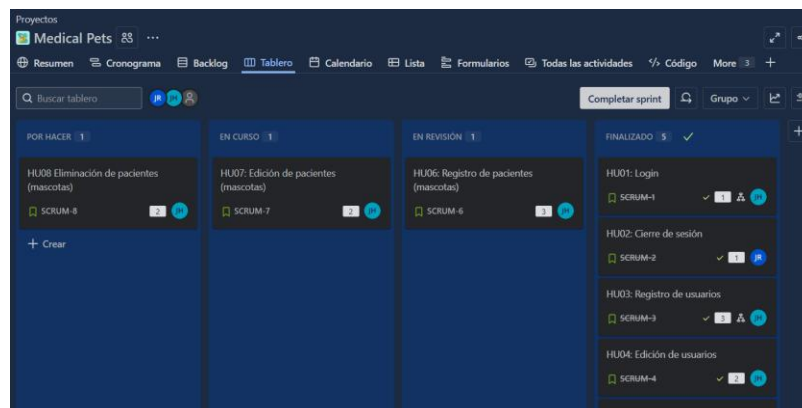
Una vez desarrollados las historias de usuarios, es importante considerar que los requisitos solicitados por el *product owner* se cumplan, es por ello que se presentan los escenarios de prueba, los resultados se encuentran en el anexo 6.

4.3.3.1.6. Velocidad de desarrollo

De acuerdo con los requerimientos establecidos por el propietario del producto y con relación a las estimaciones especificadas en las historias. La historia 1 y 2 cuentan con una estimación de 1 punto, la historia número 3 se valora en 3 puntos, de la misma manera la historia 4 y 5 se estiman en 2 puntos, la historia 6 en 3 puntos y la historia 7 y 8 en 2 puntos, lo que da como resultado un total de 16 puntos de historia. Esto se traduce a un equivalente de 16 días ideales de trabajo para el *sprint* 1.

4.3.3.1.7. Gestión de tareas y avances

La administración que se decidió para el avance de las tareas de cada historia se manejó en la plataforma *Jira*, normalmente utilizada para gestionar proyectos en especial de desarrollo de *software* como se muestra en la figura 11, donde se presenta la estructura utilizada en el marco de trabajo. Esta herramienta permite dividir la ejecución de los procesos. Figura 11. Tablero de Jira – Sprint 1 (Atlassian, 2024)



4.3.3.1.8. Sprint Backlog

Para el primer *sprint* se escogieron 8 primeras historias, la cuales sumaron un total de 16 puntos de estimación, como se visualiza en la tabla 17. Para Tymkiw et al. (2020), la importancia de un *sprint backlog* para el cronograma de tareas que deben realizar los desarrolladores es esencial para definir responsables en cada etapa del desarrollo (p.83) (Tabla 17).

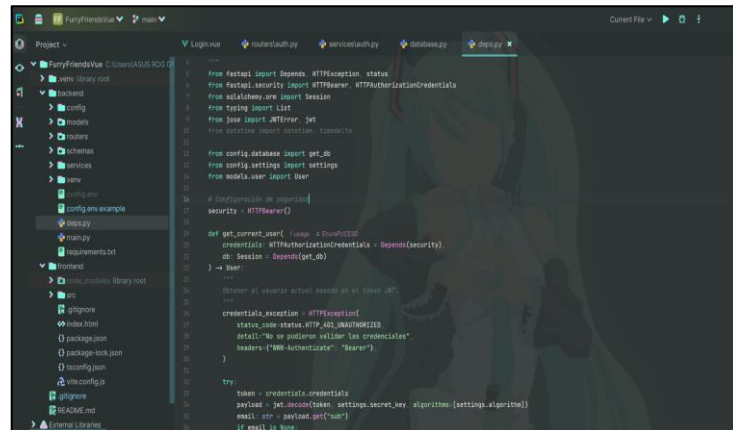
Tabla 17. *Sprint Backlog 1*

Sprint Backlog								
Objetivo: Implementar el sistema de autenticación y las funcionalidades básicas de gestión de usuarios y pacientes, sentando las bases para el funcionamiento del sistema								
Sprint	Historia	Est.	Categoría	Tarea	Est.	Responsable	Estado	
SPRINT 1	HU1 - Inicio de sesión	1	Desarrollo	Implementar endpoints de <i>autenticación JWT</i> (<i>backend/routers/auth.py</i>), (<i>backend/services/auth.py</i>), (<i>backend/models/user.py</i>)	0.25	Jordan Herrera	Completado	
			Desarrollo	Configurar el consumo del <i>Login</i> y persistencia de sesión (<i>frontend/src/config/api.js</i>), (<i>frontend/src/stores/auth.js</i>)	0.25	Jordan Herrera	Completado	
			Diseño	Diseñar e implementar la interfaz de acceso (<i>frontend/src/views/Login.vue</i>)	0.25	Jennifer Paredes	Completado	
			Prueba	Pruebas de inicio de sesión	0.25	Jennifer Paredes	Completado	
SPRINT 1	HU2 - Cierre de sesión	1	Desarrollo	Implementar <i>endpoint</i> de revocación/limpieza de sesión (<i>backend/routers/auth.py</i>), (<i>backend/services/auth.py</i>)	0.25	Jordan Herrera	Completado	
			Desarrollo	Gestionar cierre en el <i>store</i> y rutas protegidas (<i>frontend/src/stores/auth.js</i>), (<i>frontend/src/router/index.js</i>)	0.25	Jordan Herrera	Completado	
			Diseño	Ajustar UI de botones y confirmaciones en vistas con <i>logout</i> (<i>frontend/src/views/Dashboard.vue</i>), (<i>frontend/src/views/Pets.vue</i>)	0.25	Jennifer Paredes	Completado	
			Prueba	Pruebas de cierre de sesión	0.25	Jennifer Paredes	Completado	
SPRINT 1	HU3 - Registro de usuarios	3	Desarrollo	<i>Endpoints</i> y validaciones de creación de usuarios (<i>backend/routers/users.py</i>), (<i>backend/models/user.py</i>)	1	Jordan Herrera	Completado	
			Desarrollo	Formulario y manejo de estados en la vista de usuarios (<i>frontend/src/views/Users.vue</i>), (<i>frontend/src/stores/auth.js</i>)	1	Jordan Herrera	Completado	
			Diseño	UI de modal/alertas para altas de usuario según lineamientos (<i>Users.vue</i>)	0.5	Jennifer Paredes	Completado	
			Prueba	Pruebas de creación de usuario y <i>login</i> del nuevo usuario	0.5	Jennifer Paredes	Completado	
SPRINT 1		2	Desarrollo	Actualizaciones en repositorio y servicio (<i>backend/routers/users.py</i>), (<i>backend/services/users.py</i>)	1	Jordan Herrera	Completado	

	HU4 - Edición de usuarios		Sincronizar formularios de edición y refresco de actualización (<i>frontend/src/views/Users.vue</i>)	0.5	Jennifer Paredes	
			Prueba Verificación de actualización correcta	0.5	Jennifer Paredes	Completado
			Desarrollo <i>Endpoint</i> para lógica de usuarios (<i>backend/routers/users.py</i>)	0.25	Jordan Herrera	Completado
SPRINT 1	HU5 - Eliminación de usuarios	2	Confirmaciones y actualización del listado tras eliminar (<i>frontend/src/views/Users.vue</i>)	0.25	Jordan Herrera	Completado
			Diseño Botón de eliminación en interfaz (<i>crud_usuarios.py</i>)	0.25	Jennifer Paredes	Completado
			Prueba Validar eliminación, mensajes y refresco de <i>store</i>	0.25	Jennifer Paredes	Completado
			Desarrollo Formulario de registro (<i>frm_mascotas.py</i>)	1	Jennifer Paredes	Completado
SPRINT 1	HU6 - Registro de pacientes (mascotas)	3	Endpoints y relaciones con dueños (<i>backend/routers/pets.py</i>), (<i>backend/models/pet.py</i>)	1	Jordan Herrera	Completado
			Desarrollo Formulario y búsqueda de dueños en la vista (<i>frontend/src/views/Pets.vue</i>)	0.5	Jordan Herrera	Completado
			Diseño Componentes visuales para tarjetas y modal de creación (<i>Pets.vue</i>)	0.5	Jennifer Paredes	Completado
			Prueba Crear pacientes, validar campos y mensajes estandarizados	1	Jennifer Paredes	Completado
SPRINT 1	HU7 - Edición de pacientes (mascotas)	2	Desarrollo Servicio/ <i>endpoint</i> de actualización (<i>backend/routers/pets.py</i>)	0.5	Jordan Herrera	Completado
			Diseño Ajustes visuales para estados de edición en tarjetas	0.5	Jordan Herrera	Completado
			Pruebas Escenarios de edición, validaciones de edad/peso y <i>feedback</i>	1	Jennifer Paredes	Completado
SPRINT 1	HU8 - Eliminación de pacientes (mascotas)	2	Desarrollo <i>Endpoint</i> y reglas de eliminación (<i>backend/routers/pets.py</i>), (<i>backend/services/pets.py</i>)	1	Jordan Herrera	Completado
			Desarrollo Confirmaciones y actualización del <i>grid</i> tras borrar (<i>frontend/src/views/Pets.vue</i>)	0.5	Jordan Herrera	Completado
			Prueba Verificar bajas, mensajes y efectos secundarios sobre datos relacionados	0.5	Jennifer Paredes	Completado

La conexión a la base de datos y el ciclo de vida de las sesiones de los usuarios se manejan con *SQLAlchemy* y a través de *config.database.get_db* mientras que la verificación del *token* y la obtención del usuario actual se centralizan en *backend/deps.py* (figura 14 y 15).

Figura 14. Código de la autenticación del login y generación de token

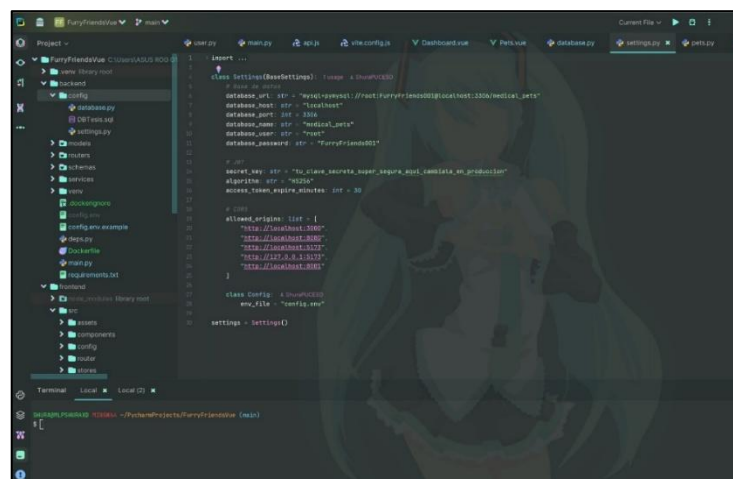


```

1 from fastapi import Depends, HTTPException, status
2 from fastapi.security import HTTPBearer, HTTPAuthorizationCredentials
3 from sqlalchemy.orm import Session
4 from typing import List
5 from jose import JWTError, jwt
6 from datetime import datetime, timedelta
7 from config.database import get_db
8 from config.settings import settings
9 from models.user import User
10
11 # Configuración de seguridad
12 security = HTTPBearer()
13
14 def get_current_user(token: str = Depends(security)):
15     credentials_exception = HTTPAuthorizationCredentials(
16         scheme="bearer", token=""
17     )
18     user = None
19     try:
20         token_data = jwt.decode(
21             token, settings.secret_key, algorithms=[settings.algorithm]
22         )
23         user = User(**token_data)
24     except JWTError:
25         raise credentials_exception
26     return user
27
28 def login(credentials: HTTPAuthorizationCredentials = Depends(security)):
29     token_data = jwt.decode(
30         credentials.token, settings.secret_key, algorithms=[settings.algorithm]
31     )
32     email = token_data.get("email")
33     if email is None:

```

Figura 15. Código de conexión a la base de datos



```

1 from pydantic import BaseSettings
2
3 class Settings(BaseSettings):
4     """
5     Configuración de la aplicación
6     """
7     database_url: str = "mysql+pymysql://root:furryfriends001@localhost:3306/medical_get"
8     database_host: str = "localhost"
9     database_port: int = 3306
10    database_name: str = "medical_get"
11    database_user: str = "root"
12    database_password: str = "furryfriends001"
13
14    # JWT
15    secret_key: str = "tu_clave_secreta_super_segura_misi_camabata_en_produccion"
16    algorithm: str = "HS256"
17    access_token_expire_minutes: int = 30
18
19    # CORS
20    allowed_origins: List[str] = [
21        "http://localhost:3000",
22        "http://localhost:3001",
23        "http://127.0.0.1:3000",
24        "http://localhost:3001"
25    ]
26
27    class Config:
28        env_file = ".env"
29
30    settings = Settings()

```

4.3.3.2. Historia de usuario 2: Cierre de sesión

El cierre de sesión se controla desde el *frontend* por el uso de autenticación basada en *JWT*. La actividad se ejecuta mediante el *store src/stores/auth.js*, *logout()* para poder eliminar el *token JWT* y limpiar el *localStorage*. Además, después de completar el proceso, *Vue Router* redirige al usuario a la vista de inicio de sesión que está en la ruta */login* en *src/router/index.js* (figura 16 y 17).

Figura 18. Código de backend/services/auth.py

```

from django.contrib.auth import authenticate, login, logout
from django.contrib.auth.models import User
from django.http import JsonResponse
from django.shortcuts import render

class CustomAuthBackend:
    def authenticate(self, request, username=None, password=None):
        user = authenticate(request, username=username, password=password)
        if user is not None:
            login(request, user)
            return user
        return None

    def get_user(self, username):
        return User.objects.get(username=username)

    def create_user(self, username, password, email=None, first_name=None, last_name=None):
        user = User.objects.create_user(username=username, password=password, email=email, first_name=first_name, last_name=last_name)
        return user

    def create_superuser(self, username, password, email=None, first_name=None, last_name=None):
        user = User.objects.create_superuser(username=username, password=password, email=email, first_name=first_name, last_name=last_name)
        return user

```

4.3.3.2.3. Historia de usuario 3: Registro de usuarios

Para el registro de usuarios se presenta un formulario, *frm_usuarios.py* para obtener los datos del nuevo usuario. La gestión para el envío de la información al *backend* se controla mediante *src/stores/users.js*, que crea al usuario usando *Axios* establecido en *src/config/api.js* (figura 19 y 20).

Figura 19. Interfaz de formulario para registro de usuarios

Figura 20. Código de frm_usuarios.py para registro de usuarios

```

from django import forms
from django.contrib.auth.models import User

class FormularioRegistro(forms.ModelForm):
    class Meta:
        model = User
        fields = ('username', 'email', 'password1', 'password2', 'cedula', 'telefono', 'direccion', 'rol')

    def clean(self):
        cleaned_data = super().clean()
        username = cleaned_data.get('username')
        email = cleaned_data.get('email')
        password1 = cleaned_data.get('password1')
        password2 = cleaned_data.get('password2')
        cedula = cleaned_data.get('cedula')
        telefono = cleaned_data.get('telefono')
        direccion = cleaned_data.get('direccion')
        rol = cleaned_data.get('rol')

        if password1 != password2:
            raise forms.ValidationError("Las contraseñas no coinciden.")

        if User.objects.filter(username=username).exists():
            raise forms.ValidationError("El usuario ya existe.")

        if User.objects.filter(email=email).exists():
            raise forms.ValidationError("El correo electrónico ya existe.")

        if rol != 'Usuario':
            raise forms.ValidationError("El rol no es válido.")

        return cleaned_data

    def save(self, commit=True):
        user = User.objects.create_user(
            username=self.cleaned_data['username'],
            email=self.cleaned_data['email'],
            password=self.cleaned_data['password1'],
            first_name=self.cleaned_data['cedula'],
            last_name=self.cleaned_data['telefono'],
            address=self.cleaned_data['direccion'],
            role=self.cleaned_data['rol']
        )
        return user

```


cache o se fuerza *refreshList*. Además, se integran *guards* de ruteo para roles/permisos y se limpian estados al desmontar (figura 23 y 24).

Figura 23. Interfaz para la edición de usuarios

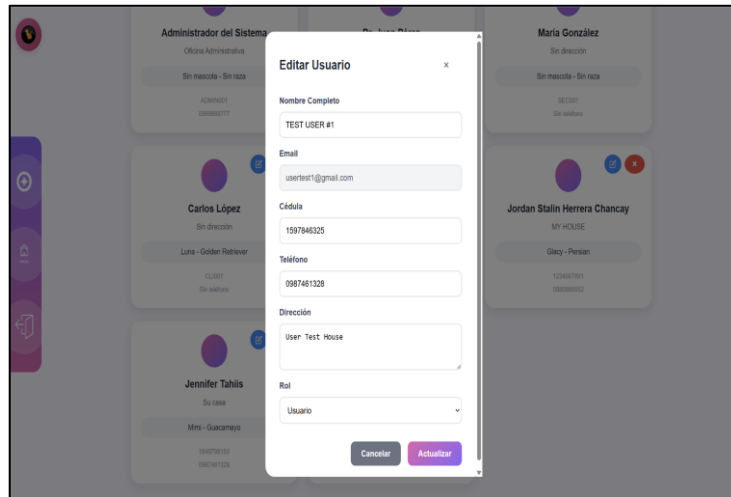
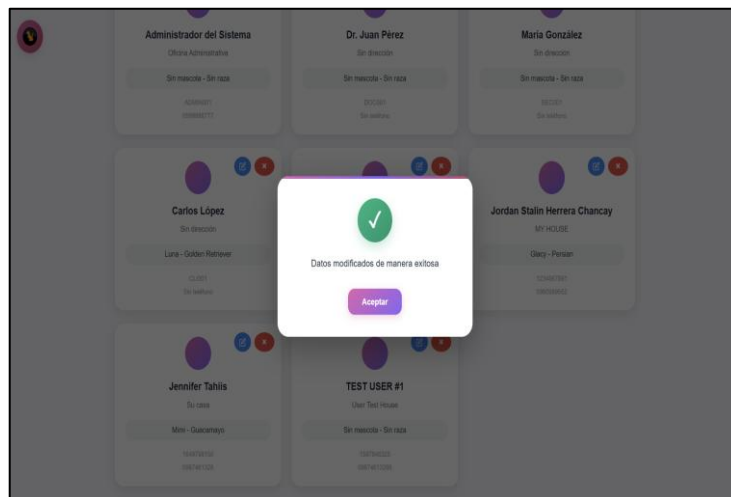
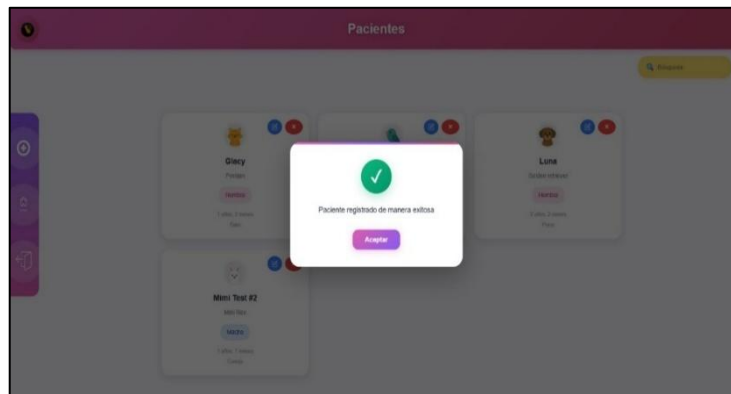


Figura 24. Respuesta a actualizaciones de usuarios

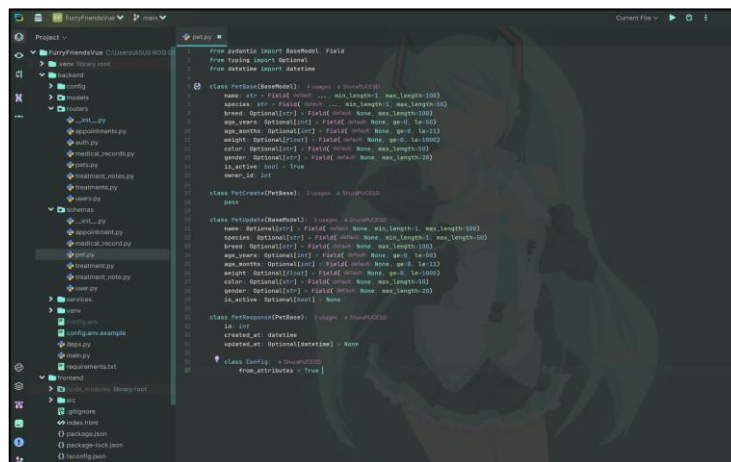


En el *backend*, *FastAPI* expone el *endpoint* de actualización (*PATCH /users/{id}* en *routers/users.py*), valida el *payload* con *Pydantic*, garantiza unicidad de *email* si cambió, también re-cifra la contraseña con *bcrypt* solo si fue modificada, persiste los cambios y devuelve un *DTO* sin campos sensibles en errores, retorna 404 si no existe el usuario o el código adecuado ante conflictos (como *email* duplicado) para que el *frontend* informe correctamente (figura 25).

Figura 29. Interfaz de respuesta a creación de mascotas



El *backend* procesa la solicitud mediante el *endpoint* *POST* */api/pets*, validando los datos en el esquema *PetCreate* en “*routers/pets.py*”. También se valida el *payload* con *Pydantic*, se normalizan textos, se asocia el registro con el propietario autenticado y se gestiona la foto, se persiste mediante la capa de servicios/repositorio y se devuelve un *DTO* (*PetRead*) sin datos sensibles (figura 30).

Figura 30. Código de *pet.py* para registro de usuarios

La información se registra en la base de datos, en la tabla *pets* descrita en *backend/models/pet.py*, utilizando una sesión de base de datos establecida con *backend/config/database.py* (figura 31).

Figura 31. *medical_pets.pets*

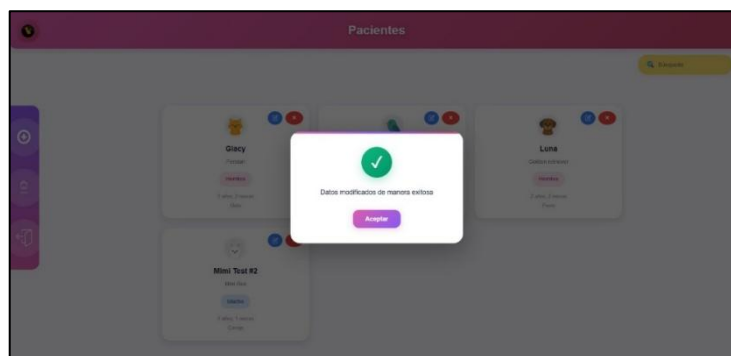
	name	species	breed	weight	gender	owner_id	created_at	updated_at	eye_years
1	Luna	Ferros	Golden Retriever	25	Hembra	4	2025-08-27 17:59:36	2025-08-29 18:36:36	3
2	Max	Gato	Siamés	4.5	Macho	4	2025-08-27 17:59:36	2025-08-29 18:36:36	1
3	Ricky	Ferros	Pastor Alemán	30	Macho	4	2025-08-27 17:59:36	2025-08-29 18:36:36	0
4	Glacy	Gato	Perlán	15	Hembra	4	2025-08-29 02:33:30	2025-10-15 02:33:39	2
5	Mia	Ave	Gatcajero	22	Macho	13	2025-10-15 02:37:47	2025-10-15 02:37:47	3
6	Dijetes	Gato	Naranjero	50	Macho	5	2025-10-29 02:33:30	2025-10-29 02:33:30	1
7	Fast pet	Kanaster	test raza	51	Macho	4	2025-10-29 02:45:19	2025-10-29 02:45:14	1
8	Lana	Coneja	Coneja Arliso	50	Hembra	13	2025-11-03 00:05:59	2025-11-03 00:05:59	0
9	June	Kanaster	Jane	50	Macho	13	2025-11-03 00:07:12	2025-11-03 00:07:12	1
10	Fluttershy	Itro	Pony	40	He	6	2025-11-03 00:09:25	2025-11-03 00:09:25	7

4.3.3.2.7. Historia de usuario 7: Edición de mascotas

La edición de los pacientes se maneja mediante un formulario de actualización de datos ubicados en `src/views/Pets/EditPetView.vue` en la ruta `/pets`. De esa manera el usuario modifica información acerca de la mascota, se validan los cambios y al confirmar se envía un `PATCH /pets/id`, al final se muestra el mensaje de confirmación (figura 32 y 33).

Figura 32. Interfaz de formulario para modificar datos de pacientes

Figura 33. Interfaz de respuesta a modificación de datos



En *FastAPI* se expone `PATCH /pets/{id}` en `routers/pets.py`, se valida el *payload* con *Pydantic* se normalizan textos, se persisten los cambios vía la capa de servicio/repositorio y se retorna un *DTO* (*PetRead*) consistente con el resto de rutas (figura 34).

Figura 34. Código de `/pets.py` para actualización de datos

```

1  @router.patch("/{pet_id}", response_model=PetRead)
2  async def update_pet(request: Request, pet_id: str):
3      pet = await PetRepository.get_by_id(pet_id)
4      if not pet:
5          raise HTTPException(status_code=404, detail="Pet not found")
6
7      # Validar y normalizar el payload
8      payload = request.json()
9      payload["name"] = str(payload.get("name", ""))
10     payload["species"] = str(payload.get("species", ""))
11     payload["breed"] = str(payload.get("breed", ""))
12     payload["age"] = int(payload.get("age", 0))
13     payload["weight"] = float(payload.get("weight", 0))
14     payload["color"] = str(payload.get("color", ""))
15     payload["gender"] = str(payload.get("gender", ""))
16     payload["owner"] = str(payload.get("owner", ""))
17
18     # Actualizar los datos en el repositorio
19     await PetRepository.update(pet_id, payload)
20
21     # Retornar el DTO PetRead
22     return PetRead(pet)

```


4.3.3.3. Revisión del sprint 1

Al concluir el desarrollo del *sprint 1*, se llevó a cabo una reunión con el propietario, con el objetivo de presentar el avance del producto y validar el tiempo establecido para el *sprint*. Además, se verificó el cumplimiento de las tareas asignadas y la funcionalidad de las historias completadas (anexo 6).

4.3.3.4. Retrospectiva del sprint 1

Se consideraron tres preguntas para la retrospectiva del *sprint 1*, las cuales se visualizan en la tabla 18, estas preguntas ayudaron a tener una visión clara sobre cómo se desarrolló el primer *sprint*.

Tabla 18. Sprint 1 retrospectiva

Aspectos exitosos del sprint	Mejoras para aplicar	Errores en el sprint
La asignación de tareas durante el proceso, fue una buena práctica para el primer <i>sprint</i> . Permitió cumplir con el cronograma establecido. Además, la integración de <i>mysql</i> como base de datos, ayudó a una interacción exitosa con el <i>backend</i>	Se identificó la necesidad de mejorar el mecanismo de expiración de tiempo en pantalla sin interacción (<i>tokens</i>)	Se detectó un problema en la prueba de funcionalidad (HU3) "Registro de usuarios", no se definieron los permisos de usuario para acceder a este módulo

4.3.4. Sprint 2

4.3.4.1. Planificación del sprint 2 - (Sprint Planning)

Para la segunda fase del proyecto, se presentan 8 siguientes historias de usuario que fueron consideradas para este *sprint*, estas historias tuvieron como objetivo desarrollar las funcionalidades en cuanto a la gestión de historiales médicos, citas y tratamientos. Además, se realizó la corrección de errores del *sprint 1*.

4.3.4.1.1. Velocidad de Desarrollo

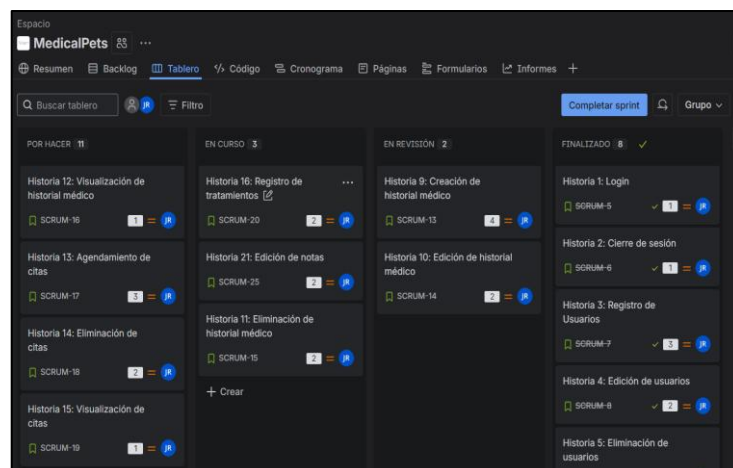
De acuerdo con los requerimientos establecidos por el propietario del producto y con relación a las estimaciones especificadas en las historias. La historia 9 cuenta con una estimación de 4 puntos, las historias número 10 y 11 se valora en 2 puntos, de la misma

manera la historia 12 se estima en 1 punto, la historia 13 en 3 puntos, la historia 14 cuenta con una estimación de 2, la historia 15 con 1 punto y 16 en 2 puntos, lo que da como resultado un total de 17 puntos de historia. Esto se traduce a un equivalente de 17 días ideales de trabajo para el *sprint* 2.

4.3.4.1.2. Gestión de tareas y avances

La administración de las tareas, al igual que el *sprint* 1 se manejó en la plataforma *Jira*, ya que ayudó a gestionar la primera parte del desarrollo de la aplicación. Esta herramienta tecnológica fue esencial para ordenar de manera adecuada cada historia. En la figura 37 se presenta la estructura utilizada en el marco de trabajo para el segundo *sprint*.

Figura 37. Tablero de Jira – Sprint 2 (Atlassian, 2024)



Se divide en 4 columnas, que demuestran el proceso por el que están pasando las historias durante el progreso de la *app*. La primera indica que la historia está pendiente por realizar, la segunda muestra que está siendo desarrollada, cuando pasa a la columna de revisión significa que la historia está hecha, pero necesita revisión para comprobar su funcionamiento y la última señala que la historia fue hecha y funciona con éxito.

4.3.4.1.3. Sprint Backlog

Para el segundo *sprint* se escogieron 8 siguientes historias, la cuales sumaron un total de 17 puntos de estimación, como se visualiza en la tabla 19.

Tabla 19. Sprint Backlog 2

Sprint Backlog								
Objetivo: Desarrollar las funcionalidades para la gestión de historial médico de las mascotas, agenda de citas y control de tratamientos, definiendo accesos y roles.								
Sprint	Historia	Est.	Categoría	Tarea	Est.	Responsable	Estado	
SPRINT 2	HU9 - Creación de historial medico	4	Desarrollo	Se aplican <i>endpoints</i> y validaciones para su creación (<i>backend/routers/medical_records.py</i>),	1	Jordan Herrera	Completado	
			Desarrollo	Creación de formulario y selección de paciente en la vista (<i>frontend/src/views/MedicalRecords.vue</i>)	1	Jordan Herrera	Completado	
			Diseño	Modal de creación y alertas según la paleta de colores (<i>MedicalRecords.vue</i>)	1	Jennifer Paredes	Completado	
			Prueba	Validar la creación, campos requeridos y mensajes estándar.	1	Jennifer Paredes	Completado	
SPRINT 2	HU10 - Edición de historial medico	2	Desarrollo	<i>Endpoint</i> y servicio de actualización (<i>backend/routers/medical_records.py</i>)	0.75	Jordan Herrera	Completado	
			Desarrollo	Modal de edición y sincronización con el store (<i>frontend/src/views/MedicalRecords.vue</i>)	0.75	Jordan Herrera		
			Prueba	Verificación de actualizaciones y mensajes de éxito/error	0.5	Jennifer Paredes	Completado	
SPRINT 2	HU11 - Eliminación de historial médico	2	Desarrollo	<i>Endpoint</i> de eliminación con validaciones (<i>backend/routers/medical_records.py</i>)	0.75	Jordan Herrera	Completado	
			Desarrollo	Confirmación y actualización del <i>grid</i> tras eliminar (<i>frontend/src/views/MedicalRecords.vue</i>)	0.75	Jordan Herrera	Completado	
			Prueba	Validar la eliminación y mensajes estándar	0.5	Jennifer Paredes	Completado	
SPRINT 2	HU12 - Visualización de historial médico	1	Desarrollo	<i>Endpoints</i> de consulta y filtros (<i>backend/routers/medical_records.py</i>)	0.25	Jordan Herrera	Completado	
			Desarrollo	<i>Grid</i> de tarjetas y búsqueda en vista (<i>frontend/src/views/MedicalRecords.vue</i>)	0.25	Jordan Herrera	Completado	
			Diseño	Tarjeta de historial médico con información estructurada	0.25	Jennifer Paredes	Completado	

SPRINT 2	HU13 - Agendamiento de citas	3	Prueba	Validar visualización, filtros y permisos por rol	0.25	Jennifer Paredes	Completado
			Desarrollo	Endpoints y validaciones de creacion (<i>backend/routers/appointments.py</i>)	1.5	Jordan Herrera	Completado
			Desarrollo	Formulario de agendamiento con selección de paciente y veterinario (<i>frontend/src/views/Appointments.vue</i>)	1	Jordan Herrera	Completado
			Prueba	Validar creación, conflictos de horarios y mensajes	0.5	Jennifer Paredes	Completado
SPRINT 2	HU14 - Eliminación de citas	2	Desarrollo	<i>Endpoint</i> de eliminación con sus respectivas validaciones (<i>backend/routers/appointments.py</i>)	0.75	Jordan Herrera	Completado
			Desarrollo	Confirmacion y actualización de la lista tras eliminar (<i>frontend/src/views/Appointments.vue</i>)	0.75	Jordan Herrera	Completado
			Prueba	Validar eliminación y mensajes de error estándar	0.5	Jennifer Paredes	Completado
SPRINT 2	HU15 - Visualización de citas	1	Desarrollo	<i>Endpoints</i> de consulta y filtros por estado y fecha (<i>backend/routers/appointments.py</i>)	0.25	Jordan Herrera	Completado
			Desarrollo	Vista diferenciada para los usuarios y administradores (<i>frontend/src/views/Appointments.vue</i>)	0,25	Jennifer Paredes	Completado
			Diseño	Tarjetas de citas con estados visuales (pendiente, confirmada, completada)	0.25	Jennifer Paredes	Completado
			Prueba	Validar visualización, filtros y permisos por rol	0.25	Jennifer Paredes	Completado
SPRINT 2	HU16 - Registro de tratamientos	2	Desarrollo	<i>Endpoint</i> y validación de creación (<i>backend/routers/treatments.py</i> , <i>backend/models/treatment.py</i>)	0.75	Jordan Herrera	Completado
			Desarrollo	Formulario de registro con selección de paciente (<i>frontend/src/views/Treatments.vue</i>)	0.75	Jordan Herrera	Completado
			Prueba	Validar creación, campos requeridos y mensajes estándar	0.5	Jennifer Paredes	Completado

4.3.4.2. Sprint 2 - Reuniones diarias - Daily Scrum

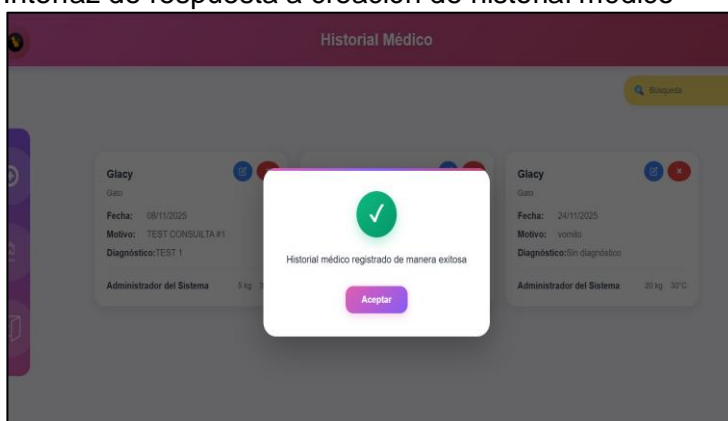
Para el segundo *sprint*, se realizaron reuniones de 15 minutos para actualizar el tablero en *Jira* y ofrecer retroalimentación sobre las tareas terminadas al propietario. Además, el monitoreo de las historias fue de acuerdo al cronograma definido.

4.3.4.2.1. Historia de usuario 9: Creación de historial médico

Para realizar la historia de usuario 9, se creó un formulario que inicia con la ficha médica de la mascota `/pets/: petId` en la ruta `/pets/: petId/records/create`. Desde allí los datos se controlan a través del *store* `src/stores/medicalRecords.js`, el cual invoca la acción de crear el historial médico `createRecord(payload)` para enviar la información al *backend* usando *Axios*, que está configurado en `src/config/api.js`, los resultados se muestran en la figura 38 y 39.

Figura 38. Interfaz de formulario para la creación de historial médico

Figura 39. Interfaz de respuesta a creación de historial médico



En el *backend* la solicitud enviada se procesa en el *endpoint* `POST /api/medical-records` declarados en `backend/routers/medical_records.py` ubicado en `backend/schemas/medical_record.py`. Además, se verifica la existencia del paciente, la

asociación al dueño y el correcto registro, esto se implementa en *backend/services/medical_records.py*. De igual manera la inserción de datos se realiza en la base de datos en la tabla *medical_records*, que se representa en la figura 40.

Figura 40. Código de *medical_records.py* para la creación de historiales

```

from pydantic import BaseModel, Field
from typing import Optional
from datetime import datetime

class MedicalRecord(BaseModel):
    """Modelo de un historial médico"""
    patient_id: int = Field(..., description="ID de la mascota")
    visit_date: datetime = Field(..., description="Fecha de la consulta")
    consultation_reason: Optional[str] = Field(None, description="Motivo de la consulta")
    symptoms: Optional[str] = Field(None, description="Síntomas observados")
    medical_history: Optional[str] = Field(None, description="Historial médico")
    diagnosis: Optional[str] = Field(None, description="Diagnóstico realizado")
    treatment: Optional[str] = Field(None, description="Tratamiento prescrito")
    prescriptions: Optional[str] = Field(None, description="Medicamentos prescritos")
    notes: Optional[str] = Field(None, description="Notas del veterinario")
    observations: Optional[str] = Field(None, description="Observaciones adicionales")
    temperature: Optional[float] = Field(None, ge=36, le=42, description="Temperatura corporal en °C")
    weight: Optional[float] = Field(None, ge=0, le=1000, description="Peso en kg")

class MedicalRecordCreate(MedicalRecord):
    """Modelo para crear un historial médico"""
    pass

class MedicalRecordUpdate(MedicalRecord):
    """Modelo para actualizar un historial médico"""
    pass

class MedicalRecordResponse(MedicalRecord):
    """Modelo para responder un historial médico"""
    id: int
    created_at: datetime
    updated_at: Optional[datetime] = None

    class Config:
        from_attributes = True

if __name__ == "__main__":
    """Ejemplo de uso del modelo"""
    record = MedicalRecordCreate(
        patient_id=1,
        visit_date=datetime.now(),
        consultation_reason="Consulta de rutina",
        symptoms="Sin síntomas aparentes",
        medical_history="Estado de salud excelente",
        diagnosis="Sin diagnóstico",
        treatment="Sin tratamiento",
        prescriptions="Sin prescripciones",
        notes="Sin notas",
        observations="Sin observaciones",
        temperature=37.5,
        weight=10.5
    )
    print(record)

```

4.3.4.2.2. Historia de usuario 10: Edición de historial médico

La edición del historial médico es una base importante para la aplicación, ya que permite mantener actualizada la información del paciente, esta se implementa con la ruta */pets/:petId/records/:recordId/edit* que carga el componente utilizando el store *src/stores/medicalRecords.js* para que muestre la información actual del paciente. Además para editarla se usan las acciones *getRecordById()* y *updateRecord()* (figura 41 y 42).

Figura 41. Interfaz de respuesta a actualización de datos en los historiales

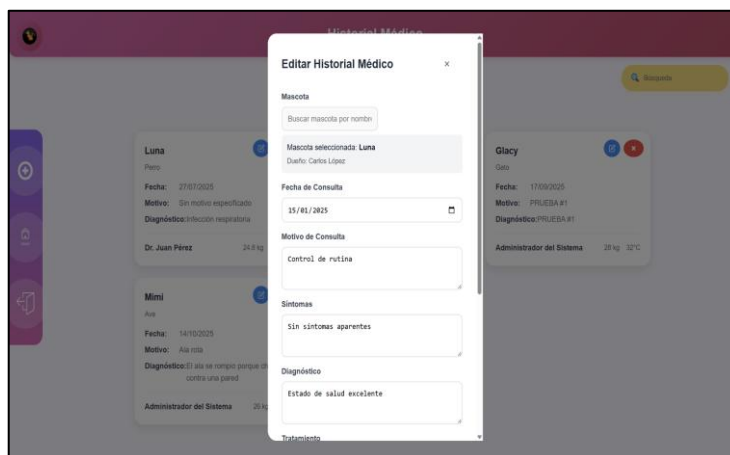
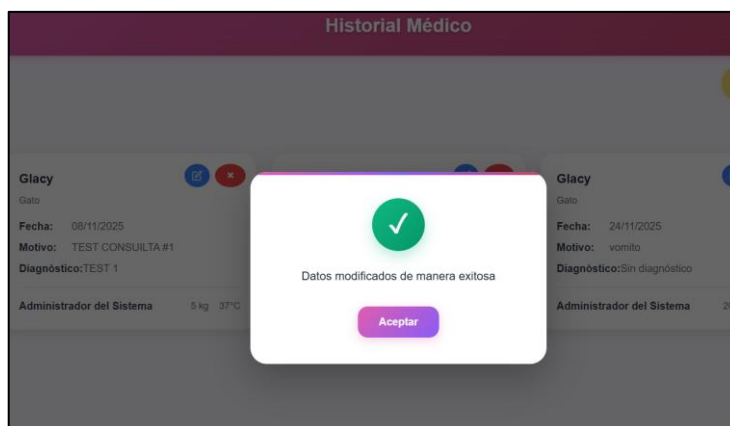
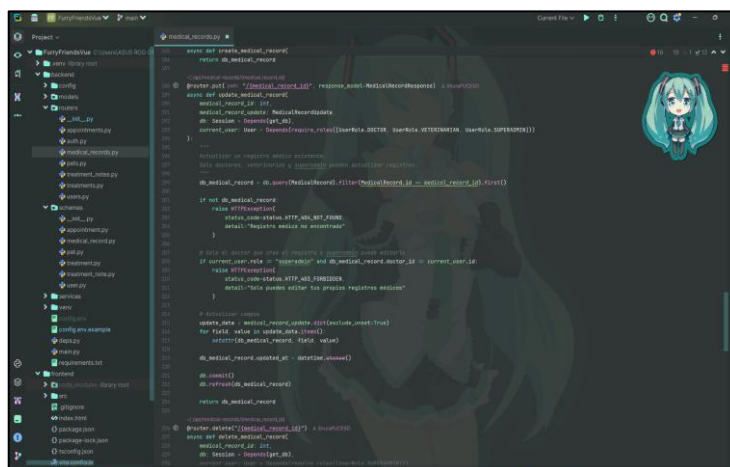


Figura 42. Interfaz de respuesta a la actualización de historiales



En el lado del *backend*, la actualización de los datos se gestiona con el *endpoint PUT* `/api/medical-records/{id}`. Allí los datos son validados con `MedicalRecordUpdate` ubicado en `backend/schemas/medical_record.py`, lo que permite la edición segura del historial médico y la actualización de la nueva información en la base de datos. Además, se aplican actualizaciones parciales desde la capa de servicios/repositorio y se retorna un *DTO* ("*HistoryRead*") como se muestra en la figura 43.

Figura 43. Código para la edición de historiales

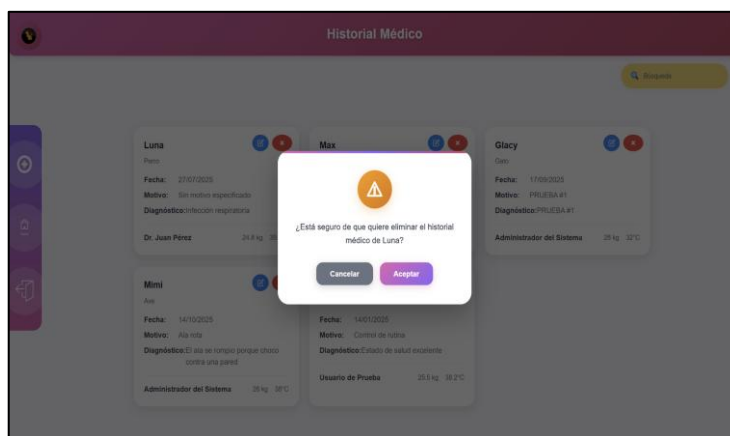


4.3.4.2.3. Historia de usuario 11: Eliminación de historial médico

Para la eliminación de un historial médico, el usuario debe seleccionar la opción eliminar en las opciones que ofrece el listado de historiales como se evidencia en la figura 44. Esta vista está ubicada en `src/views/MedicalRecords/RecordsListView.vue` usando el *store* `src/stores/medicalRecords.js`, ejecutando la acción `deleteRecord(id)` que envía la petición

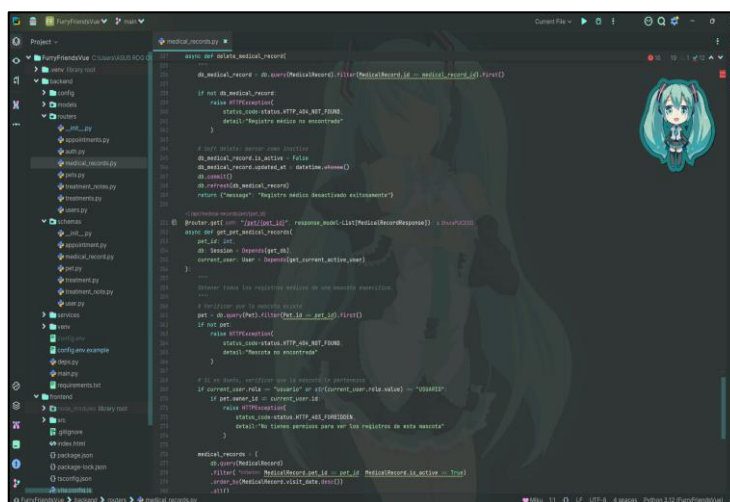
DELETE mediante *Axios* previamente configurados. Además, el interceptor añade el *JWT*, se bloquea el botón mientras se procesa y se aplica “*optimistic UI*” para que la tarjeta del historial desaparezca de inmediato y se revierta si algo falla. Al finalizar, se actualiza la lista de historiales mostrando un mensaje con el resultado correcto.

Figura 44. Interfaz de mensaje de eliminación de historiales



Para el *backend* en *FastAPI* se expone *DELETE /pets/{id}/records/{record_id}* en *routers/medical_records.py*, el código se visualiza en la figura 45. La historia de eliminación se implementa en *backend/services/medical_records.py* donde se verifica la existencia del historial y se ejecuta su eliminación mediante *SQLAlchemy*. La validación de permisos y autenticación se realiza utilizando las dependencias de seguridad previstas en *backend/deps.py*.

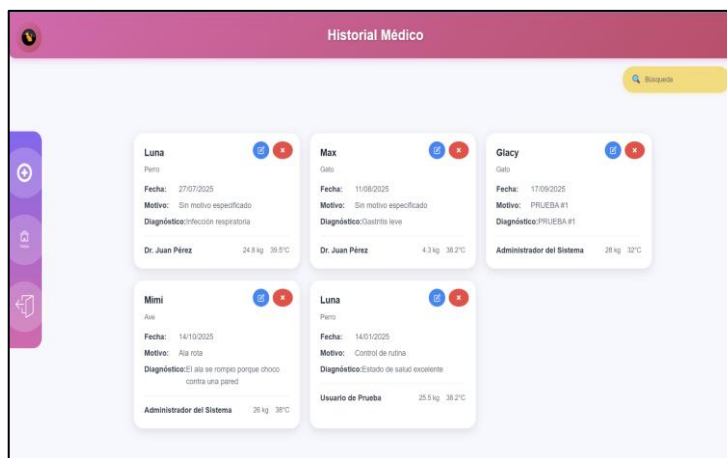
Figura 45. Código para la eliminación de historiales



4.3.4.2.4. Historia de usuario 12: Visualización de Historial médico

La visualización de los historiales médicos se implementa a través de un conjunto de vistas en el *frontend*, desde la ficha del dueño de la mascota registrada `/pets/:petId`, vista `src/views/Pets/PetDetailView.vue`. El usuario visualiza el historial médico de su mascota mediante la misma ruta de las historias anteriores `/pets/:petId/records`, que carga el componente `src/views/MedicalRecords/RecordsListView.vue`, en el que se usa el *store* `MedicalRecords`, lo que le permite al usuario (dueño de la mascota) revisar el historial médico de su mascota y sus actualizaciones, el resultado de evidencia en la figura 46.

Figura 46. Interfaz de la vista de historiales médicos



En *FastAPI* se exponen `GET /pets/{id}/records` (listado con filtros, orden y paginación) y `GET /pets/{id}/records/{record_id}` (detalle), las rutas validan parámetros con *Pydantic*, resuelven la relación con el paciente. También, consultan la capa de servicios/repositorio, formatean fechas/unidades y devuelven *DTOs* consistentes (`"HistoryListItem"` y `"HistoryRead"`) incluyendo metadatos de adjuntos para su previsualización segura.

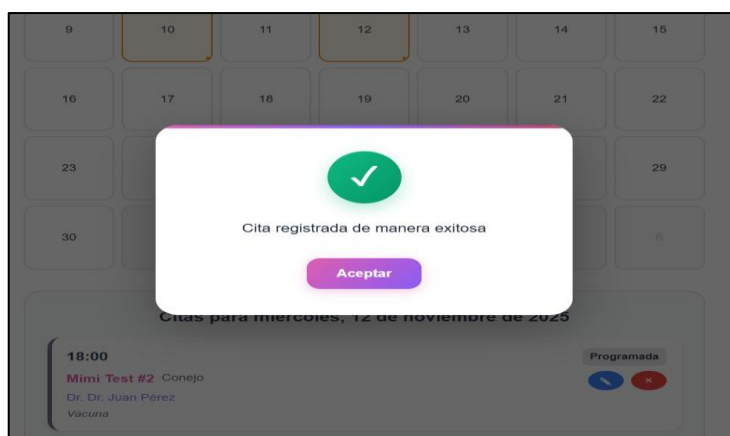
4.3.4.2.5. Historia de usuario 13: Agendamiento de citas

Para agendar una nueva cita se implementan las vistas específicas en el *frontend* que permiten registrar nueva información. Desde la lista de citas `/appointments` se ingresa a la ruta `/pets/:petId/appointments/create` que carga el componente para ejecutar la acción.

De igual manera, en la vista de Citas se integró un calendario interactivo como componente (*Vue wrapper* de “*FullCalendar*”) configurado en español, con franjas horarias, deshabilitado de fechas pasadas y reglas de disponibilidad como se observa en la figura 47. De igual forma, junto a él se muestra un formulario reactivo para seleccionar paciente, profesional, motivo y duración. Al hacer *clíc* en un bloque libre o confirmar el formulario, se dispara un “*POST /appointments*” vía *Axios* (el interceptor añade el *JWT*), validación en tiempo real, se bloquea los controles con “*isSaving*” en *Pinia*, se pinta la nueva cita en el calendario y se actualiza la lista sin recargar. Además, se usa un *date-time picker* para entradas manuales y sincronización para evitar inconsistencias (figura 48).

Figura 47. Interfaz para crear nueva cita

Figura 48. Respuesta a agendar nueva cita

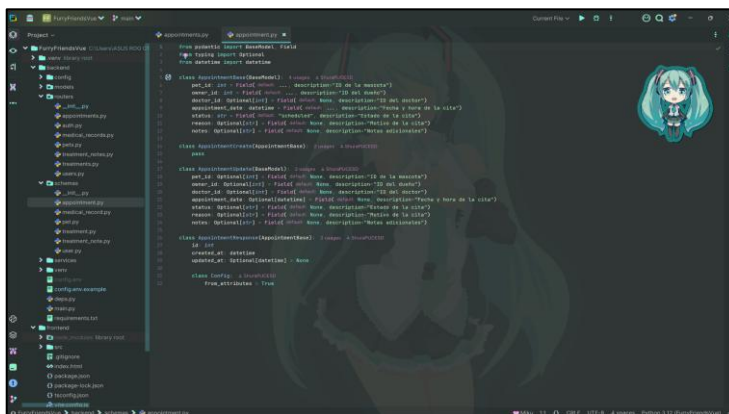


En *FastAPI* se expone *GET /appointments* (listado y filtros), *GET /appointments/availability* (disponibilidad por fecha/profesional) y *POST /appointments* en *routers/appointments.py*. Se validan los datos con *Pydantic* (*AppointmentCreate*), se normalizan zonas horarias, se verifica solapamiento contra la agenda del profesional, se

asocia la cita al paciente autenticado y se persiste vía la capa de servicios/repositorio.

También, la respuesta devuelve un *DTO (AppointmentRead)* listo para que el *frontend* pinte el evento en el calendario (figura 49).

Figura 49. Código de `appointment.py` para agendar de citas



```

class AppointmentRead(BaseModel):
    """AppointmentRead"""
    id: int = Field(..., description="ID de la cita")
    patient_id: int = Field(..., description="ID del paciente")
    doctor_id: int = Field(..., description="ID del doctor")
    status: str = Field(..., description="Estado de la cita")
    reason: str = Field(..., description="Motivo de la cita")
    notes: str = Field(..., description="Notas adicionales")

class AppointmentCreate(BaseModel):
    """AppointmentCreate"""
    patient_id: int = Field(..., description="ID de la cita")
    doctor_id: int = Field(..., description="ID del paciente")
    status: str = Field(..., description="Estado de la cita")
    reason: str = Field(..., description="Motivo de la cita")
    notes: str = Field(..., description="Notas adicionales")

class AppointmentService(BaseService):
    """AppointmentService"""
    def create(self, appointment: AppointmentCreate) -> AppointmentRead:
        """Create appointment"""
        appointment = Appointment(**appointment.dict())
        db_session.add(appointment)
        db_session.commit()
        return appointment

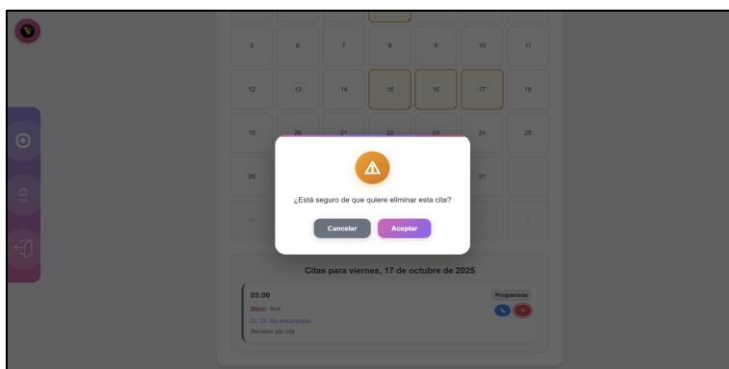
class AppointmentView(BaseView):
    """AppointmentView"""
    def create(self, appointment: AppointmentCreate) -> AppointmentRead:
        """Create appointment"""
        appointment = self.service.create(appointment)
        return appointment

```

4.3.4.2.6. Historia de usuario 14: Eliminación de citas

La eliminación de citas se implementa a partir de la vista del listado de citas `/appointments` que es gestionada por el componente `src/views/Appointments/` donde cada registro necesita una acción para eliminar una cita, como se muestra en la figura 50. Es decir, desde el calendario o la lista de citas, al seleccionar una cita y pulsar el ícono de eliminar se muestra una confirmación, si se acepta se envía un `DELETE /appointments/id` con `Axios`. Se procesa y aplica *optimistic UI*: retirando el evento del calendario (`FullCalendar/vue-cal`) y de la lista, en *rollback* en caso fallara, el *store* de `Pinia` sincroniza el estado y se notifica el resultado al usuario.

Figura 50. Interfaz para eliminar una cita



En el *backend* se expone `DELETE /appointments/{id}` en `routers/appointments.py`, con esto la ruta resuelve el identificador, luego delega a la capa de servicios para remover la

cita, actualiza la agenda del profesional y envía una respuesta estandarizada para que el *frontend* refleje el cambio en el calendario (figura 51).

Figura 51. Código de appointments.py

```

def update_appointment(appointment_id, appointment_status):
    """Actualizar el estado de una cita"""
    db_appointment = db.query(Appointment).filter(Appointment.id == appointment_id).first()
    if db_appointment is None:
        raise HTTPException(status_code=404, detail="Cita no encontrada")

    db_appointment.appointment_status = appointment_status

    db.commit()

    return db_appointment

def delete_appointment(appointment_id):
    """Eliminar una cita"""
    db_appointment = db.query(Appointment).filter(Appointment.id == appointment_id).first()
    if db_appointment is None:
        raise HTTPException(status_code=404, detail="Cita no encontrada")

    db.delete(db_appointment)

    db.commit()

    return db_appointment

def create_appointment(appointment_status):
    """Crear una cita"""
    db_appointment = db.query(Appointment).filter(Appointment.appointment_status == appointment_status).first()
    if db_appointment is None:
        raise HTTPException(status_code=404, detail="Cita no encontrada")

    db_appointment.appointment_status = appointment_status

    db.commit()

    return db_appointment

def get_appointments():
    """Obtener todas las citas"""
    db_appointments = db.query(Appointment)

    return db_appointments

def get_appointment(appointment_id):
    """Obtener una cita por ID"""
    db_appointment = db.query(Appointment).filter(Appointment.id == appointment_id).first()
    if db_appointment is None:
        raise HTTPException(status_code=404, detail="Cita no encontrada")

    return db_appointment

def get_appointments_by_professional(professional_id):
    """Obtener citas por profesional"""
    db_appointments = db.query(Appointment).filter(Appointment.professional_id == professional_id)

    return db_appointments

def get_appointments_by_patient(patient_id):
    """Obtener citas por paciente"""
    db_appointments = db.query(Appointment).filter(Appointment.patient_id == patient_id)

    return db_appointments

def get_appointments_by_date(date):
    """Obtener citas por fecha"""
    db_appointments = db.query(Appointment).filter(Appointment.date == date)

    return db_appointments

def get_appointments_by_status(status):
    """Obtener citas por estado"""
    db_appointments = db.query(Appointment).filter(Appointment.appointment_status == status)

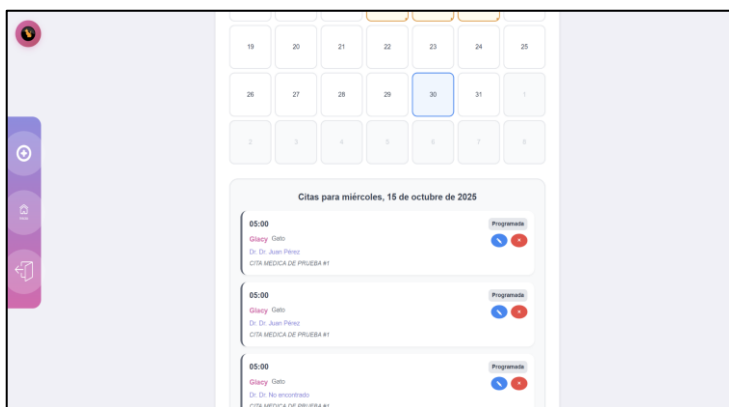
    return db_appointments

```

4.3.4.2.7. Historia de usuario 15: Visualización de citas

La visualización de citas se realiza con un calendario interactivo configurado en español y sincronizado con una lista lateral, el *store* de *Pinia* solicita eventos para el rango visible mediante *GET /appointments* mantiene, cachea por rango y aplica filtros por profesional, paciente y estado. El usuario puede cambiar entre vista, mes, semana o día, navegar por fechas y abrir un detalle en panel al hacer *clic* sobre un evento, donde se muestra el motivo de la cita, el doctor asignado y demás datos, el resultado se indica en la figura 52.

Figura 52. Interfaz de la vista a citas asignadas



En el *backend* la visualización de citas se controla con los *endpoints* en *backend/routers/appointments.py* y *GET /api/appointments/{id}*. La consulta se centra en *backend/services/appointments.py*, los parámetros se validan con *Pydantic*. Se normalizan las zonas horarias (almacenando en *UTC*), luego se consulta la capa de servicios y se responde con *DTOs* para mostrar el calendario (“empezó”, “finalizó”, “en curso”).

4.3.4.2.8. Historia de usuario 16: Registro de tratamientos

Para el registro de tratamientos se crea un formulario con requerimientos de datos nombre del medicamento, dosis, unidad, frecuencia, vía de administración, duración, instrucciones y recordatorios opcionales. El formulario es reactivo (Vue 3), usa un selector de fechas – horas y al confirmar envía un *POST /pets/id/treatments* vía *Axios*. Además, añade el tratamiento a la lista del paciente y muestra la confirmación visual, manteniendo el estado sincronizado sin recargar como se indica en la figura 53 y 54.

Figura 53. Interfaz de formulario para crear un tratamiento

Figura 54. Interfaz de respuesta a nuevo tratamiento



En *FastAPI* se expone *POST /pets/{id}/treatments* en *routers/treatments.py*; se valida el *payload* con *Pydantic* (*TreatmentCreate*), se normalizan unidades (*mg*, *ml*) y periodo de tratamiento, se vincula el tratamiento a la mascota y al médico que realiza la receta. Opcionalmente se genera el calendario de tomas/recordatorios, se persiste mediante la capa de servicios y se retorna un *DTO* ("*TreatmentRead*") consistente para que el *frontend* pinte el nuevo registro en la ficha (figura 55).

Figura 55. Código de treatment.py

```

class TreatmentBase(BaseModel):
    """Base class for Treatment"""
    name: str
    description: str
    status: str

class Treatment(TreatmentBase):
    """Treatment class"""
    name: str
    description: str
    status: str

class TreatmentHistory(BaseModel):
    """Treatment History class"""
    treatment_id: int
    user_id: int
    date: datetime
    status: str
  
```

4.3.4.3. Revisión del sprint 2

Al finalizar el desarrollo del segundo *sprint*, se realizó la reunión de finalización de *sprint* con el propietario, la misma que tuvo el propósito de analizar el avance del proyecto, la revisión del cumplimiento de las tareas asignadas a los desarrolladores y verificar la funcionalidad de las historias de usuario que conformaron el *sprint* 2, como se evidencia en el anexo 6.

4.3.4.4. Retrospectiva del sprint 2

Para la retrospectiva del segundo *sprint*, se consideraron las directrices utilizadas en el primer *sprint*, las mismas que se visualizan en la tabla 20, estas preguntas ayudaron a retroalimentar las fases del proyecto creadas y corregir falencias.

Tabla 20. Sprint 2 retrospectiva

Aspectos exitosos del sprint	Mejoras para aplicar	Errores en el sprint
La implementación de nuevos módulos en los <i>enpoints</i> de <i>FastAPI</i> permitió el desarrollo eficiente de las operaciones 'CRUD'. Además, gracias a la integración que se realizó entre <i>Vue.js</i> y el <i>backend</i> se pudo crear una interfaz fluida para registrar los tratamientos de las mascotas.	Se decidió implementar la paginación desde la base de datos <i>MySQL</i> en los momentos que se hacía la solicitud de los historiales médicos, debido a que historiales más completos podrían ralentizar el tiempo de respuesta de la aplicación.	Se detectó un error al momento de validar la zona horaria desde el <i>frontend</i> para agendar citas (HU13), lo que causaba un desfase de horas al momento de guardar la cita en la base de datos.

4.3.5. Sprint 3

4.3.5.1. Planificación del sprint 3 - (Sprint Planning)

Para la fase final del proyecto al igual que los *sprints* anteriores, se desarrolló la planificación de tareas, este *sprint* tuvo como objetivo el desarrollo de funcionalidades para

la integración del *machine learning* en la aplicación, tomando en cuenta la prioridad y estimación de cada historia.

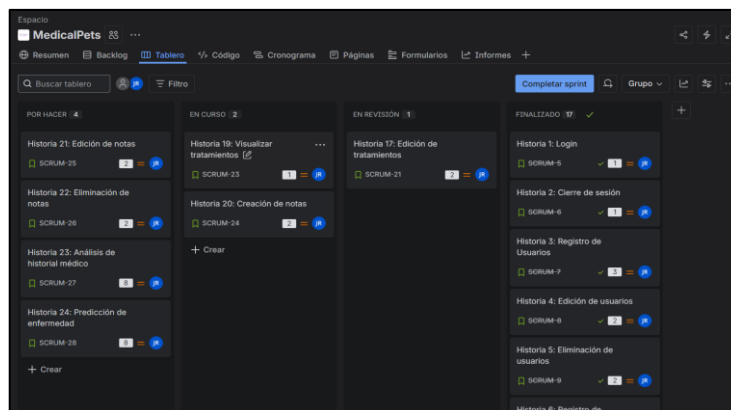
4.3.5.1.1. Velocidad de Desarrollo

Según las funcionalidades requeridas por el propietario del producto y con relación a las estimaciones para cada historia. La historia 17 y 18 cuentan con una estimación de 2 punto, la historia número 19 se valora en 1 punto, de la misma manera la historia 20, 21 y 22 se estiman en 2 puntos, finalmente la historia 23 y 24 en 8 puntos, lo que da como resultado un total de 27 puntos de historia.

4.3.5.1.2. Gestión de tareas y avances

Al igual que los *sprint* 1 y 2 para el control en el avance de las historias, se lo hizo mediante *Jira*, ya que fue de gran ayuda durante los dos primeros procesos para gestionar de manera correcta el desarrollo del proyecto. Permitted definir bien las tareas del equipo y tener un control para presentar cada proceso. En la figura 56 se presenta la estructura utilizada en el marco de trabajo.

Figura 56. Tablero de Jira – Sprint 3 (Atlassian, 2024)



4.3.5.1.3. Sprint Backlog

Para el tercer *sprint* se escogieron 8 siguientes historias, sumando un total de 27 puntos de estimación, como se visualiza en la tabla 21.

Tabla 21. Sprint Backlog 3

Sprint Backlog							
Objetivo: Desarrollar las funcionalidades para la gestión de historial médico de las mascotas, posible predicción de enfermedades							
Sprint	Historia	Est	Categoría	Tarea	Est	Responsable	Estado
SPRINT 3	HU17 - Edición de tratamientos	2	Desarrollo	Endpoint y servicio de actualización (<i>backend/routers/treatments.py</i>)	0.75	Jordan Herrera	Completado
			Desarrollo	Modal de edición y sincronización con store (<i>frontend/src/views/Treatments.vue</i>)	0.75	Jennifer Paredes	Completado
			Prueba	Verificar actualizaciones y mensajes de éxito/error	0.5	Jennifer Paredes	Completado
SPRINT 3	HU18 - Finalización de tratamientos	2	Desarrollo	Endpoint para marcar tratamiento como finalizado (<i>backend/routers/treatments.py</i>)	0.75	Jordan Herrera	Completado
			Desarrollo	Botón y lógica de finalización de la vista (<i>frontend/src/views/Treatments.vue</i>)	0.75	Jordan Herrera	Completado
			Diseño	Indicadores visuales de estado (activo/finalizado) en tarjetas	0.25	Jennifer Paredes	Completado
			Prueba	Validar finalización y cambio de estado	0.25	Jennifer Paredes	Completado
SPRINT 3	HU19 - Visualizar tratamientos	1	Desarrollo	Endpoints de consulta y filtros por estado (<i>backend/routers/treatments.py</i>)	0.5	Jordan Herrera	Completado
			Desarrollo	Grid de tarjetas y búsqueda en vista (<i>frontend/src/views/Treatments.vue</i>)	0.25	Jordan Herrera	Completado
			Prueba	Validar visualización, filtros y permisos por rol	0.25	Jennifer Paredes	Completado
SPRINT 3	HU20 - Creación de Notas	2	Desarrollo	Endpoints y validación de creación (<i>backend/routers/treatment_notes.py</i>), (<i>backend/models/treatment_note.py</i>)	0.25	Jordan Herrera	Completado
			Desarrollo	Formulario de creación con selección de tratamientos (<i>frontend/src/views/Notes.vue</i>)	0.25	Jordan Herrera	Completado
			Diseño	Modal de creación y alertas según paleta de colores previamente definida (<i>Notes.vue</i>)	0.25	Jennifer Paredes	Completado

SPRINT 3	HU21 - Edición de Notas	2	Prueba	Validar creación y mensajes estándar	0.25	Jennifer Paredes	Completado
			Desarrollo	Endpoint y servicio de actualización (<i>backend/routers/treatment_notes.py</i>)	1	Jordan Herrera	Completado
			Desarrollo	Modal de edición y sincronización store (<i>frontend/src/views/Notes.vue</i>)	0.5	Jennifer Paredes	Completado
			Prueba	Verificación actualizaciones y permisos de edición	0.5	Jennifer Paredes	Completado
SPRINT 3	HU22 - Eliminación de Notas	2	Desarrollo	Endpoint de eliminación con validaciones (<i>backend/routers/treatment_notes.py</i>)	0.75	Jordan Herrera	Completado
			Desarrollo	Confirmación y actualización del <i>gripd</i> tras eliminar (<i>frontend/src/views/Notes.vue</i>)	0.75	Jordan Herrera	Completado
			Prueba	Validar eliminación y mensajes estándar	0.5	Jennifer Paredes	Completado
SPRINT 3	HU23 – Análisis del historial médico	8	Desarrollo	Endpoint REST para comunicar <i>Frontend</i> con el motor de la IA	2	Jordan Herrera	Completado
			Desarrollo	<i>backend/routers/predictions.py</i>	2	Jordan Herrera	Completado
			Diseño	Creación de <i>templates</i> para carga de los datos y constante comunicación con los datos de entrada en tiempo real.	2	Jennifer Paredes	Completado
			Prueba	Validación y limpieza de datos de entrada en tiempo real con (<i>predictions.py</i>) localizado en la ruta <i>backend/routers/predictions.py</i>	2	Jennifer Paredes	Completado
SPRINT 3	HU24 – Predicción de enfermedad	8	Desarrollo	Llamado y correcta utilización del modelo entrenado (<i>disease_prediction_model.pkl</i>)	3	Jordan Herrera	Completado
			Desarrollo	Predicción y limpieza de datos de los pacientes (<i>backend/ml_logic/predictor.py</i>)	3	Jordan Herrera	Completado
			Prueba	Entrenamiento con <i>scikit</i> -para realizar el localizando los datos revisados en el archivo (<i>training_data.csv</i>)	2	Jennifer Paredes	Completado

Tabla 22. Sprint Backlog Técnico

Sprint Backlog						
Historia	Est	Categoría	Tarea	Responsable	Est	Estado
HT1: Seguridad y privacidad de la aplicación	8	Desarrollo	Autenticación utilizando <i>JSON Web Tokens</i> y <i>FastAPI</i>	Jordan Herrera	1	Completado
		Desarrollo	Encriptación de las contraseñas mediante métodos de <i>hashing</i> unidireccional	Jordan Herrera	1	Completado
		Desarrollo	Autorización para brindar acceso a la aplicación mediante <i>Middleware</i>	Jordan Herrera	2	Completado
		Desarrollo	Inyección de dependencia de roles	Jordan Herrera	2	Completado
		Desarrollo	Protección de rutas por medio de <i>Vue-router</i>	Jennifer Paredes	2	Completado

4.3.5.2. Sprint 3 - Reuniones diarias - Daily Scrum

Para el último *sprint* de este proyecto, se efectuaron reuniones diarias de 20 minutos, orientadas a actualizar los avances del tablero en *Jira* y retroalimentar sobre las tareas realizadas. De igual manera, se mantuvo el orden establecido en el cronograma marcado en la plataforma de gestión.

4.3.5.2.1. Historia de Usuario 17: Edición de tratamientos

Para la edición de tratamientos se implementó un formulario donde el usuario (doctor) accede con los datos precargados y se validan en tiempo real, dicho formulario está ubicado en `src/views/Treatments/EditTreatmentView.vue`. Los datos del tratamiento escogido se cargan mediante *Pinia* usando `src/stores/treatments.js` que ejecuta la actualización mediante *Axios* (figura 57 y 58).

Figura 57. Interfaz de formulario para edición de tratamientos

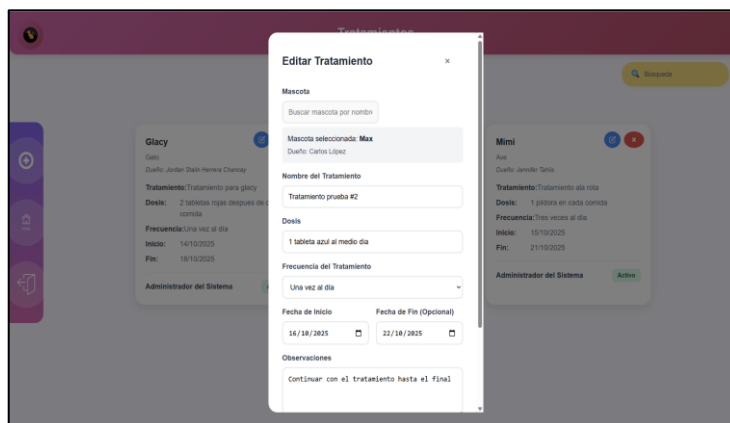
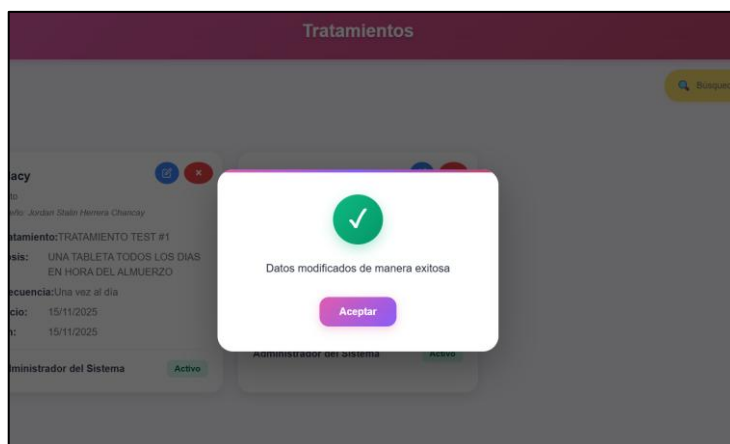
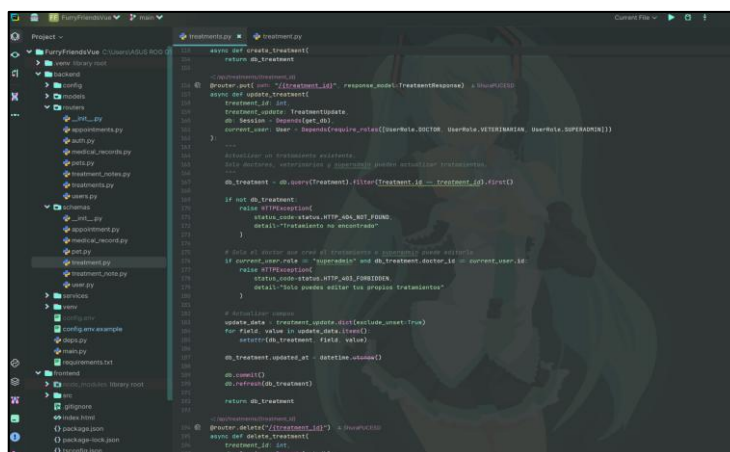


Figura 58. Interfaz de respuesta a edición de tratamientos



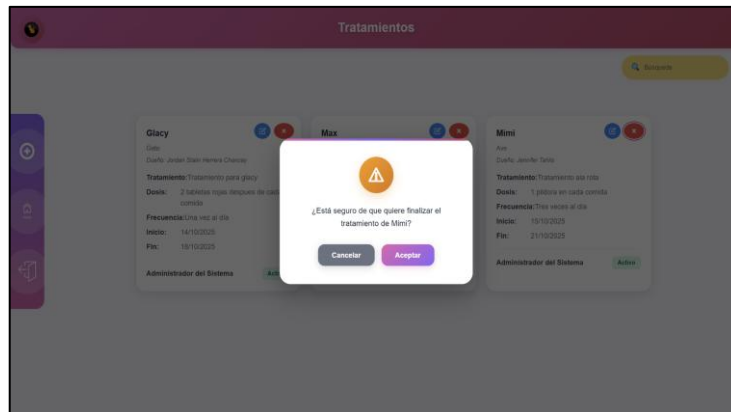
Por el lado del *backend*, la actualización de un tratamiento se gestiona mediante `/pets/{id}/treatments/{treatment_id}`, allí los datos se validan con `TreatmentUpdate` y se actualizan en la tabla `treatments`, como se visualiza en la figura 59. Al editar el tratamiento se envía un “`PATCH /pets/id/treatments/treatmentId`” para mantener el flujo del proceso sin recargar la vista.

Figura 59. Código de `treatments.py`

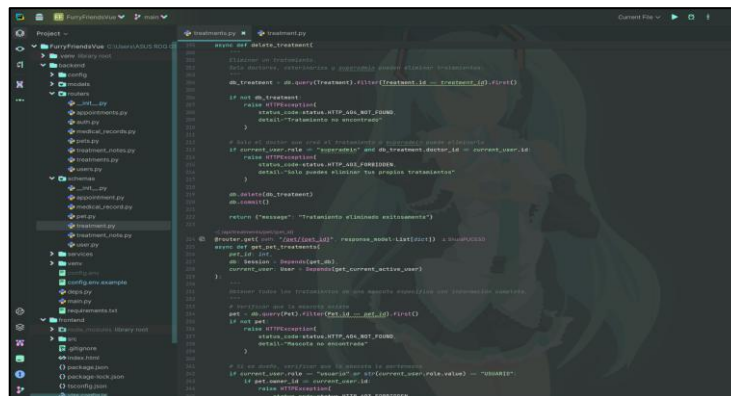
4.3.5.2.2. Historia de Usuario 18: Finalización de tratamientos

Para la historia de dar por terminado el tratamiento de alguna mascota, se realiza desde `src/views/Treatments/TreatmentDetailView.vue`, cuando el usuario (doctor) ejecuta la acción se activa `treatmentsStore.finishTreatment()`. Al confirmar que sea desea finalizar el tratamiento, se envía un `PATCH /pets/id/treatments/treatmentId` actualizando el campo de estado a “Finalizado”. La petición es enviada vía `Axios`, con la autenticación `JWT` (figura 60).

Figura 60. Interfaz de finalizar un tratamiento



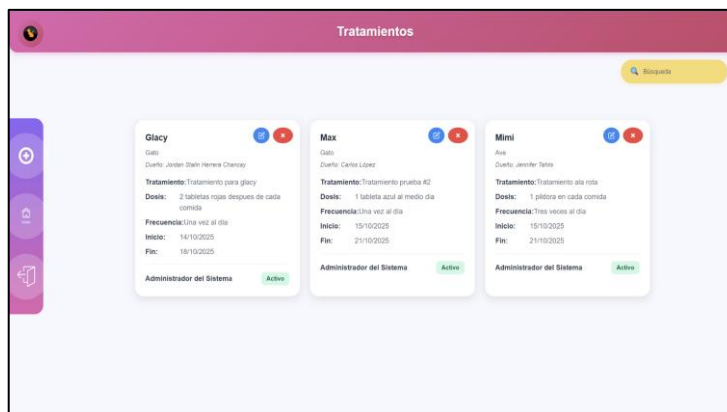
En *FastAPI* el final de un tratamiento se gestiona a través del *endpoint PATCH* `/pets/{id}/treatments/{treatment_id}` en `routers/treatments.py`. La ruta valida el *payload* con *Pydantic*, luego se delega a la capa de servicios para actualizar el estado, registrar la fecha de finalización y guardar el cambio, se evidencia en la figura 61. Las consultas de *GET* `/pets/{id}/treatments` excluyen por defecto los tratamientos que están en estado *finished* (finalizado).

Figura 61. Código de `treatments.py` – `delete_treatment`

4.3.5.2.3. Historia de Usuario 19: Visualización de tratamientos

La visualización de tratamientos se centra en una lista clara en de la ficha del paciente, con filtros por estado activos o finalizados, posee un filtro de búsqueda por texto, así como por fecha. Al entrar, el *store* de *Pinia* solicita los datos con *GET* `/pets/id/treatments` luego cachea los resultados para una navegación fluida y permite alternar rápidamente entre “activo” y “finalizado”. Además, cada uno cuenta con nombre del medicamento, dosis de aplicación y los indicadores del estado del tratamiento en el panel, en la figura 62 se ilustra el resultado.

Figura 62. Interfaz de la lista de tratamientos



En *FastAPI* se exponen *GET /pets/{id}/treatments* (listado con filtros de estado, paginación y orden) y *GET /pets/{id}/treatments/{treatment_id}*, estos parámetros se validan con *Pydantic* para normalizar fechas almacenadas en *UTC*. Se consulta la capa de servicios y se devuelven *DTOs* para la *UI*, allí por defecto el listado muestra solo los tratamientos activos y los que ya finalizaron se incluyen si el usuario lo solicita mediante el filtro.

4.3.5.2.4. Historia de Usuario 20: Creación de Notas

En las notas se puede especificar las novedades con la mascota, para eso se creó un formulario *src/views/TreatmentNotes/CreateNoteView.v*. Al igual que los formularios de las historias anteriores el proceso es manejado por *Pinia* a través de *src/stores/treatmentNotes.js*, el cual envía un *POST /pets/id/notes* usando *Axios*. El *store* de *Pinia* maneja “*isSaving*”, errores y la actualización de la cache para mejor fluidez de la aplicación (figura 63 y 64).

Figura 63. Interfaz para creación de notas

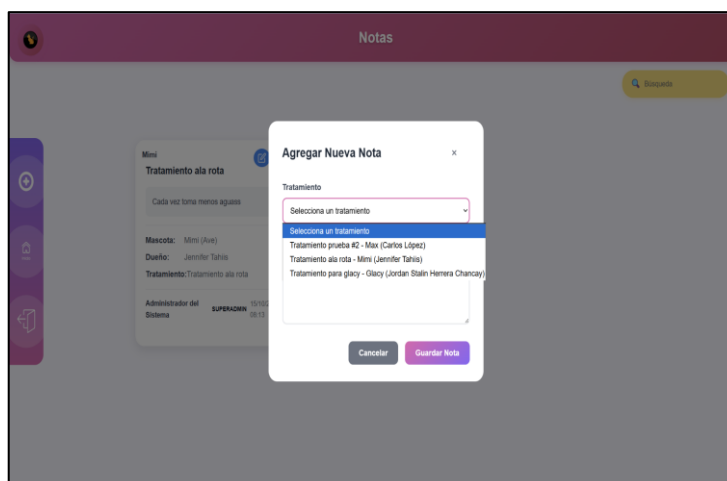
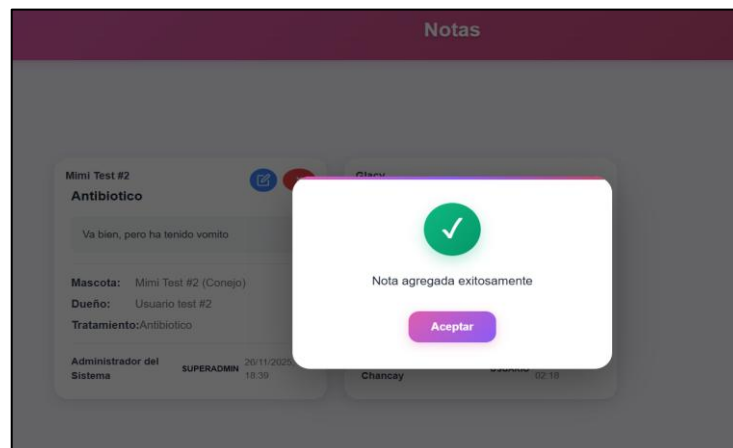
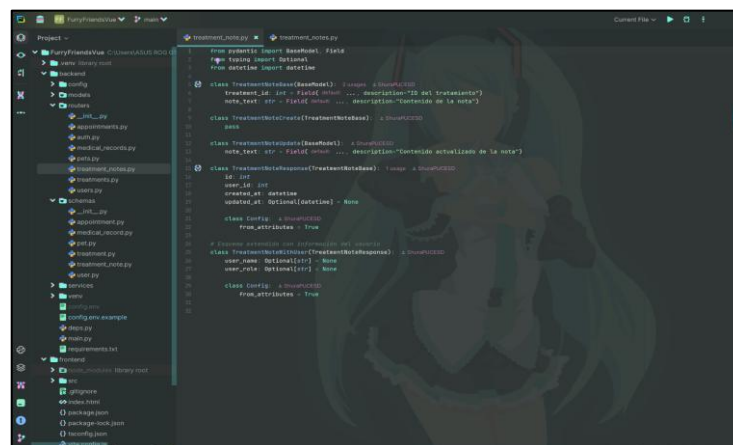


Figura 64. Interfaz de respuesta a creación de notas



En el *backend* se publica `POST /pets/{id}/notes` en `routers/notes.py`, después se valida el cuerpo con *Pydantic* para asociar la nota al cliente y al usuario que la creó. Además, se verifican campos de fechas que se procesan adjuntos con *UploadFile* y la respuesta devuelve un *DTOs* para que el *frontend* agregue la nota recién creada en la vista (figura 65).

Figura 65. Código de `treatment_note.py` para crear notas

4.3.5.2.5. Historia de Usuario 21: Edición de Notas

En la edición de una nota, los cambios se validan automáticamente para que al momento de guardar se envíe un `PATCH /pets/id/notes/notelid` con *Axios* y *JWT* por interceptor. También, se deshabilita acciones, se aplica *optimistic UI* para ver el ajuste al instante con *rollback*, en caso de que algo falla, el *store* de *Pinia* mantiene el estado y sincroniza la lista sin recargar, el resultado se indica en las figuras 66 y 67.

Figura 66. Interfaz para edición de notas

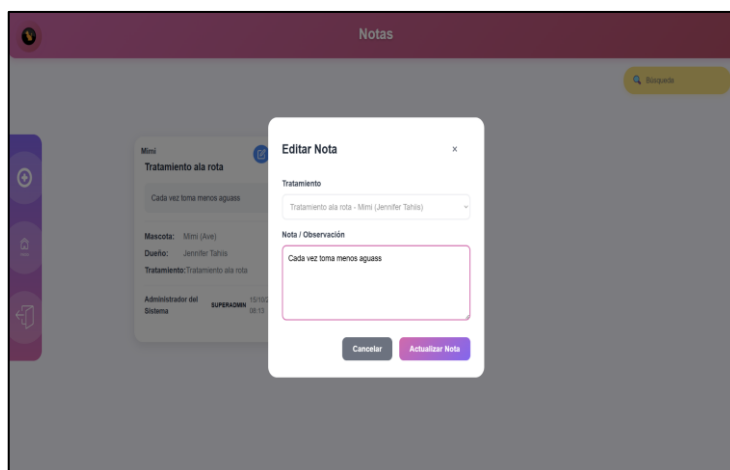
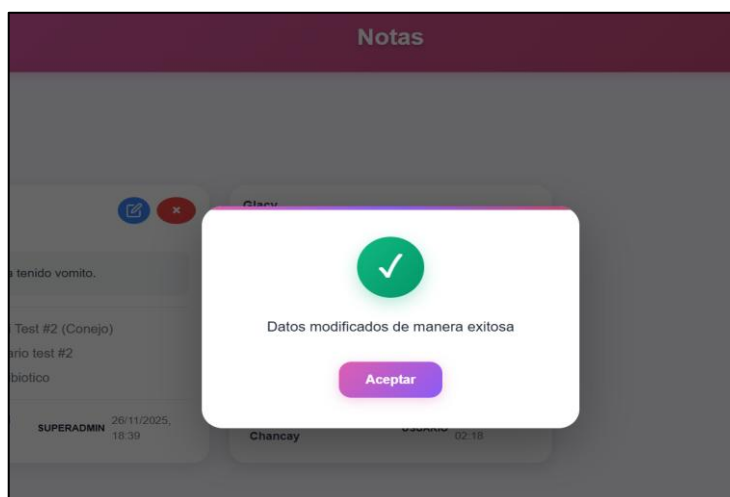


Figura 67. Interfaz de respuesta a edición de notas



Por el lado del *backend*, la actualización del contenido de una nota que se creó se implementa a través del *endpoint* `PATCH/api/treatments/{treatment_id}/notes/{note_id}`, en `routers/notes.py`, se valida los datos con *Pydantic*, la respuesta devuelve un *DTO* para que el *frontend* refleje la nota actualizada inmediatamente, como se muestra en la figura 68.

El proceso garantiza que las notas clínicas puedan ser corregidas y complementadas de manera correcta por el usuario, manteniendo la integridad de los datos en el historial médico de la mascota.

Figura 68. Código de treatment_notes.py

```

107     async def create_treatment_note(
108         db: AsyncSession, note: TreatmentNote
109     ):
110         return db_note
111
112     @staticmethod
113     @router.put("/{id}/{id2}", response_model=TreatmentNoteResponse)
114     async def update_treatment_note(
115         note_id: int,
116         note_update: TreatmentNoteUpdate,
117         db: AsyncSession = Depends(get_db),
118         current_user: User = Depends(get_current_active_user)
119     ):
120         # Actualizar una nota de tratamiento
121         note = await db.get(TreatmentNote, note_id)
122         # Solo si pertenece de la nota = @dependencia de la nota
123         db_note = db.query(TreatmentNote).filter(
124             TreatmentNote.id == note_id, TreatmentNote.is_active == True).first()
125         if not db_note:
126             raise HTTPException(
127                 status_code=HTTP_404_NOT_FOUND,
128                 detail="Nota no encontrada"
129             )
130         # Solo si el creador de la nota = @dependencia de la historia
131         if current_user.role == "superadmin" and db_note.user_id != current_user.id:
132             raise HTTPException(
133                 status_code=HTTP_403_FORBIDDEN,
134                 detail="Solo puedes editar tus propias notas"
135             )
136         # Actualizar los datos de la nota
137         db_note.note_text = note_update.note_text
138         db_note.updated_at = datetime.utcnow()
139         db.commit()
140         db.refresh(db_note)
141         return db_note
142
143     @staticmethod
144     @router.delete("/{id}/{id2}")
145     async def delete_treatment_note(
146         note_id: int,
147         db: AsyncSession = Depends(get_db),
148         current_user: User = Depends(get_current_active_user)
149     ):
150

```

4.3.5.2.6. Historia de Usuario 22: Eliminación de Notas

Para la historia de eliminación de notas para los tratamientos se debe ejecutar la vista *src/views/Treatments/TreatmentDetailView.vue*, en donde cada nota ligada a un tratamiento indica una opción de borrar que a su vez despliega un cuadro de confirmación mediante *src/components/ModalConfirm.vue*. Después de confirmar la acción, se invoca a la función *deleteNote* el cual envía una petición *DELETE* al servidor usando *Axios*. Se bloquea el botón mientras se procesa y se aplica *optimistic UI* para retirar la nota de la vista del usuario (figura 69 y 70).

Figura 69. Interfaz de eliminación de notas

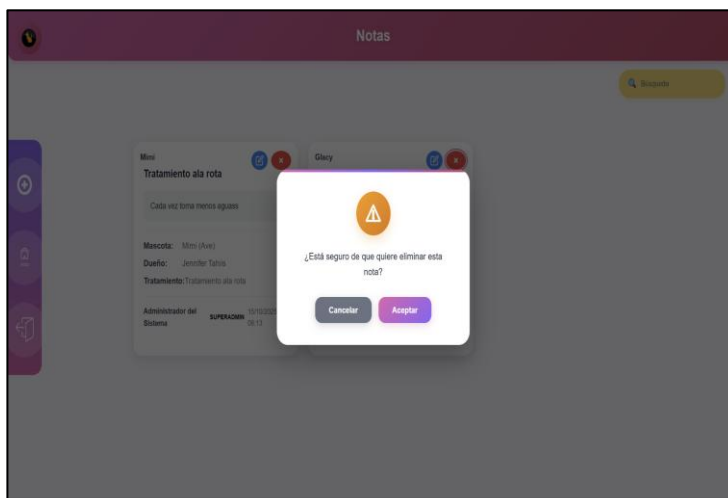
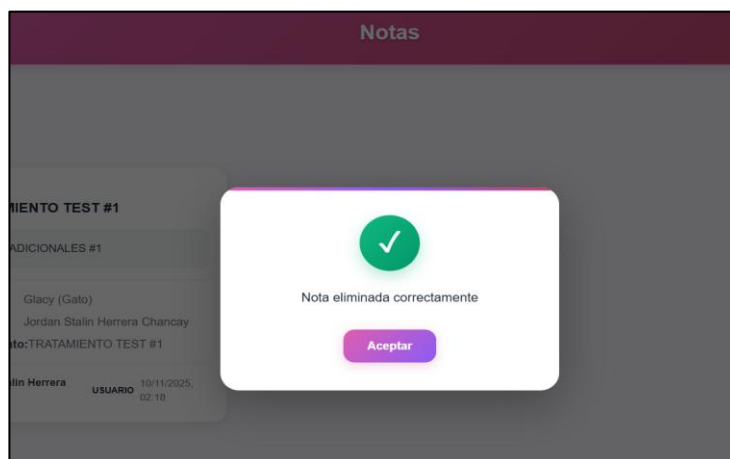
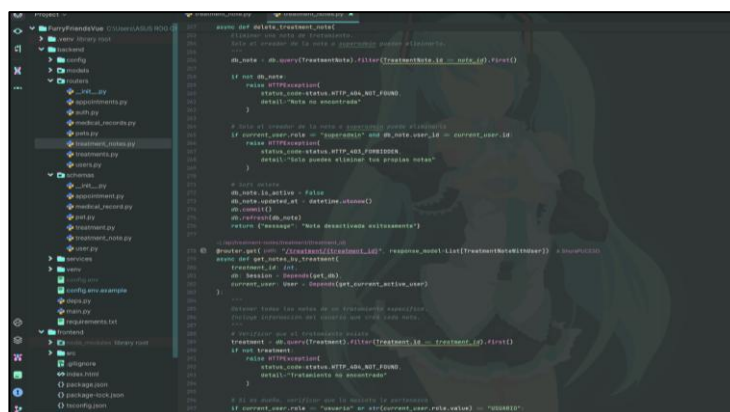


Figura 70. Interfaz de respuesta a eliminación de notas



En *FastAPI* se publica `DELETE /pets/{id}/notes/{note_id}` en `routers/notes.py`, allí la ruta resuelve los identificadores, delega a la capa de servicios para ejecutar el borrado y gestiona el proceso de eliminación, garantizando la consistencia del historial médico, como indica el código en la figura 71.

Figura 71. Código de `treatment_notes.py` - DELETE

4.3.5.2.7. Historia de Usuario 23: Análisis del Historial Médico ML

Se optó por utilizar aprendizaje supervisado, este enfoque permite entrenar al modelo para que identifique patrones y posteriormente realice predicciones sobre posibles casos, mejorando la toma de decisiones dentro del sistema.

Para la historia del análisis del historial médico, se creó un *endpoint Rest* para comunicar el *Frontend* con el motor de la *IA* previamente entrenado (*predictor.py*). Este proceso se pudo realizar gracias a el archivo '*predictions.py*' localizado en la ruta

`backend/routers/predictions.py`, mismo que ayuda en la obtención el *ID* de la mascota para poder consultar el historial médico registrado en la base de datos, se muestra en la figura 72.

Cabe recalcar que, se toman en cuenta datos como la raza, el peso y la edad del animal para evitar que aprenda con ruido (*overfitting*). Una vez se obtienen los datos mencionados son enviados al archivo `predictor.py` para realizar la posible predicción y retornar una respuesta al *frontend* mediante un objeto *JSON*.

Figura 72. Código predictions.py

```

def predict_disease(pet_id):
    """
    Predict disease based on pet ID, species, age, and weight.
    """
    # Fetch pet details
    pet = db.session.query(Pet).filter(Pet.id == pet_id).first()
    if not pet:
        return HTTPException(status_code=404, detail="Pet not found")

    # Fetch medical history
    records = db.session.query(MedicalRecord).filter(MedicalRecord.pet_id == pet_id).all()

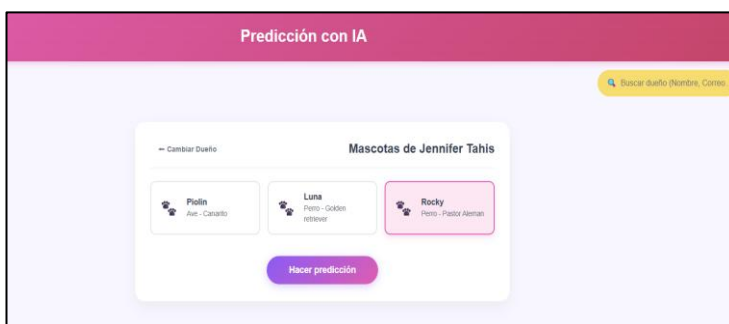
    # Prepare data for model
    features = []
    for record in records:
        if record.symptoms:
            symptoms_list = record.symptoms.split(',')
            symptoms_text = ','.join(symptoms_list)

            # Prepare data for model
            features.append({
                'species': pet.species if pet.species else 'Other', # Default for Pet's of unknown or multiple species
                'age': pet.age_years if pet.age_years else 5, # Default age of 5 years
                'weight': pet.weight if pet.weight else 10.0, # Default weight of 10kg
                'symptoms_text': symptoms_text
            })

    # Predict
    try:
        result = predict_disease(features)
    
```

Por otro lado, el usuario puede interactuar en la aplicación en la que funciona primero utilizando el filtro de búsqueda para localizar al dueño de la mascota, (esto porque un cliente puede ser dueño de varias mascotas), luego el doctor deberá seleccionar la mascota para que analice sus datos.

Figura 73. Interfaz de Análisis del Historial Médico ML.



4.3.5.2.8. Historia de Usuario 24: Predicción de Enfermedades

Para la historia número 24, el principal actor es el archivo *'predictor.py'*, que se encuentra ubicado en la ruta *backend/ml_logic/predictor.py*, es el encargado de gestionar el modelo entrenado *'disease_prediction_model.pkl'* y la generación de posibles predicciones en tiempo real. Los datos obtenidos gracias a *'predictions.py'* sirven para realizar la posible predicción, mediante la implementación de *post*-procesamiento que está encargado de recalcular los porcentajes basándose en 3 resultados probables para alguna posible enfermedad que pueda ser diagnosticada a la mascota antes seleccionada.

Por otra parte, en esta historia se lleva a cabo la reutilización de *'VeterinaryPreprocessor'*, para garantizar que los datos de entrada desde la *web* reciban el mismo tratamiento de limpieza y formateo que los datos de entrenamiento.

Para la obtención de los recursos para entrenar el modelo predictor, se utilizó *Deep Search* de *Gemini* para recolectar información sobre enfermedades en animales domésticos disponible en línea. Se creó una enciclopedia médica estructurada a partir de esta información, que se utilizó como guía para generar datos sintéticos con el propósito de entrenar el modelo.

Después de recopilar y organizar la información, se estableció una carpeta llamada *"scripts"*, donde se guardan los datos de las características de las mascotas. En la figura 74 se expone el archivo que sirvió como enciclopedia médica, para la creación y preparación de los datos para empezar a entrenarlos.

Figura 74. Código *diseases_db.py*

```

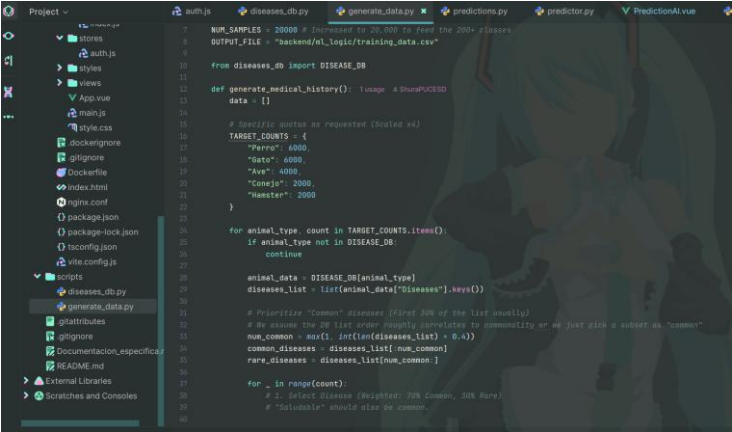
diseases_db.py
"""
Difteria: ["Labrador", "Golden Retriever", "Pastor Alemán", "Bulldog", "Beagle", "Poodle", "Chihuahua", "Borquis Terrier", "Boxer", "Dachshund", "Puli", "Weimaraner"]
Escarlatina: ["Pastor Alemán", "Golden Retriever", "Labrador", "Bulldog", "Beagle", "Poodle", "Chihuahua", "Borquis Terrier", "Boxer", "Dachshund", "Puli", "Weimaraner"]
Fiebre tifoidea: ["Pastor Alemán", "Golden Retriever", "Labrador", "Bulldog", "Beagle", "Poodle", "Chihuahua", "Borquis Terrier", "Boxer", "Dachshund", "Puli", "Weimaraner"]
...
"""

```

En *generate_data.py*, se desarrolla el proceso de la generación de datos sintéticos, a través de “*scripts*” se genera aproximadamente 2000 registros de datos relacionados con enfermedades que pueden sufrir las mascotas. El procedimiento se ejecuta mediante un bucle iterativo que continua hasta alcanzar el número de registro establecidos. En cada iteración se escoge una enfermedad mediante un valor probabilístico que establece si la mascota está sana, tiene una posible enfermedad común o padece una posible enfermedad poco frecuente.

Después, se les atribuyen atributos realistas como la raza, el peso y la edad de la mascota. Por último, se generan los síntomas asociados, eligiendo de manera aleatoria entre tres y nueve síntomas vinculados con la enfermedad ya establecida. Pese a que esta selección aleatoria puede hacer que los datos contengan un ruido mínimo, este aspecto es ventajoso en el proceso de entrenamiento del modelo de *machine learning*, pues ayuda a aumentar la capacidad del modelo para generalizar y disminuye el peligro de sobreajuste (*overfitting*). Para evaluar el modelo, se utilizaron métricas de clasificación: *accuracy* (exactitud) que permite medir el porcentaje de las posibles predicciones.

Figura 75. Código *generate_data.py*



```

NUM_SAMPLES = 20000 # Increased to 20,000 to fill the 200+ classes
OUTPUT_FILE = "backend/ml_logic/training_data.csv"

from diseases_db import DISEASE_DB

def generate_medical_history():
    """Generate medical history for a given animal type"""
    data = []

    # Specify quotes as requested (located at)
    TARGET_COUNTS = {
        "kitten": 4000,
        "dog": 4000,
        "ave": 4000,
        "conejo": 2000,
        "hamster": 2000
    }

    for animal_type, count in TARGET_COUNTS.items():
        if animal_type not in DISEASE_DB:
            continue

        animal_data = DISEASE_DB[animal_type]
        diseases_list = list(animal_data["diseases"].keys())

        # Prioritize "common" diseases (first 20% of the list usually)
        # We assume the 20 list items already belonging to "common" or we just pick a subset as "common"
        num_common = max(1, int(len(diseases_list) * 0.4))
        common_diseases = diseases_list[:num_common]
        rare_diseases = diseases_list[num_common:]

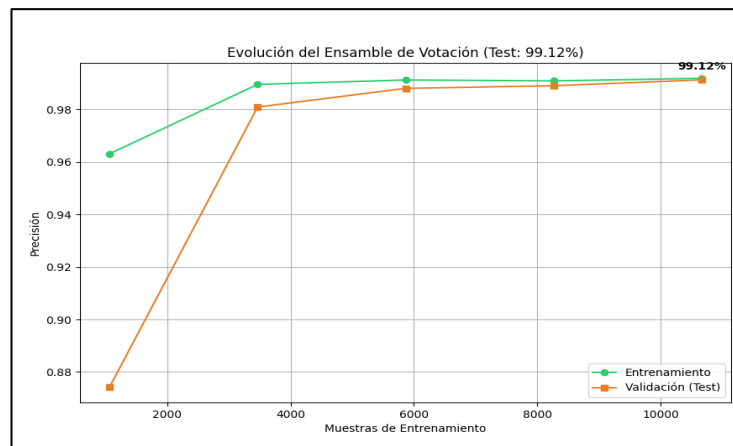
        for _ in range(count):
            # 1. Select disease (weighted: 70% Common, 30% Rare)
            # 2. Select disease (weighted: 70% Common, 30% Rare)
            # 3. Selectable should also be common.

```

Por otra parte, la creación del modelo entrenado se centró en el archivo “*train_model.py*” ubicado en la ruta *backend/ml_logic/train_model.py*, el mismo que usa la librería de *scikit* para realizar el entrenamiento que funciona localizando los datos revisados anteriormente que se encuentran en el archivo *training_data.csv*. Se los dividió

en dos grupos en donde el 80% se utilizaron para el entrenamiento y el 20% restante para realizar las pruebas y verificar si la curva de aprendizaje está estrechamente relacionada con los *test*. En la figura 76, se puede verificar que la aplicación realiza el respectivo entrenamiento y *test* de manera satisfactoria, dando como resultado un porcentaje de confianza bastante alto.

Figura 76. Librería matplotlib.pyplot



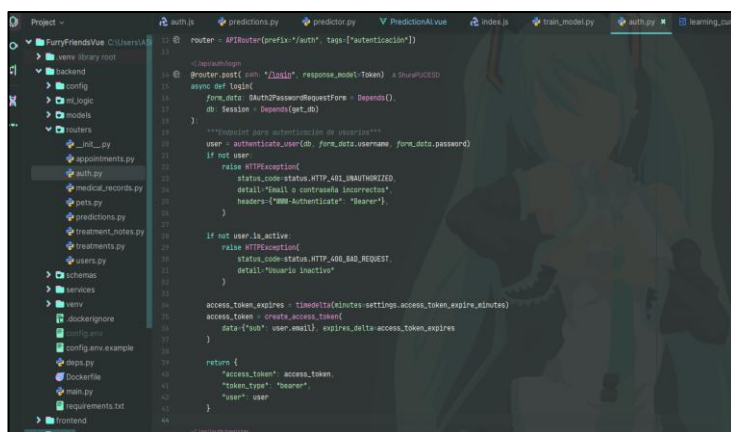
Para conseguir el resultado que se expone en la figura 76, se implementaron dos algoritmos dentro de un árbol de decisión:

- **Regresión logística:** Modelo encargado de separar las enfermedades si encontrar relación con 1000 casos de parvovirus y 10 de rabia, el modelo le presta más atención a la rabia para no descartarla, adicional el algoritmo tiene 3000 intentos para encontrar antes de rendirse temporalmente.
- **Random Forest:** Basado en arboles de decisiones, se inicia creando un bosque de 300 árboles de decisiones diferentes en donde cada árbol puede recopilar datos como si tiene fiebre, vómitos, etc. Tiene una técnica de protección por si los arboles no llegan a una conclusión con al menos 3 casos similares que lo respalden.

Siguiendo el contexto, se usó el método de optimización:

mantener la sesión activa de los mismos, mientras utilizan la aplicación. Cuando un usuario realiza un inicio de sesión exitoso, el *endpoint* definido en la ruta *backend/routers/auth.py* a través del *framework FastAPI* permite la generación de un token. También, se encapsulan los datos esenciales del usuario para permitir el acceso a la aplicación ya que cuenta con un tiempo limitado de expiración (figura 79).

Figura 79. Código backend/routers/auth.py para Tokens



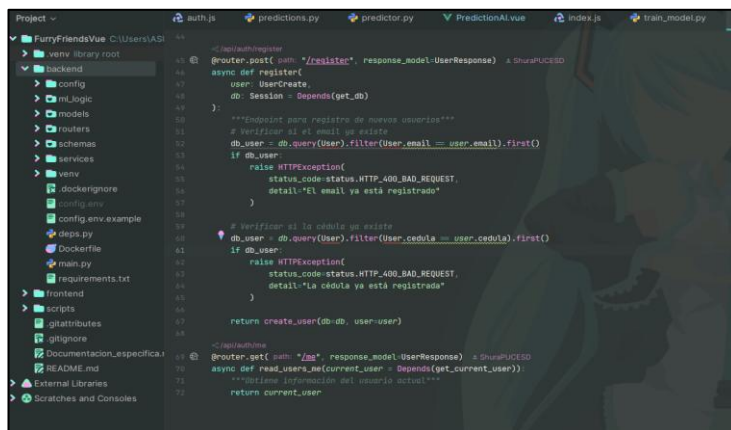
```

17 router = APIRouter(prefix="/auth", tags=["autenticación"])
18
19 @router.post(path="/login", response_model=UserResponse) &lt;div data-bbox="310 275 770 467" style="background-color: #f0f0f0; padding: 5px;">
20     async def login(
21         form_data: AuthPasswordRequestForm = Depends(),
22         db: Session = Depends(get_db)
23     ):
24         """Endpoint para autenticación de usuarios"""
25         user = authenticate_user(db, form_data.username, form_data.password)
26         if not user:
27             raise HTTPException(
28                 status_code=status.HTTP_401_UNAUTHORIZED,
29                 detail="Email o contraseña incorrectos",
30                 headers={"WWW-Authenticate": "Bearer"},
31             )
32         if not user.is_active:
33             raise HTTPException(
34                 status_code=status.HTTP_400_BAD_REQUEST,
35                 detail="Usuario inactivo",
36             )
37         access_token_expire = timedelta(minutes=settings.access_token_expire_minutes)
38         access_token = create_access_token(
39             data={"sub": user.email}, expires_delta=access_token_expire
40         )
41         return {
42             "access_token": access_token,
43             "token_type": "bearer",
44             "user": user
45         }

```

Para brindar mayor seguridad a los usuarios las contraseñas no se guardan en texto plano en la base de datos de *MySQL*, en su lugar se empleó medidas de *hashing* iterativo para la encriptación unidireccional de la contraseña antes de su persistencia. Es en el proceso de registro donde el sistema genera una cadena aleatoria (*salt*) que se combina con la clave original dando como resultado un *hash* criptográfico (figura 80).

Figura 80. Código auth/register para contraseñas



```

46 @router.post(path="/register", response_model=UserResponse) &lt;div data-bbox="310 685 770 876" style="background-color: #f0f0f0; padding: 5px;">
47     async def register(
48         user: UserCreate,
49         db: Session = Depends(get_db)
50     ):
51         """Endpoint para registro de nuevos usuarios"""
52         # Verificamos si el email ya existe
53         db_user = db.query(User).filter(User_email == user.email).first()
54         if db_user:
55             raise HTTPException(
56                 status_code=status.HTTP_400_BAD_REQUEST,
57                 detail="El email ya está registrado",
58             )
59         # Verificamos si la contraseña coincide
60         db_user = db.query(User).filter(User_cedula == user.cedula).first()
61         if db_user:
62             raise HTTPException(
63                 status_code=status.HTTP_400_BAD_REQUEST,
64                 detail="La cédula ya está registrada",
65             )
66         return create_user(db=db, user=user)
67
68 @router.get(path="/info", response_model=UserResponse) &lt;div data-bbox="310 685 770 876" style="background-color: #f0f0f0; padding: 5px;">
69     async def read_users_me(current_user = Depends(get_current_user)):
70         """Muestra información del usuario actual"""
71         return current_user

```

Por otra parte, el tipo de control de acceso y autorización que se tiene a nivel de servidor se gestiona mediante las dependencias en el archivo `'deps.py'`. Donde se implementan un *middleware* al ir interceptando las peticiones entrantes, extrayendo el *token* de la cabecera *HTTP* autorizada y validando su firma como su vigencia, el código se evidencia en la figura 81.

Figura 81. Archivo `deps.py` para peticiones

```

Project - FurryFriendsVue C:\Users\ASD
├── .venv library root
├── backend
│   ├── config
│   ├── ml_logic
│   ├── models
│   ├── routers
│   ├── schemas
│   └── services
├── dockerignore
├── config.env
├── config.env.example
├── deps.py
├── Dockerfile
├── main.py
├── requirements.txt
├── frontend
├── scripts
├── gitattributes
├── gitignore
├── Documentacion_especifica
├── README.md
├── External Libraries
└── Scratches and Consoles

auth.js  predictions.py  predictor.py  PredictionAI.vue  index.js  train_model.py
1 from fastapi import Depends, HTTPException, status
2 from fastapi.security import HTTPBearer, HTTPAuthorizationCredentials
3 from sqlalchemy.orm import Session
4 from typing import List
5 from jose import JWError, jwt
6 from datetime import datetime, timedelta
7
8 from config.database import get_db
9 from config.settings import settings
10 from models.user import User
11
12 # Configuración de seguridad
13 security = HTTPBearer()
14
15 def get_current_user(token: str = Depends(security)):
16     credentials = HTTPAuthorizationCredentials.from_header(token)
17     db = Session(get_db)
18     user = db.query(User).filter(User.id == credentials.username).first()
19     if not user:
20         raise HTTPException(status_code=401, detail="No se pudieron validar las credenciales")
21     return user
22
23 """
24 Recuperar el usuario actual basado en el token JWT.
25 """
26 credentials_exception = HTTPException(
27     status_code=status.HTTP_401_UNAUTHORIZED,
28     detail="No se pudieron validar las credenciales",
29     headers={"WWW-Authenticate": "Bearer"},
30 )
31
32 try:
33     token = credentials.credentials
34     payload = jwt.decode(token, settings.secret_key, algorithms=[settings.algorithm])
35     email_str = payload.get("sub")
36     if email is None:
37         raise credentials_exception
38 except:
39     raise credentials_exception

```

A pesar que la aplicación tiene distintas funcionalidades los usuarios tienen acceso a módulos específicos según su rol. Esto se aplica con el fin de garantizar la seguridad de la aplicación y brindar una experiencia personalizada de acuerdo al usuario y sus necesidades. Para realizar dicho control de roles, se apoya en el archivo `'auth.js'` que se encuentra en la ruta `frontend/stores/auth.js` en donde se puede configurar el tipo de rol que tiene cada usuario (figura 82).

Figura 82. Código `auth.js` para conceder permisos

```

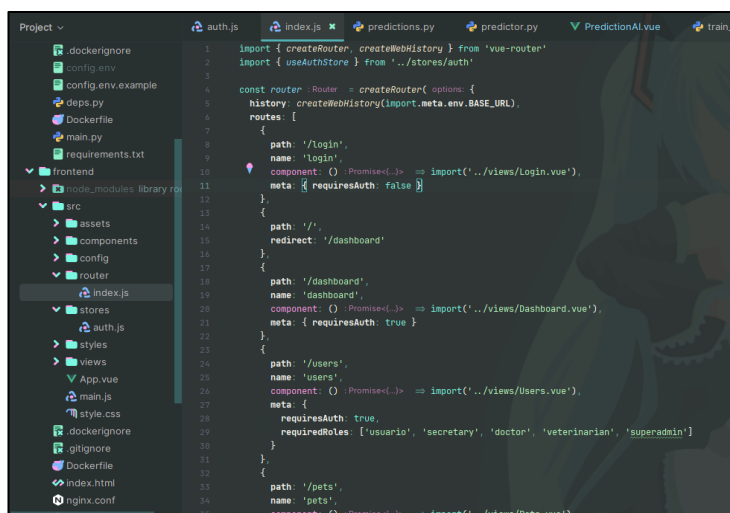
Project - FurryFriendsVue C:\Users\ASD
├── models
├── routers
├── Pull Requests
├── services
├── venv
├── dockerignore
├── config.env
├── config.env.example
├── deps.py
├── Dockerfile
├── main.py
├── requirements.txt
├── frontend
├── node_modules library
├── src
├── assets
├── components
├── config
├── hooks
├── auth.js
├── styles
├── views
├── App.vue
├── main.js
├── style.css
├── dockerignore
├── README.md

auth.js  predictions.py  predictor.py  PredictionAI.vue  train_model.py  auth.py  deps.py  learning_curve.png
1 import { defineStore } from 'pinia'
2 import { ref, computed } from 'vue'
3 import api, { endpoints } from '../configs/api'
4
5 export const useAuthStore = defineStore('auth', {
6     state: () => ({
7         token: ref(''),
8         user: ref({}),
9         isAuthenticated: ref(false),
10    }),
11    actions: {
12        login({ email, password }) {
13            return api.post(endpoints.login, { email, password })
14        },
15        logout() {
16            return api.delete(endpoints.logout)
17        },
18        refresh() {
19            return api.post(endpoints.refresh)
20        },
21        validateToken() {
22            return api.get(endpoints.validateToken)
23        },
24        validateUser() {
25            return api.get(endpoints.validateUser)
26        },
27    },
28    getters: {
29        isAuthenticated: () => this.isAuthenticated,
30        user: () => this.user,
31        token: () => this.token,
32    },
33    persist: true,
34 })
35
36 const useRole = defineStore('role', {
37     state: () => ({
38         role: ref(''),
39     }),
40     actions: {
41        setRole(role) {
42            this.role = role
43        },
44    },
45    getters: {
46        role: () => this.role,
47    },
48    persist: true,
49 })
50
51 // Función para validar los permisos
52 const hasPermission = (requiredRole) => {
53     return (userRole) => {
54         if (!isAuthenticated.value || !user.value) return false
55         if (isSuperAdmin.value) return true
56         if (isVeterinarian.value) return true
57         if (isDoctor.value) return true
58         if (isSecretary.value) return true
59         if (isUser.value) return true
60         return false
61     }
62 }
63
64 // Función para validar los permisos
65 const validatePermissions = (requiredRoles) => {
66     return (userRole) => {
67         if (!isAuthenticated.value || !user.value) return false
68         if (isSuperAdmin.value) return true
69         if (isVeterinarian.value) return true
70         if (isDoctor.value) return true
71         if (isSecretary.value) return true
72         if (isUser.value) return true
73         return false
74     }
75 }
76
77 const login = async (email, password) => {
78     const response = await useAuthStore().login({ email, password })
79     if (response.status === 200) {
80         useAuthStore().token = response.token
81         useAuthStore().user = response.user
82         useAuthStore().isAuthenticated = true
83         useRole().setRole(response.user.role)
84     }
85 }

```

Finalmente, con la intención de resguardar la información del usuario se implementó la capacidad de evaluar cada transición entre vistas. De tal manera que si una página requiere autenticación y el gestor de estado global no localiza un *token* válido en el almacenamiento local, a este mismo se le interrumpe la navegación y es redirigido a la página del *Login* para que el usuario pueda ingresar nuevamente. Esto se hizo mediante *router.beforeEach* ubicado en el archivo *index.js* de la ruta *frontend/router/index.js* (figura 83).

Figura 83. Código archivo *index.js*



```

1 import { createRouter, createWebHistory } from 'vue-router'
2 import { useAuthStore } from '../stores/auth'
3
4 const router = Router = createRouter({
5   history: createWebHistory(import.meta.env.BASE_URL),
6   routes: [
7     {
8       path: '/login',
9       name: 'login',
10      component: () => import('../views/Login.vue'),
11      meta: { requiresAuth: false }
12    },
13    {
14      path: '/',
15      redirect: '/dashboard'
16    },
17    {
18      path: '/dashboard',
19      name: 'dashboard',
20      component: () => import('../views/Dashboard.vue'),
21      meta: { requiresAuth: true }
22    },
23    {
24      path: '/users',
25      name: 'users',
26      component: () => import('../views/Users.vue'),
27      meta: {
28        requiresAuth: true,
29        requiredRoles: ['usuario', 'secretary', 'doctor', 'veterinarian', 'superadmin']
30      }
31    },
32    {
33      path: '/pets',
34      name: 'pets',
35      component: () => import('../views/Pets.vue')
36    }
37  ]
38 })
  
```

4.3.5.3. Revisión del sprint 3

Una vez finalizado el desarrollo del *sprint* 3, se llevó a cabo la reunión de fin de *sprint* con el propietario, esta reunión tuvo como propósito la revisión del *sprint*, validar que las historias estén realizadas por completo y comprobar la funcionalidad de las mismas (anexo 6). Además, se hizo entrega e implementación de la aplicación *web* en el Hospital Veterinario Santa Martha, los detalles se muestran en el anexo 13. De igual forma se entregó el manual de usuario que se encuentra en el anexo 7.

4.3.5.4. Retrospectiva del sprint 3

Para la retrospectiva del tercer *sprint*, se consideraron las preguntas utilizadas en los *sprints* anteriores, las mismas que se visualizan en la tabla 23.

Tabla 23. Sprint 3 retrospectiva

Aspectos exitosos del sprint	Mejoras para aplicar	Errores en el sprint
La integración de la lógica de posible predicción para las enfermedades de las mascotas dentro de la arquitectura de <i>FastAPI</i> , resultó eficiente para el procesamiento del análisis de los historiales médicos.	Se identificó la necesidad de implementar a futuro un cache temporal para almacenar los historiales médicos analizados.	Se detectó un problema de reactividad en la edición de los tratamientos (HU17), lo que impidió precargar correctamente la información previa en los formularios. Este error fue depurado de manera exitosa.

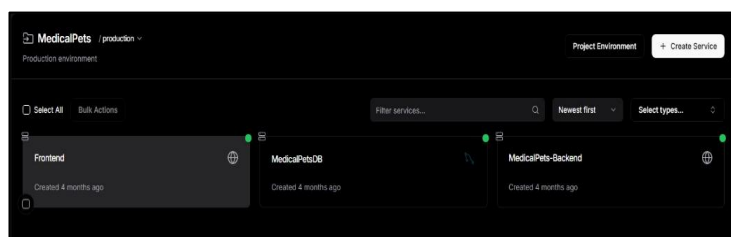
4.4. Resultado general

El resultado del proyecto, se abarcó con respecto al desarrollo y la implementación en la plataforma correspondiente para la gestión de historiales médicos de mascotas.

4.4.1. Implementación de aplicación web con machine learning.

Para la implementación de la aplicación *web* con *machine learning*, se adoptó una arquitectura en tres capas basada en contenedores, fragmentando el sistema en *frontend*, *backend* y base de datos. Este modelo centralizado garantiza alta disponibilidad y permite escalabilidad. Internamente, el *backend* implementa el patrón de diseño Modelo-Vista-Controlador (MVC), facilitando la separación de responsabilidades y el mantenimiento del sistema.

Figura 84. Estructura de la aplicación dividida en contenedores.

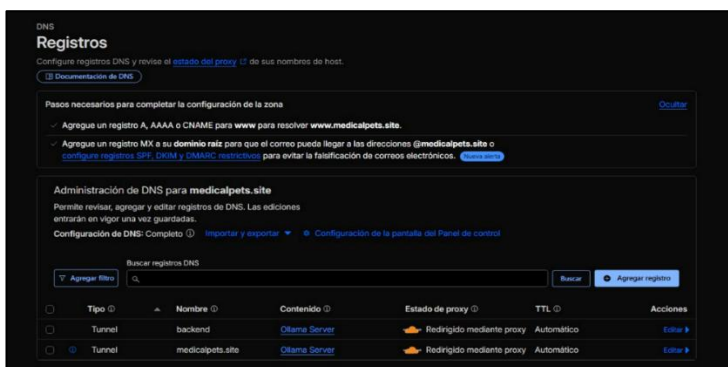


4.4.1.1. Configuración y registro de dominio

La primera fase del despliegue, se centró en la adquisición del dominio *web*, se adquirió por medio de *GoDaddy* escogiendo "*medicalpets.site*", para que los usuarios tengan acceso desde cualquier dispositivo que cuente con internet. El servidor en el que se alojó "*Medical Pets*" es *StarSoftware*, se realizaron configuraciones para asegurar que el servidor pueda salir por medio de un túnel donde se comunica con el dominio. Asimismo,

se asegura que los usuarios puedan reconocer de manera rápida la aplicación mientras navegan en la *web* y permitiendo realizar los procedimientos como los certificados *SSL* (figura 85).

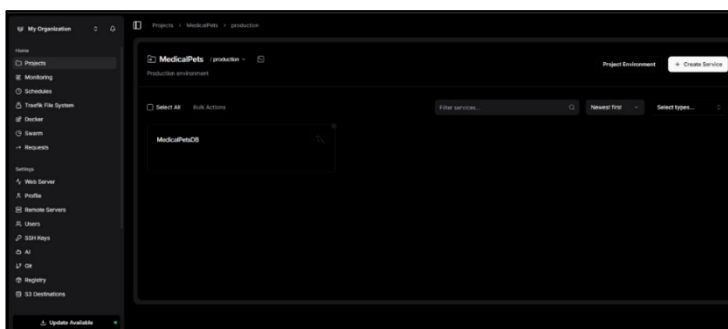
Figura 85. Configuración del DNS para túnel donde la aplicación se conecta con el dominio.



4.4.1.2. Configuración de base de datos en mysql

Para el almacenamiento estructurado y seguro de la información se seleccionó un sistema de base de datos relacional (*MySQL*). Este repositorio fue alojado en un contenedor separado mediante las herramientas de *Docker*. Durante la instalación y envío a producción se validaron las dependencias por medio de claves foráneas, normativas de integridad referencial y políticas de almacenamiento en disco a través de volúmenes de *Docker*.

Figura 86. Contenedor de la base de datos *MedicalPetsDB*.



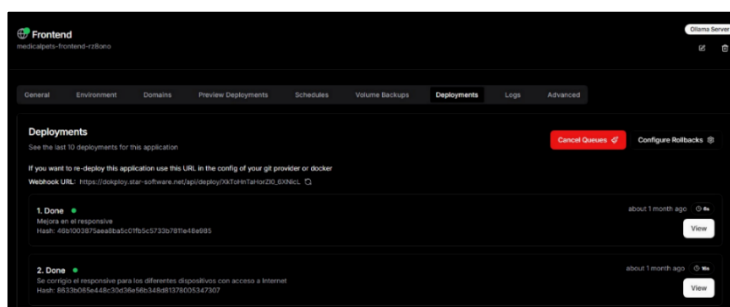
4.4.1.3. Despliegue del Backend y Frontend en el servidor

Para culminar el despliegue de la aplicación y que esté en producción se envió el *frontend* y *backend* al servidor, se utilizó el potencial de orquestación de *Docker Compose*

para administrar y empaquetar los distintos micro servicios usados, garantizando que el entorno de producción sea estable y replicable.

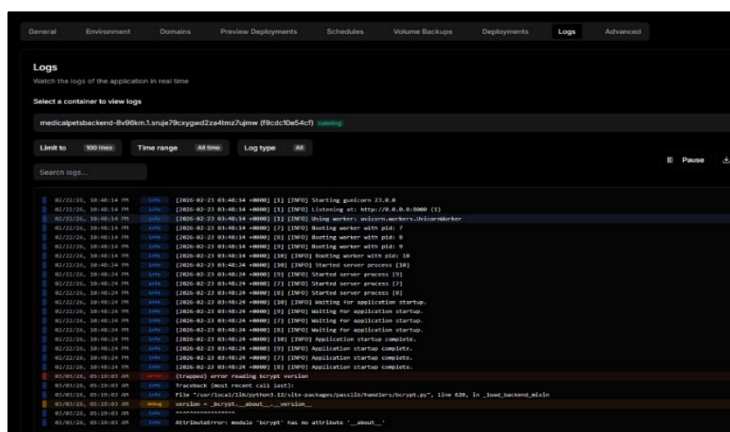
- **Frontend:** Es la parte visual de la aplicación que interactúa con el usuario, consta de varios recursos gráficos, por lo que para poder enviar todo esto al servidor se compilaron los recursos que fueron instalados en un contenedor enrutador (*Nginx*) que actúa como un servidor *web* estático (figura 87).

Figura 87. Despliegue *Frontend*



- **Backend y Machine Learning:** Construido completamente con *Python* y *FastAPI* posee toda la lógica de aplicación, junto con la implementación del sistema de posibles predicciones de enfermedades de las mascotas. De la misma manera para conseguir la materialización de este servicio se verificó la habilitación controlada de *endpoints*, mediante reglas *CORS* para recibir las peticiones enviadas por parte del *frontend* y la clasificación segura de variables externas. Al igual que el *frontend*, se utilizó un contenedor enrutador (*Nginx*) para facilitar la instalación de las dependencias que se utilizaron para la creación de la aplicación *web* (figura 88).

Figura 88. Despliegue Backend



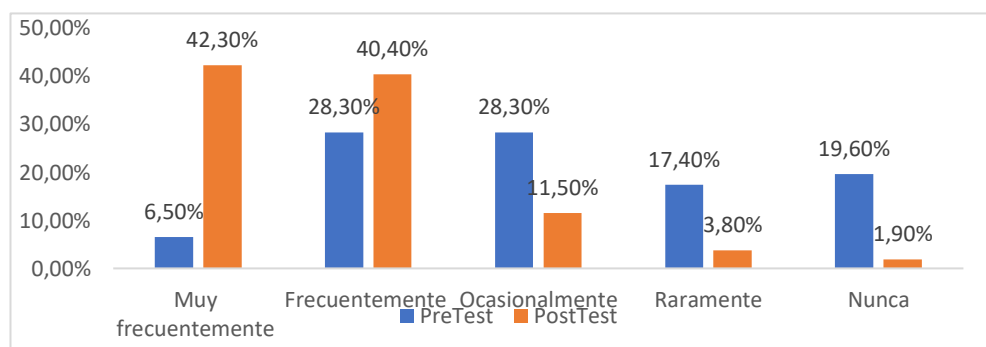
4.4.2. Validación de la propuesta

A través de la recolección de datos por medio de la encuesta a 50 clientes del hospital veterinario, se evidenció un alto nivel de satisfacción respecto al proceso de gestión de los historiales médicos de mascotas, las preguntas incluidas en la encuesta se encuentran descritas detalladamente en el Anexo 12.

Se llevó a cabo un análisis comparativo de los resultados obtenidos en las fases del *pretest* y *postest*, considerando 7 preguntas de un total de 20. Este procedimiento permitió evaluar el grado de fortalecimientos en la gestión de los historiales médicos tras implementar la aplicación *web*.

Pregunta 1: ¿Con qué frecuencia actualiza la información médica de su mascota?

Figura 89. Frecuencia de actualizaciones de historial médico



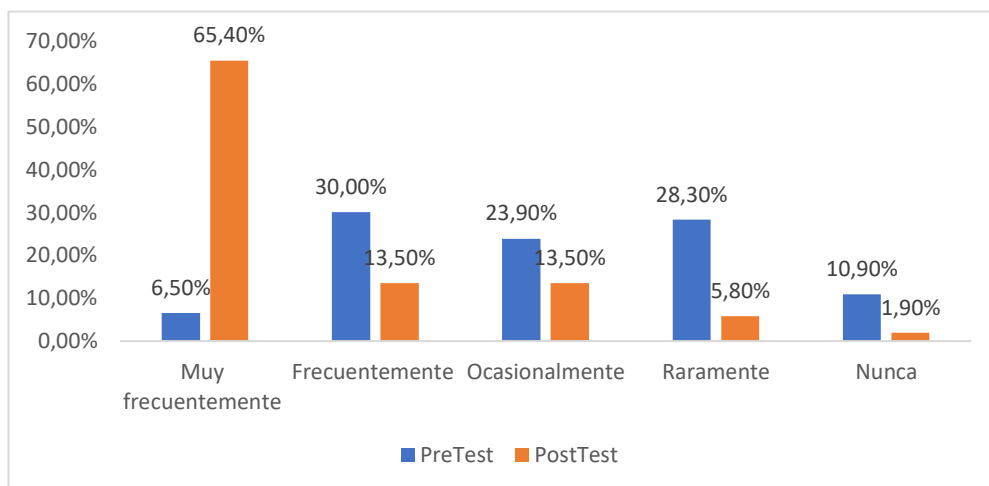
Análisis e interpretación: Los datos obtenidos por la información médica actualizada de las mascotas en la figura 89 del *pretest* y *postest*, demuestra un aumento en “muy frecuentemente” de 6.50% al 42.30%, mientras que “frecuentemente” pasa del 28.30% en el *pretest* al 40.40% del *postest*, y “ocasionalmente” disminuye del 28.30% al 11.50%, de igual manera “raramente” presenta una disminución del 17.40% al 3.80% y “nunca” del 19.60% al 1.90%.

Esto indica que existe una mejora en el historial médico, en el primer nivel se evidencia un aumento en “muy frecuentemente” del 35.80%, y del 12.10% en la categoría de “frecuentemente”, adicional un incremento del 16.8% en “ocasionalmente”. Los

resultados reflejan que la aplicación *web* permite actualizar los historiales médicos de manera más constante.

Pregunta 2: ¿Con qué frecuencia está informado/a sobre citas, vacunas o tratamientos pendientes de su mascota?

Figura 90. Frecuencia de información de citas médicas

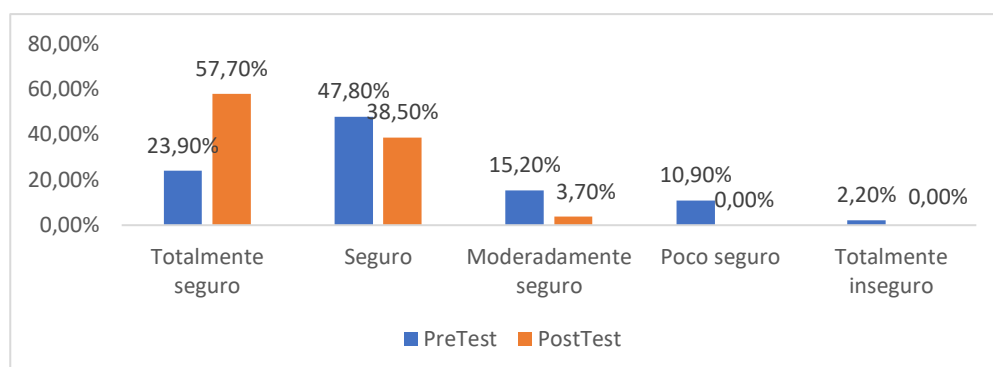


Análisis e interpretación: Los datos obtenidos de las encuestas del *pretest* y *posttest*, respecto a la frecuencia de la información que obtienen los dueños de las mascotas sobre citas o vacunas pendientes se encuentran en la figura 90. Se evidencia un aumento considerable en la categoría “muy frecuentemente” pasando del 6.5% al 65.4%. Además, un decremento en las opciones “ocasionalmente” con el 23.9% al 13.5%, “raramente” con el 28.3% al 5.8% y “nunca” con el 10.9% a 1.9%.

Las cifras obtenidas indican un aumento del 58.9% en la categoría “muy frecuentemente”, en la recurrencia con la que reciben información los clientes del hospital. Además, se indica un decremento del 10.4% en “ocasionalmente”, del 22.5% en “raramente” y 22.5% en “nunca”. Los resultados demuestran el fortalecimiento del seguimiento médico de las mascotas tras la implementación de la aplicación *web*.

Pregunta 3: ¿Considera seguro la administración del historial médico de su mascota?

Figura 91. Seguridad en los historiales médicos

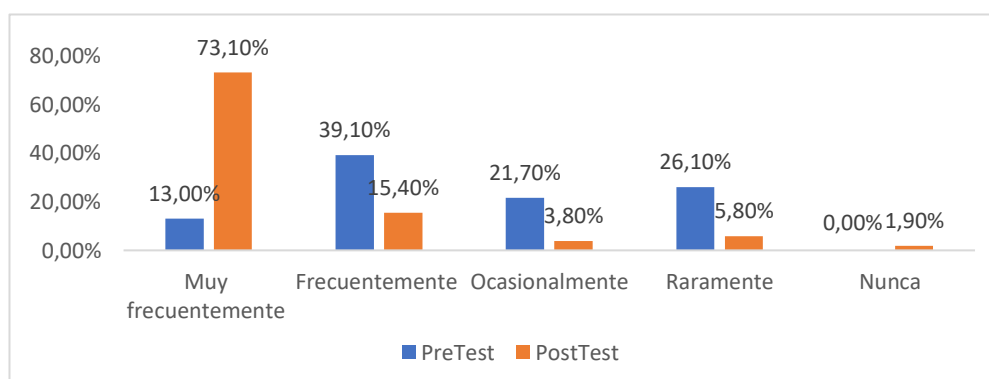


Análisis e interpretación: Con respecto a las respuestas en cuánto a la administración de los historiales médicos, se pudo observar un aumento en “totalmente seguro” del 23.90% del *pretest* al 57.70% en el *posttest*. Además, existe una disminución en la categoría de “seguro” del 47.80% al 38.50%, sin dejar a un lado el decremento del 15.20% al 3.70% en la opción de “moderadamente seguro” (figura 91).

Además, existe un aumento del 33.80% en “totalmente seguro” y una disminución del 9.30% en “seguro”, también se observa un decremento del 11.50% en “moderadamente seguro”. Esto indica el nivel de confianza de los clientes a la nueva herramienta implementada en el hospital veterinario, además las cifras demuestran una mejora positiva en cuanto a la seguridad de la información médica.

Pregunta 4: ¿Con qué frecuencia se siente involucrado/a activamente en el seguimiento de la salud de su mascota?

Figura 92. Seguimiento de la salud de la mascota

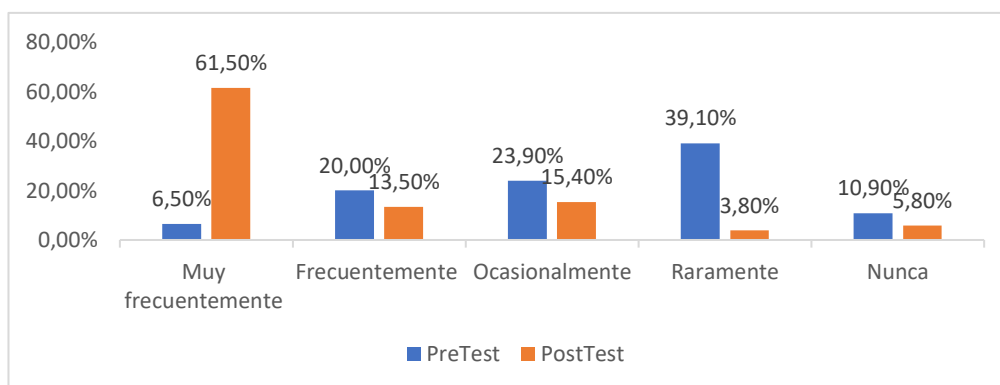


Análisis e interpretación: Los datos obtenidos respecto al seguimiento de la salud de las mascotas por parte del dueño se muestran en la figura 92. Durante el *pretest* “muy frecuentemente” cuenta con el 13% a diferencia del 73.10% que obtiene en el *postest*. De igual manera, durante el *pretest* “frecuentemente” cuenta con el 39.10% reduciendo al 15.40% en el final del *postest*, también se evidencia la disminución de 21.70% al 3.80% en “ocasionalmente”.

Los números evidencian el nivel de satisfacción de los clientes al ser partícipes activos del seguimiento de la salud de su mascota. Se observa el aumento del 60% en “muy frecuentemente”, la disminución del 23.70% en “frecuentemente” y del 17.9% en “ocasionalmente” después del uso de la aplicación *web*. Esto se deduce, en que mejora la percepción en cuanto al involucramiento de los dueños con la salud de su mascota.

Pregunta 5: ¿Con qué frecuencia accede a información médica de su mascota sin ayuda de personal médico?

Figura 93. Acceso a la información médica de las mascotas

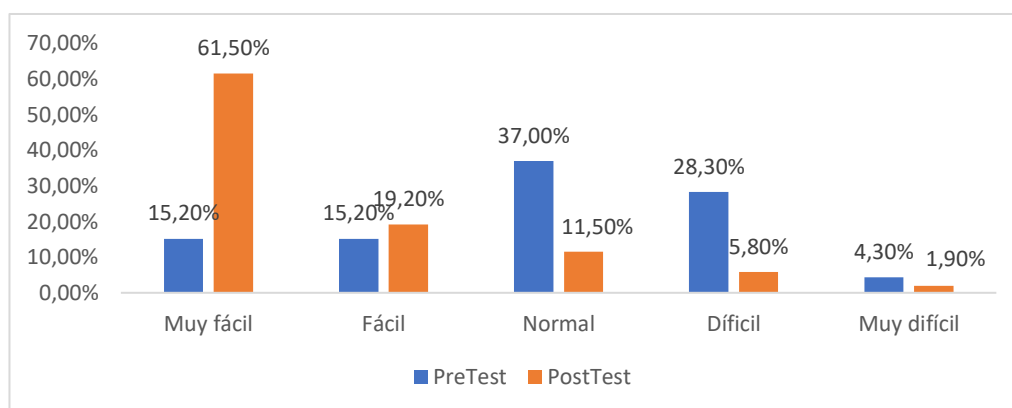


Análisis e interpretación: Los números obtenidos en el *pretest* y *postest* con respecto a la frecuencia con la que acceden a la información médica de la mascota sin necesidad del personal médico se encuentran en la figura 93. Durante el *pretest* “muy frecuentemente” presentaba 6.50% aumentando al 61.50% en la fase final del *postest*, mientras que “raramente” presenta una disminución del 39.10% al 3.80% y “ocasionalmente” disminuyendo del 23.90% al 15.40%.

Los resultados muestran el aumento del 55% en “muy frecuentemente”, por otro lado, se evidencia la disminución del 35.30% en “raramente” y del 8.50% en “ocasionalmente”. Esto se deduce a que, a los usuarios les resulta mucho más sencillo acceder al historial médico de su mascota sin ayuda del personal del hospital, después de la implementación de la aplicación *web*.

Pregunta 6: ¿Qué tan fácil le resulta registrar posibles reacciones a los medicamentos suministrados?

Figura 94. Registro de reacciones a medicamentos recetados

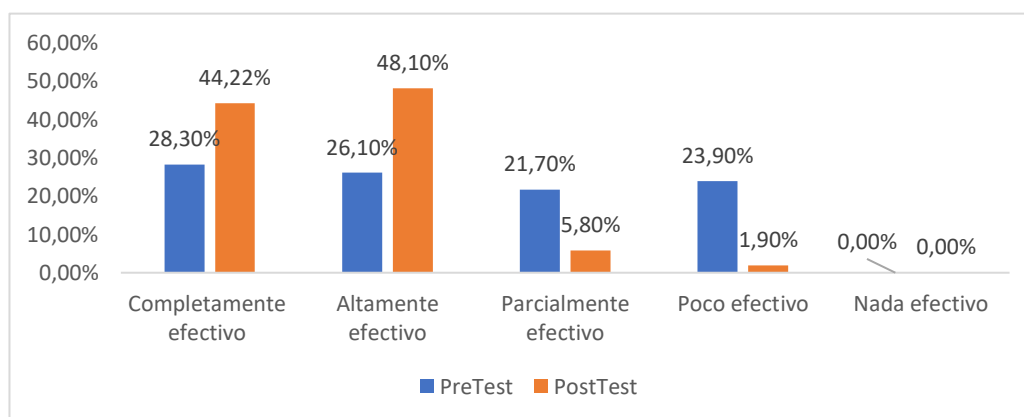


Análisis e interpretación: Los datos obtenidos en los resultados del *pretest* y *postest* respecto a la facilidad de registrar posibles reacciones a los medicamentos recetados se encuentran en la figura 94. Durante el *pretest* “muy fácil” contaba con el 15.20% mientras que en el *postest* aumenta al 61.50%. La categoría “normal” presenta una disminución del 37% al 11.50%, mientras que “dificil” disminuye del 28.30% al 5.80%.

Se evidencia el aumento del 46.30% en la categoría “muy fácil”, mientras que “normal” sufre un decremento del 25.5%, y “dificil” disminuye un 22.50%. Es así, que los resultados determinan que se reduce la dificultad para el registro de reacciones adversas mediante el uso de la aplicación *web*.

Pregunta 7: ¿Considera eficaz que una aplicación *web* tenga la capacidad de hacer posibles predicciones sobre la salud de su mascota basándose en el historial clínico de la misma?

Figura 95. Predicción en posibles predicciones sobre la salud de la mascota



Análisis e interpretación: Los resultados obtenidos acerca de la eficacia en la predicción de enfermedades de las mascotas se encuentran en la figura 95. Durante el *pretest* “altamente efectivo” con el 26.10% mientras que en el *posttest* sube al 48.10%. Además “parcialmente efectivo” con 21.70% disminuyendo al 5.80% en la fase final del *posttest* y “poco efectivo” decrece del 23.90% al 1.90%.

Los resultados demuestran el aumento del 22% en la categoría “altamente efectivo”, pero también el decremento del 15.90% en “parcialmente efectivo” y la disminución del 22% en la opción “poco efectivo”. Con las cifras recolectadas se evidencia un cambio positivo en la percepción de hacer posibles predicciones de enfermedades dentro de una aplicación *web*, además se visualiza la aceptación de los clientes con la nueva herramienta tecnológica.

4.4.3. Validación de la hipótesis

Una vez establecidos los parámetros correspondientes a los escenarios del *pretest* y *posttest*, se realiza la asignación de valores a las respuestas recopiladas de las encuestas. Los detalles de la codificación se presentan en la tabla 24.

Tabla 24. Recodificación de escenarios

Recodificación	Escenario
0	Sin Aplicación <i>Web</i> y sin <i>Machine Learning</i>
1	Con Aplicación <i>Web</i> y <i>Machine Learning</i>

Para el procesamiento de los datos, se adecuaron las puntuaciones correspondientes a la escala *Likert*, fueron usadas para medir el impacto de los resultados siguiendo el esquema detallado en el Anexo 11. Esta etapa de análisis se llevó a cabo con *IBM SPSS*, cuyo procedimiento se encuentra representado en las figuras 96 y 97.

Figura 96. Análisis con el SPSS (IBM Corporation, 2023)

The screenshot shows the 'Variables View' in IBM SPSS Statistics. It lists 36 variables, each with a name, a label, and a scale. The scales are categorized into 'Nominal', 'Ordinal', and 'Scale' (which includes 'Continuous' and 'Discrete').

Variable	Etiqueta	Escala
1	¿Establecer un historial de enfermedades?	Ordinal
2	¿Poco de acuerdo, Moderadamente de acuerdo?	Ordinal
3	¿Tratamiento de ac. importante?	Ordinal
4	¿Tratamiento de ac. Muy importante?	Ordinal
5	¿Basista de ac. importante?	Ordinal
6	¿Basista de ac. Muy importante?	Ordinal
7	¿Basista de ac. importante?	Ordinal
8	¿Basista de ac. Muy importante?	Ordinal
9	¿Basista de ac. importante?	Ordinal
10	¿Basista de ac. Muy importante?	Ordinal
11	¿Tratamiento de ac. importante?	Ordinal
12	¿Basista de ac. importante?	Ordinal
13	¿Basista de ac. importante?	Ordinal
14	¿Basista de ac. importante?	Ordinal
15	¿Tratamiento de ac. importante?	Ordinal
16	¿Basista de ac. importante?	Ordinal
17	¿Tratamiento de ac. importante?	Ordinal
18	¿Tratamiento de ac. importante?	Ordinal
19	¿Poco de acuerdo, Moderadamente de acuerdo?	Ordinal
20	¿Basista de ac. importante?	Ordinal
21	¿Tratamiento de ac. importante?	Ordinal
22	¿Poco de acuerdo, Moderadamente de acuerdo?	Ordinal
23	¿Tratamiento de ac. importante?	Ordinal
24	¿Basista de ac. importante?	Ordinal
25	¿Basista de ac. importante?	Ordinal
26	¿Basista de ac. importante?	Ordinal
27	¿Basista de ac. importante?	Ordinal
28	¿Basista de ac. importante?	Ordinal
29	¿Basista de ac. importante?	Ordinal
30	¿Tratamiento de ac. importante?	Ordinal
31	¿Basista de ac. importante?	Ordinal
32	¿Tratamiento de ac. importante?	Ordinal
33	¿Tratamiento de ac. importante?	Ordinal
34	¿Tratamiento de ac. importante?	Ordinal
35	¿Tratamiento de ac. importante?	Ordinal
36	¿Poco de acuerdo, Moderadamente de acuerdo?	Ordinal

Figura 97. Análisis con el SPSS (IBM Corporation, 2023)

The screenshot shows the 'Variables View' in IBM SPSS Statistics, displaying a list of 36 variables with their respective scales and measurement levels. The scales are categorized into 'Nominal', 'Ordinal', and 'Scale' (which includes 'Continuous' and 'Discrete').

Variable	Etiqueta	Escala
1	¿Establecer un historial de enfermedades?	Ordinal
2	¿Poco de acuerdo, Moderadamente de acuerdo?	Ordinal
3	¿Tratamiento de ac. importante?	Ordinal
4	¿Tratamiento de ac. Muy importante?	Ordinal
5	¿Basista de ac. importante?	Ordinal
6	¿Basista de ac. Muy importante?	Ordinal
7	¿Basista de ac. importante?	Ordinal
8	¿Basista de ac. Muy importante?	Ordinal
9	¿Basista de ac. importante?	Ordinal
10	¿Basista de ac. Muy importante?	Ordinal
11	¿Tratamiento de ac. importante?	Ordinal
12	¿Basista de ac. importante?	Ordinal
13	¿Basista de ac. importante?	Ordinal
14	¿Basista de ac. importante?	Ordinal
15	¿Tratamiento de ac. importante?	Ordinal
16	¿Basista de ac. importante?	Ordinal
17	¿Tratamiento de ac. importante?	Ordinal
18	¿Tratamiento de ac. importante?	Ordinal
19	¿Poco de acuerdo, Moderadamente de acuerdo?	Ordinal
20	¿Basista de ac. importante?	Ordinal
21	¿Tratamiento de ac. importante?	Ordinal
22	¿Poco de acuerdo, Moderadamente de acuerdo?	Ordinal
23	¿Tratamiento de ac. importante?	Ordinal
24	¿Basista de ac. importante?	Ordinal
25	¿Basista de ac. importante?	Ordinal
26	¿Basista de ac. importante?	Ordinal
27	¿Basista de ac. importante?	Ordinal
28	¿Basista de ac. importante?	Ordinal
29	¿Basista de ac. importante?	Ordinal
30	¿Tratamiento de ac. importante?	Ordinal
31	¿Basista de ac. importante?	Ordinal
32	¿Tratamiento de ac. importante?	Ordinal
33	¿Tratamiento de ac. importante?	Ordinal
34	¿Tratamiento de ac. importante?	Ordinal
35	¿Tratamiento de ac. importante?	Ordinal
36	¿Poco de acuerdo, Moderadamente de acuerdo?	Ordinal

A través de la implementación de la técnica de análisis de datos de regresión logística binaria, con un grado de libertad igual a uno ($gl=1$), se procesaron los datos recolectados del instrumento con el propósito de identificar factores específicos dentro de la gestión de historiales médicos de las mascotas. Se evaluaron indicadores como la actualización de los historiales médicos, la información sobre citas y tratamientos, la seguridad de la información, el involucramiento en la salud de la mascota, el acceso a la información de la mascota, facilidad en registro a reacciones médicas y las posibles

predicciones de la salud de las mascotas. Los resultados evidencian que las preguntas examinadas presentan una significancia estadística inferior a 0.05 ($p < 0.05$), lo que permite validar la hipótesis.

Tabla 25. Estudio cruzado en base a la aplicación

Recodificación	Puntuación	gl	Sig.
¿Con qué frecuencia actualiza la información médica de su mascota?	31.474	1	0.001
¿Con qué frecuencia está informado/a sobre citas, vacunas o tratamientos pendientes de su mascota?	32.710	1	0.020
¿Considera seguro la administración del historial médico de su mascota?	21.387	1	0.001
¿Con qué frecuencia se siente involucrado/a activamente en el seguimiento de la salud de su mascota?	26.226	1	0.001
¿Con qué frecuencia accede a información médica de su mascota sin ayuda de personal médico?	33.091	1	0.001
¿Qué tan fácil le resulta registrar posibles reacciones a los medicamentos suministrados?	29.197	1	0.001
¿Considera eficaz que una aplicación web tenga la capacidad de hacer posibles predicciones sobre la salud de su mascota basándose en el historial clínico de la misma?	18.196	1	0.001

En base a los resultados presentados en la tabla 25, se descarta la hipótesis nula (H_0), lo que evidencia que la implementación de la Aplicación *Web* con *Machine Learning* incide significativamente en el fortalecimiento del proceso de gestión de historiales médicos en el Hospital Veterinario Santa Martha, del cantón Santo Domingo.

5. DISCUSIÓN

Para identificar las necesidades de la clínica en cuanto al proceso de gestión de los historiales médicos de las mascotas, se realiza la reunión de planificación con el propietario. Durante el proceso mencionado, se pudo constatar que el propietario está de acuerdo a la implementación de la aplicación propuesta, ya que facilita procesos y garantiza mejor comunicación entre el dueño de la mascota y el veterinario. Por tal motivo, lo mencionado se adecua a lo planteado por Zaninelli et al. (2007), ya que menciona que el uso de aplicaciones tiene gran compatibilidad con centros veterinarios sin generar costos adicionales a los requeridos para el área médica de las mascotas.

Adicional, se aplica la encuesta dirigida a 50 clientes del hospital veterinario, los resultados reflejan que la implementación de la aplicación *web* aumenta el 30.45% en la frecuencia con la que se actualizan los historiales médicos, también demuestran un aumento de 65.40% en la comunicación de citas y tratamientos de las mascotas. Además, se evidencia que existe un incremento del 60% en el involucramiento de dueños con el seguimiento médico de su mascota, también se presencia el aumento del 55% en el acceso independiente a la información de la mascota. De la misma manera, se observa un incremento del 22% en la confianza en la efectividad de posibles predicciones de enfermedades de las mascotas. El criterio indicado por los clientes se sincroniza con lo propuesto por Hina et al (2024) que mencionan la importancia de contar con un sistema para el control de historias médicas para los dueños de las mascotas.

Con las mismas bases, el segundo objetivo busca determinar las herramientas tecnológicas enfocadas en el desarrollo de la aplicación *web*. Para la elección de los recursos digitales seleccionados se realizaron comparaciones entre varias tecnologías que cumplen con las necesidades requeridas para el proyecto. La aplicación fue desarrollada con *Python* como lenguaje de programación escogido, con la ayuda de dos *frameworks*: *Vue.js* para *frontend* y *FastAPI* para la parte del *backend* para optimizar el tiempo de desarrollo. Este criterio coincide con Molina et al. (2016), ya que indagan que

los *frameworks* son instrumentos importantes para desarrollar una aplicación, porque ayudan a acelerar el proceso de programación reutilizando código e incentivando el uso de buenas prácticas.

Una parte importante del proyecto es la elección de algoritmos de *machine learning* para la aplicación, se llega a la conclusión de utilizar árboles de decisión los cuales ayudan al procesamiento de los datos de manera gráfica, lo que optimiza la capacidad de aprendizaje mediante mecanismos supervisados en tareas complejas. Esto complementa lo mencionado por Patil et al (2023) sobre el papel importante del *machine learning* para la transformación y avance de la medicina en cualquier ámbito.

Es importante recalcar que, en el desarrollo de la aplicación *web* con *machine learning* para gestionar los historiales médicos de la veterinaria se utiliza *SCRUM* como metodología de desarrollo, es elegida por su enfoque colaborativo con el desarrollo de proyectos, lo que permite llevar un mejor control de tareas con *Jira*. Además, se define roles para el equipo, días de trabajo y reuniones para mostrar avances para detectar errores. En este contexto, se alinea con la expresión de Quitian et al. (2022), ya que indagan en que las metodologías ágiles permiten detectar fallos en el desarrollo de manera más precisa y rápida (pp. 6-8).

6. CONCLUSIONES Y RECOMENDACIONES

6.1. Conclusiones

El desarrollo del presente trabajo de titulación permite identificar las necesidades del hospital veterinario y sus clientes en cuanto a la administración de los historiales médicos de las mascotas. Además, con la participación de los clientes en las encuestas y la reunión de planificación con el propietario se evidencia la importancia de la implementación de la aplicación *web*. Con ello, se muestra que el hospital veterinario apoya el proceso a la digitalización de la información de la salud de las mascotas.

Por otra parte, el análisis comparativo entre opciones de herramientas tecnológicas ayuda a determinar las bases estables para el desarrollo de la aplicación *web*, asegurando que estas herramientas respondan a las necesidades identificadas anteriormente. La elección de *Vue.js* y *FastAPI* ayudan a construir interfaces amigables, pero con una arquitectura limpia e integración con servicios modernos.

El desarrollo de la aplicación es un proceso gestionado por fases, integrando módulos de registro, consulta y seguimiento de los datos, ayudando a fortalecer la toma de decisiones dentro del hospital. Además, las historias definidas desde el comienzo de la aplicación son artefactos que permiten cumplir con las funcionalidades requeridas. La arquitectura es diseñada siguiendo principios *REST*, lo que facilita una expansión en el futuro e incorporación a nuevas tecnologías conforme lo requiera el hospital.

En conclusión, la aplicación presentada demuestra ser una solución viable, moderna y segura para llevar el control de la información de las mascotas dentro del hospital. Los resultados indican que la digitalización de estos procesos, no solo mejora la organización interna del hospital, si no que optimiza el servicio brindado a los dueños de las mascotas. Además, el uso de *Machine Learning* abre oportunidades a la innovación de los análisis marcando una diferencia en el mercado.

6.2. Recomendaciones

Mediante el proceso de identificar los procesos de la gestión de historiales médicos de mascotas, se recomienda proporcionar una retroalimentación a los clientes que son usuarios de la aplicación *web* respecto a las funcionalidades de la nueva herramienta. Esto ayuda a que los dueños de las mascotas se familiaricen de la transición de procesos manuales a la digitalización de los datos de su mascota.

Además, las herramientas tecnológicas usadas ayudan a comprender las características de la aplicación, es por eso que se recomienda consultar de manera periódica la documentación oficial de la aplicación como: manuales de usuario y manual técnico, adjuntado en el apartado de anexos. Esto permite resolver cualquier duda relacionada a las actividades que se pueden llevar a cabo dentro de la aplicación *web* y garantizar que el sistema opere correctamente.

Finalmente, se recomienda seguir adoptando el uso de la tecnología en todos los procesos que contribuyan en la mejora del servicio del hospital. Además, es recomendable mantener el sistema actualizado y en revisión continua para adaptarlo a las herramientas modernas. Para la incorporación de futuras funcionalidades, se recomienda realizar encuestas a los usuarios y personal administrativo con la finalidad de identificar las dificultades y mejorar la experiencia del usuario.

7. REFERENCIAS

- Frómata Guerra, A., Sánchez Figueredo, S. A., Maya Castro, M. A., Jara Lalama, J., & Valarezo Sevilla, D. V. (2017). El método Clínico: Perspectivas actuales. *Revista Bionatura*, 2. doi:10.21931/RB/2017.02.01.8
- (AWS), A. W. (2024). *Amazon Web Services, Inc.* Obtenido de <https://aws.amazon.com/>
- Albán, J., & Fuentes, Y. (2018). *Desarrollo de Aplicación Web Para la Gestión de Historial Médico de Pacientes de la Clínica "San Miguel"*. Universidad Politécnica Salesiana Sede Guayaquil.
- Alexander Eduardo Castro Flores, J. P. (2020). *Desarrollo de un software para la gestión de los procesos del area de atención en la clínica veterinaria Colitas Felices de ICA.* Chincha, Ica. doi: Orcid N° 0000-0002-1946-8717
- Angular. (2024). *Angular*. Obtenido de Documentación oficial de Angular: <https://angular.dev/essentials>
- Arias Gonzáles, J. L., & Covinos Gallardo, M. (2021). *Diseño y metodología de la investigación*. (Vol. 1era Edición). Perú: Enfoques Consulting EIRL. doi:ISBN: 978-612-48444-2-3
- Armbrust, M, Fox, A, Griffith, R, Anthony D. Joseph, Randy Katz, Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, Matei Zaharia. (2010). A view of cloud computing. *The ACM Digital Library, Volume 53, Number 4*, 50-58.
- Atlassian, P. L. (2024). *Atlassian*. Obtenido de Atlassian.: <https://www.atlassian.com/>
- Áviles, S., Avila, D., & Avila, M. (2020). Desarrollo de sistema Web basado en los frameworks de Laravel y VueJs, para la gestión por procesos: Un estudio de caso. *Revista Peruana de Computación y Sistemas*, 3-10.
- Barrientos, R., Cruz, N., Acosta, H., Suárez, I., Gogeoascoechea, M., Pavón, P., & Blázquez, S. (2009). Árboles de decisión como herramienta en el diagnóstico.
- Bastar, S. G. (2012). *Metodología de la investigación*. Red Tercer Milenio - Primera Edición. doi:ISBN 978-607-733-149-0

- Becerra Correa, N., & Leguizamón Páez, M. (2024). *Regresión Logística Técnica de Machine Learning para predicciones académicas* (Vols. 12, Número Especial). Cundinamarca, Colombia: XIKUA Boletín Científico de la Escuela Superior de Tlahuelilpan. doi:<https://doi.org/10.29057/xikua.v12iEspecial.12746>
- Bernal Torres, C. (2016). *Metología de la investigación* (Pearson ed., Vol. 4to). Universidad de la Sabana, Colombia: Géminis Ltda. doi:ISBN: 978-958-699-309-8
- Blaz, F., Garcia, C., Maguiña, C., Tenorio, P., & Zevallos, M. (2023). *Plan de negocio para la creación de una startup que proporciona una herramienta digital en línea sobre el cuidado y seguimiento de mascotas, para el registro y uso de su historial clínico en centros veterinarios asociados en Lima Metropolitana.*
- Borrego del Pino, S. (2008). ISBN: 978-1-4562-6096-5. Ángel de Saavedra, Córdoba. doi:ISSN 1988-6047
- Castillo, J. D. (2016). *HTML5, CSS Y JAVASCRIPT. Crea tu web y apps con el estándar de desarrollo.* RC Libros .
- Cedeno Ochoa, A., Catuto Murillo, A., & Rodas-Silva. (2020). *El uso de aplicaciones Web para la Gestión de clínicas.* Ecuadorian Science Journal. Ecuador- Guayas: Licencia Creative Commons AtribuciónNoComercial-SinDerivar 4.0 Internacional. doi:<https://doi.org/10.46480/esj.5.4.174>
- Chaves, A. T., & Zuleta, A. M. (2014). Procesamiento del lenguaje natural, un reto de la inteligencia artificial. *Revista Matices Tecnológicos, 4.*
- Chávez, A., & Zuleta, A. (2012). Procesamiento del lenguaje natural, un reto de la inteligencia artificial. *Revista Matices Tecnológicos.*
- Chikkanayakanahalli , N. (2024). MVVM Design Pattern in Software Development. *International Journal of Computer Trends and Technology, 72*, 114-119. doi: <https://doi.org/10.14445/22312803/IJCTT-V72I9P117>
- Constituyente., A. (Registro Oficial No. 449, 20 de octubre de 2008). Constitución de la República del Ecuador. Ecuador - Monstecristi. Obtenido de <https://www.asambleanacional.gob.ec/>

- Corporation, I. B. (2023). *IBM*. Obtenido de <https://www.ibm.com/mx-es>
- Corporation, O. (2020). *Oracle Database*. Obtenido de <https://www.oracle.com/database/>
- Cruz Hernández, J., Hernández García, P., Abraham Marcel, E., Dueñas Gobel, N., & Salvato Dueñas, A. (2012). Importancia del método clínico. *Revista Cubana de Salud Pública*, 422-437. Obtenido de <http://scielo.sld.cu>
- Cruz, J., Hernández, P., & Dueñas, N. (2012). Importancia del Método Clínico. *Revista Cubana de Salud Pública*, 422-437.
- Díaz, J. (2021). Aprendizaje Automático y Aprendizaje Profundo. *Revista Chilena de ingeniería*, 182-183.
- Djar. (2024). *Documentación de Django*, 6.0. Obtenido de <https://docs.djangoproject.com/en/6.0/intro/overview/>
- Domingo., G. A. (2020). Ordenanza Municipal No. E. Obtenido de <https://www.epmtd.gob.ec/ordenanzas/views/modules/ordenanzas/Ordenanza>
- dominio, A. d. (2002). Prieto, Félix; Crespo, Yania; Marqués, José; Laguna, Miguel. Ecuador., A. N. (2018). *Código Orgánico del Ambiente*. Obtenido de <https://www.ambiente.gob.ec/>
- Ecuador., A. N. (2018). *Ley Orgánica de Bienestar Animal (LOBA)*. Obtenido de <https://www.asambleanacional.gob.ec/>
- Eíto, R., & Méndez, J. (2017). Normas técnicas para historia clínica electrónica en el proyecto Hcdsns. *Profesional De La información*, 1199-1210.
- Eíto-Brun, R., & Méndez-Solar, J. (2017). *Normas técnicas para historias clínicas en el proyecto HCDSNS*. Madrid - España.
- Escobar, N., Fong, J., Terazón, O., Alcaraz, M., & García, M. (2019). La relación médico-paciente; ¿necesidad o conveniencia? *Revista Cubana de Medicina*.
- Esperanza, M. R. (2020). Machine Learning: análisis de lenguajes de. *Revista Ibérica de Sistemas Y Tecnologías de la información*, 586-599.
- Esquivel, G. L. (2021). Paralelismos entre bases de datos relacionales y no relacionales (un enfoque en seguridad).

- Fajardo Ortiz , G., & Robledo, H. (2018). *Gestión de la atención médica, herramienta fundamental para los médicos residentes*. PubMed, Facultad de medicina, Ciudad de México. doi:10.21931/RB/2017.02.01.8
- FastAPI. (2024). *Documentación Oficial de FastApi*. Obtenido de Tiangolo. EE. UU.: <https://fastapi.tiangolo.com/features/>
- Fernández, Y., & Díaz, Y. (2012). Patrón Modelo-Vista-Controlador. *Revista Digital de las Tecnologías de la Información y las Comunicaciones*, 11, 47-57. doi:ISSN 1729-3804
- Fung, K., Xu, J., Brear, H., Lane, A., Lau, M., Wong, A., & D'Havé, A. (2024). Promoting interoperability between SNOMED CT and ICD-11: lessons learned from the pilot project mapping between SNOMED CT and the ICD-11 Foundation. *Journal of the American Medical Informatics Association*, 1631–1637.
- García, C., Augusto, C., Flores, C., Alexander , E., Ramírez, T., & Piero, J. (2021). *DESARROLLO DE UN SOFTWARE PARA LA GESTIÓN DE LOS PROCESOS DEL ÁREA DE ATENCIÓN EN LA CLINICA VETERINARIA COLITAS FELICES DE ICA*. Universidad Autonoma de ICA, Perú. doi:20.500.14441/1270
- Gómez Bastar, S. (2012). *Metodología de la investigacion*. México: Red Tercer Milenio S.C.
- Gourley, D., Totty, B., Sayer, M., Aggarwal, A., & Reddy, S. (2002). *HTTP The Definitive Guide*. O'Reilly Media, Inc.
- Gull, H., Abdulrahman, A., Abdullah, L., Saleh, R., Sami, S., Saeed, S., . . . Abdullah, M. (2024). Vet the Pet: Design of an Intelligent Health Prediction System and Health Care Application for Furry Friends in Saudi Arabia. *2024 2nd International Conference on Advancement in Computation & Computer Technologies (InCACCT)* (pág. 223). IEEE. doi:10.1109/InCACCT61598.2024.10551242
- Gunawan, R., & Rahmatulloh, A. (2019). JSON Web Token (JWT) untuk Authentication pada Interoperabilitas Arsitektur berbasis RESTful Web Service. *Jurnal Edukasian Penelitian Informatika*, 74-79.
- Guzmán, F., & Arias, C. (2012). La historia clínica: elemento fundamental del acto médico. *Revista Colombiana de Cirugía*, 15-24.

- Haz López, L. M. (2017). Implementación de un sistema en la nube para controlar y gestionar procesos clínicos. Caso veterinaria de mascotas. *3C Tecnología: glosas de innovación aplicadas a la pyme. Área de Innovación y Desarrollo, S.L., Edición 22 - Vol.6 – Nº 2*, 17-31. doi:<http://dx.doi.org/10.17993/3ctecno.2017.v6n2e22.17-31/>
- Hernández Sampieri, R., Fernández Collado, C., & Baptista Lucio, P. (2014). *Metodología de la investigación* (Vol. 1). México: Mc Graw Hill. doi:978-1-4562-2396-0
- Hernández, M., & Gómez, J. (2013). Aplicaciones de Procesamiento de Lenguaje Natural. *Revista Politécnica*, 87-96.
- Hernández-Rueda, K. V. (2022). Evaluación del rendimiento de una aplicación web: Evaluating the performance of a web application. *South Florida Journal of Development*, 3(1) - 445-457. Obtenido de <https://ojs.southfloridapublishing.com/ojs/index.php/jdev/article/view/1075>
- Hernández-Sampieri, R., & Mendoza-Torres, C. (2018). Enfoques de la Investigación. *Boletín Científico de las Ciencias Económicas Administrativas del ICEA*, 7 Núm.13. doi:<https://doi.org/10.29057/icea.v7i13.3519>
- Hina, G., Alharbi, A. A., Alahmari, L. A., Alghamdi, R. S., Nasser, S. S., & Alqahtani, S. S. (2024). Diseño de un sistema inteligente de predicción de salud y una aplicación para el cuidado de mascotas en Arabia Saudita. *Conferencia Internacional sobre Avances en Computación y Tecnologías Informáticas, 2da Conferencia*. doi:10.1109/InCACCT61598.2024.10551242
- Hinestroza Ramírez, D. (2018). *El Machine Learning a través de los tiempos, y los aportes a la humanidad*. doi:oai:repositorio.unilibre.edu.co:10901/17289
- Jose, S. C. (2025). *Niveles de Investigación* (Vol. 1era). Lima, Perú: BIOESTADISTICO. doi:978-612-99163-0-9
- Kober, J., Andrew, B., & Peters, J. (2013). *The International Journal of Robotics Research*. doi:10.1177/0278364913495721
- Lozada, J. (2014). *Investigación aplicada: Definición, Propiedad Intelectual e Industria*. CIENCIAMÉRICA.

- M. Iglesias-Urkiá, DC - Mansilla, S. Mayer y A. Urbieta. (2018). Validación de un CoAP según IEC 61850: Mapeo y Benchmarking vs HTTP-REST y WS-SOAP. 23.^a *Conferencia Internacional IEEE sobre Tecnologías Emergentes y Automatización de Fábricas*. Turín, Italia. doi:10.1109/ETFA.2018.8502624.
- Marinos. A, Gerard. B. (2009). *Cloud Computing*. Springer-Verlag Berlin Heidelberg. doi:LNCS 5931
- Matich, D. J. (2001). *Redes Neuronales: Conceptos básicos y aplicaciones*. Universidad Tecnológica Nacional, México.
- Mell. P; Grance. T. (2011). *The NIST Definition of Cloud Computing*. United States of America. doi:MD 20899-8930
- Memon, M., Thurasamy, R., Ting, H., & Cheah, J.-H. (2025). CONVENIENCE SAMPLING: A REVIEW AND GUIDELINES FOR QUANTITATIVE RESEARCH. doi:10.47263/JASEM.9(2)01
- Mendoza, S. L. H., & Monroy, T. I. S. (2018). Boletín Científico de las Ciencias económico Administrativas del ICEA. *Enfoques de la Investigación.*, 7(13), págs. 67-68. doi:<https://doi.org/10.29057/icea.v7i13.3519>
- Molina, J., Loja, N., Zea, M., & Loaiza, E. (2016). Evaluación de los Frameworks en el Desarrollo de Aplicaciones Web con Python. *Revista Latinoamericana de Ingeniería de Software* , 201-207.
- Nikita PATIL, A. Y. (2023). Application of Machine Learning in Pharmaceutical Industry. *Latin American Journal of Pharmacy*, 25-118. doi:10.7641/CTA.2019.80976
- Niño, J. (2010). *Aplicaciones Web*. Editex.
- Object Management Group (OMG). (2011). *Business Process Model and Notation (BPMN)*. doi:OMG Document Number: formal/2011-01-03
- Organización de Naciones Unidas. (1993). *Convenio sobre la Diversidad Biológica*. Obtenido de Organización de las Naciones Unidas.
- Organización Mundial de la Salud. (2024). *Organización Mundial de la Salud*. Obtenido de <https://www.who.int/es>

- Peña García, I., & Vidal Fernández, F. (2006). Historia Clínica Veterinaria Informatizada. *Revista Electrónica de Veterinaria, vol. VII, núm. 10*, 1-19. doi:E-ISSN: 1695-7504
- Pérez, S., Quispe, J., Mullicundo, F., & Lamas, D. (2021). *Herramientas y tecnologías para el desarrollo web desde el FrontEnd al BackEnd*.
- Perry, L. (2020). *IoT y Edge Computing para arquitectos: Implementación de sistemas de IoT y de borde desde sensores hasta nubes con sistemas de comunicación, análisis y seguridad* (Vol. 2nd Edición). Packt Publishing. doi:ISBN-13 : 978-1839218873
- PostgreSQL. (2026). *Documentación de PostgreSQL 18.3*. Obtenido de The PostgreSQL Global Development Group: <https://www.postgresql.org/docs/current/>
- Quishpe, M. V. (2023). Herramientas para el desarrollo de software. Una revisión. *Revista Científica de Informática ENCRYPTAR, 6 N°11*, 53-64. doi:<https://doi.org/10.56124/encryptar.v6i11.0004>
- Quispe Uznayo, M. M. (2019). Implementación de codificación diagnóstica basado en SNOMED-CT Swiss Medical Group Argentina. *Congreso Argentino de Informática y Salud (CAIS) - JAIIO 48*, (págs. 142-146). Argentina. Obtenido de <http://sedici.unlp.edu.ar/handle/10915/88163>
- Quispe Uznayo, M., Mammana, M., Monteverde, E., Gonzalez, F., Castro, D., & Novick, G. (2019). Implementación de codificación diagnóstica basado en SNOMED-CT Swiss Medical Group Argentina. *Congreso Argentino de Informática y Salud*. Argentina: JAIIO 48.
- Quitian Monroy, J. S., Martínez Padua, J. V., Castiblanco Jiménez, I. A. (2022). Caracterización y comparación de metodologías ágiles y tradicionales de desarrollo de producto. *Ciencia E Ingeniería Neogranadina, 2*, 9–26. Obtenido de <https://doi.org/10.18359/rcin.5168>
- RAMÍREZ, D. H. (2018). *EL MACHINE LEARNING A TRAVÉS DE LOS TIEMPOS, Y LOS APORTES A LA HUMANIDAD*. PROGRAMA DE INGENIERÍA DE SISTEMAS, UNIVERSIDAD LIBRE . Obtenido de <https://hdl.handle.net/10901/17289>

- Ramos Galarza, C. (2021). *Diseño de investigación experimental*. CienciAmérica . doi:ISSN 1390-9592
- Ríos, J. R. (2017). Estado del arte: Metodologías de desarrollo en aplicaciones web. 54-71. Obtenido de <http://dx.doi.org/10.17993/3ctecno.2017.v6n3e23.54-71/>
- Rivero, F. (2022). Árbol de Decision en Aprendizaje Automático . *Revista Varianza*, 39-46.
- Rojas, E. (2018). *Glosario de los seis términos básicos del Machine Learning*. Obtenido de <https://www.muycomputerpro.com/noticias>
- Rojas, E. M. (2020). Machine Learning: análisis de lenguajes de programación y herramientas para desarrollo. *Revista Ibérica de Sistemas y Tecnologías de Información*, 586-599. Obtenido de <https://www.proquest.com/openview/c7e24c997199215aa26a39107dd2fe98/1?pq-origsite=gscholar&cbl=1006393>
- Romero, Y. F., & González, Y. D. (2012). Patrón modelo-vista-controlador. *Revista Telem@tica*, 11(1), 47-57.
- Sánchez-Hernández, D., Lizano-Madriz, F., & Sandoval-Carvajal, M. M. (2020). ntegración de pruebas remotas de usabilidad en Programación Extrema: revisión de literatura. *Revista Uniciencia* , Vol. 34, 20-31. doi:<http://dx.doi.org/10.15359/ru.34-1.2>
- Sandoval Serrano, L. J. (2018). Algoritmos de aprendizaje automático para análisis y predicción de datos. *Revista Tecnológica*(N° 11). Obtenido de <http://hdl.handle.net/10972/3626>
- Schwaber, K., & Sutherland, J. (2020). *Le Guide Scrum*. Obtenido de <https://creativecommons.org/licenses/by-sa/4.0/deed.en>
- Serna M, E. (2017). *DESARROLLO E INNOVACIÓN EN INGENIERÍA* (Vol. Segunda Edición). Medellín – Antioquia: Editorial IAI. doi:ISBN: 978-958-59127-5-5
- Sommerville, I. (2011). *INGENIERÍA DE SOFTWARE Novena edición Ian Sommerville*. (L. M. Castillo, Ed.) México: PEARSON EDUCACIÓN. Recuperado el 12 de 05 de 2025, de

https://gc.scalahed.com/recursos/files/r161r/w25469w/ingdelsoftwarelibro9_compressed.pdf

Suárez, E. (2008). *¿Qué es una base de datos racional?*

Tejera-Martínez, F. A.-G. (2020). Lenguajes de programación y desarrollo de competencias clave. *Revista Electrónica de Investigación Educativa*, 22, e27, 1-16.

Tymkiw, N., Bournissen, J., & Tumino, M. (2020). SCRUM como Herramienta Metodológica para el Aprendizaje de la Programación. *Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología*, 26, 81-89. doi:10.24215/18509959.26.e9

Unidas., O. d. (1992). Convención sobre la Diversidad Biológica. Obtenido de <https://www.cbd.int/convention/text/>

Vue.js. (2024). *Vue.js*. Obtenido de Documentación oficial Vue.js: <https://vuejs.org/>

Yenisleidy Fernández Romero, Y. D. (2012). Patrón Modelo-Vista-Controlador. *Revista Digital de las tecnologías de la información y las comunicaciones*, 11- N°1, 47-57. Obtenido de <http://revistatelematica.cujae.edu.cu/index.php/tele>

Zaninelli, F.M., T., S., C., A., F., E., F., P.G., B., . . . C., S. (2007). El proyecto O3-Vet: Un registro electrónico veterinario del paciente basado en la tecnología web y el actor ADT-IHE para hospitales veterinarios. *ScienceDirect*, 9. doi:10.1016/j.jyme.2019.04.004

8. ANEXOS

Anexo 1: Carta de Asignación



**Pontificia Universidad
Católica del Ecuador**
Seréis mis testigos

**SANTO
DOMINGO**

DIRECCIÓN DE INVESTIGACIÓN
VINCULACIÓN E INNOVACIÓN

Santo Domingo, 17 de noviembre del 2025

Dr.
Patricio Intriago
Propietario de la Clínica Veterinaria Santa Martha
Presente. --

Reciba un atento saludo de quienes formamos parte de la Pontificia Universidad Católica del Ecuador, Sede Santo Domingo.

Por medio del presente informo a usted que el estudiante HERRERA CHANCAY JORDAN STALIN con C.C: 1722153887 y la estudiante PAREDES ROMERO JENNIFFER TAHIS con C.C: 1316463387 son estudiantes del OCTAVO nivel, asignatura INTEGRACIÓN CURRICULAR, Escuela de SISTEMAS. Actualmente se encuentran elaborando su Trabajo de Titulación denominado: **APLICACIÓN WEB CON MACHINE LEARNING PARA LA GESTIÓN DE HISTORIAL MEDICO DE LAS MASCOTAS EN EL HOSPITAL VETERINARIO SANTA MARTHA EN EL CANTÓN SANTO DOMINGO** PERÍODO 2025-2026; por tal motivo, solicito muy comedidamente se le permita desarrollar en su Institución las siguientes actividades:

ACTIVIDADES:

1. Solicitud de información referente al manejo de procesos de historiales médicos en la veterinaria.
2. Aplicación de las encuestas dirigidas a los clientes del negocio.
3. Aplicación de la entrevista a la gerente de la Clínica Veterinaria Santa Martha.
4. Entrega de aplicación y documentación del sistema desarrollado.

Los estudiantes deberán mantener la disciplina durante su estancia en la Institución, así como acogerse a las normativas necesarias para su desempeño.

Es necesario aclarar que los resultados de la investigación serán difundidos, mismos que previamente deberán ser comunicados a su persona a partir de lo cual le emitirá una carta que avale el impacto generado.

De antemano le agradezco su amable atención y ayuda en formar profesionales idóneos para servir a la sociedad.

Cordialmente,



MIKEL UGANDO PEÑATE

PhD. Mikel Ugando Peñate

RESPONSABLE DE INVESTIGACIÓN FORMATIVA PUCESD

cdivi@pucesd.edu.ec

(02)3702868 Ext. 204

Recibido 18/Noviembre/25
HUSH

Dirección: Vía a Chone Km. 2.
Código postal: 230203 / Teléfono: (593-993283425)
Santo Domingo, Ecuador / <https://pucesd.edu.ec/>

f t @ in y



Tres décadas
formando con
**fe, razón
y corazón**



Anexo 2: Tabla de recursos y cronograma

REGISTRO DE RECURSOS DEL PROYECTO				
GASTOS				
Recursos	Cantidad	Valor unitario (USD)	Valor total USD	
Humano				
Estudiantes	2	\$0,00	\$0,00	
Tecnología				
Computadora	2	\$1.500,00	\$3.000,00	
Software de desarrollo	1	\$0,00	\$0,00	
Materiales				
Impresiones	32	\$0,25	\$8,00	
Servicios				
Internet	1	\$20,00	\$20,00	
Alojamiento web mensual	5	\$10,00	\$50,00	
Base de datos	1	\$0,00	\$0,00	
Dominio	1	\$1,99	\$1,99	
Transporte				
Transporte equipo	16	\$0,39	\$6,24	
			TOTAL GASTOS	\$3.086,23
INGRESOS				
Recursos	Cantidad	Valor unitario (USD)	Valor total USD	
Recursos propios	2	\$150,00	\$300,00	
			TOTAL INGRESOS	\$300,00

PRODUCT BACKLOG DEL TRABAJO DE TITULACIÓN DE GRADO																		
N°	Actividades	Estado	2025															
			Producto Mínimo Viable I = 8° nivel						Producto Mínimo Viable II = 9° nivel									
			Marzo	Abril	Mayo	Junio	Julio	Agosto	Septiembre	Octubre	Noviembre	Diciembre	Enero	Febrero	Marzo			
EPICA 1	Diseño de la propuesta	HECHO																
EPICA 2	Ejecución del Proyecto	HECHO																
EPICA 3	Revisión literaria	HECHO																
EPICA 4	Metodología de la investigación	HECHO																
EPICA 5	Resultados	HECHO																
EPICA 5.1	Del primer objetivo específico	HECHO																
EPICA 5.2	Del segundo objetivo específico	HECHO																
EPICA 5.3	Del tercer objetivo específico	HECHO																
EPICA 6	Diseño de la propuesta de intervención (Creación adaptativa con incremento iterativo "Scrum")	HECHO																
EPICA 6.1	Prototipo de diseño operativo, tamaño solución en pasta	HECHO																
EPICA 6.2	Sprint 1	HECHO																
EPICA 6.3	Sprint 2	HECHO																
EPICA 6.4	Sprint 3	HECHO																
EPICA 7	Conclusiones y Recomendaciones	HECHO																
EPICA 8	Informe Final del Trabajo de Titulación de Grado	HECHO																
EPICA 8.1	Conclusiones de Lecturas	HECHO																
EPICA 8.2	Anexos	HECHO																
EPICA 9	Defensa de Grado	HECHO																

Anexo 3: Carta de impacto y consentimiento informado



SANTO DOMINGO

DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN

Santo Domingo, 11 de diciembre de 2025

PhD. Mikel Ugando Peñate
Responsable de investigación formativa PUCESD
 Pontificia Universidad Católica del Ecuador Sede Santo Domingo
 Presente. -


De mi consideración:

Reciba un cordial saludo y deseos de éxitos en sus delicadas funciones.

Por medio del presente, pongo en su conocimiento que el Trabajo de Titulación denominado **Aplicación Web con Machine Learning para la Gestión de Historial Médico de las mascotas en el Hospital Veterinario Santa Martha en el Cantón Santo Domingo**; elaborado por los estudiantes **Jordan Herrera y Jennifer Paredes**, estudiantes de Ingeniería en Tecnologías de la Información ha favorecido al desarrollo organizacional del Hospital generando un impacto en el cliente interno, externo y en la comunidad de los propietarios de las mascotas.

Por la atención dada a la presente, me suscribo de usted.

Atentamente,

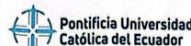


Dr. Patricio Intriago
 Propietario del Hospital Veterinario Santa Martha

Dirección: Vía a Chono Km. 2
 Código postal: 230203 / Teléfono: (593-993283425)
 Santo Domingo, Ecuador / <https://pucesd.edu.ec/>







SANTO DOMINGO

DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN

Consentimiento Informado


El Hospital Veterinario Santa Martha ubicado en Santo Domingo, libre y voluntariamente participa en el proyecto de Trabajo de Titulación de Grado de la Pontificia Universidad Católica del Ecuador Sede Santo Domingo, con el título **"Aplicación Web con Machine Learning para la Gestión de Historial Médico de las mascotas en el Hospital Veterinario Santa Martha en el Cantón Santo Domingo"**, elaborado por Jordan Herrera y Jennifer Paredes, estudiantes de la carrera de Ingeniería de Tecnologías de la Información.

Luego de firmar este documento certifico lo siguiente:

- Recibimos una copia de este documento de consentimiento informado.
- Estamos de acuerdo en que los datos recopilados, fotografías y resultados de este proyecto de Trabajo de Titulación de Grado se publiquen en artículos académicos, conferencias, en páginas web institucionales y en otros medios de comunicación.
- No esperamos recibir beneficios o pago por la participación.


Y a los efectos que procedan, firmamos el presente consentimiento informado.
 Santo Domingo, 11 de diciembre del 2025.

Firma del doctor propietario del Hospital Veterinario Santa Martha




Dr. Patricio Intriago

Firma de los autores del Trabajo de Titulación de Grado:




Sr. Jordan Stalin Herrera Chancay



Srta. Jennifer Tahis Paredes Romero

Dirección: Vía a Chono Km. 2
 Código postal: 230203 / Teléfono: (593-993283425)
 Santo Domingo, Ecuador / <https://pucesd.edu.ec/>





Anexo 4: Validación de instrumentos de recolección de datos

PUCE

Santo Domingo, 15 de junio del 2025

Estimado Mg. Jonathan Moreno,

De mi consideración:


El motivo del presente es que le hemos elegido a usted para redactar la solicitud de revisión y validación de los instrumentos de recolección de datos.


A continuación, encontrará la reunión de planificación y encuesta que contienen las preguntas que permitirán recolectar información de acuerdo al trabajo de titulación de grado "APLICACIÓN WEB CON MACHINE LEARNING PARA LA GESTIÓN DE HISTORIAL MÉDICO DE LAS MASCOTAS EN EL HOSPITAL VETERINARIO SANTA MARTHA EN EL CANTÓN SANTO DOMINGO", dirigida al doctor veterinario y los dueños de las mascotas del hospital veterinario Santa Martha.

Para la validación de los instrumentos se adjunta la operacionalización de variables, con la finalidad de que se visualice la relación de las preguntas con las categorías e indicadores. Además, se encuentran divididos los instrumentos en dos partes, la primera corresponde a la reunión de planificación (preguntas de fondo verde para las dos variables) y la segunda a la encuesta (preguntas de la variable independiente de fondo turquesa y las preguntas de la variable dependiente fondo gris).

Gracias por su valiosa colaboración en este trabajo de titulación de grado.

Atentamente


Jordan Herrera
jsherrera@pucesd.edu.ec


Jennifer Paredes
jtparedes@pucesd.edu.ec

Operacionalización de la variable independiente aplicación web

Conceptualización	Dimensión	Indicadores	Preguntas	Herramientas
Hernández-Rueda et al. (2022), afirma que una aplicación web se define como un software el que se accede por medio de un navegador.	Herramientas de Desarrollo	Lenguaje de programación	¿Considera que el manejo de aplicaciones web es una alternativa más fácil al acceder a la información de una mascota?	Reunión de planificación con veterinario
		Frameworks de desarrollo	¿Cree necesario que una aplicación web cuente con un diseño entendible? ¿Estás de acuerdo en visualizar los datos médicos de su mascota desde un dispositivo sin acceso a internet? ¿Qué dispositivo electrónico utiliza regularmente?	Encuesta a los clientes
Base de datos	Base de datos	Base de datos relacional (MySQL)	¿Qué tan importante es para usted digitalizar el historial clínico de su mascota?	Encuesta a clientes
		NoSQL	¿Cómo se respalda la información de los historiales médicos en el hospital veterinario?	Reunión de planificación con veterinario
		Cloud Computing	¿Con qué frecuencia tiene acceso a internet?	Encuesta a clientes
Tecnologías de acceso	Tecnologías de acceso	HTTP/REST	¿Con qué frecuencia utiliza navegadores web, como Google Chrome, Safari o Firefox?	Encuesta a clientes
		WLAN / WAN	¿Está familiarizado con el uso de tecnologías de computación en la nube? ¿Cuál es el procedimiento para actualización de citas en el hospital?	Reunión de planificación con veterinario
Metodología de desarrollo	Metodología de desarrollo	Método ágil (SCRUM)	¿Qué tan importante es para usted que una aplicación web le permita obtener información de la salud de su mascota de manera organizada y segura?	Reunión de planificación con veterinario

PUCE

Aplicaciones de Machine Learning

Lenguaje Natural (NLP)

¿Qué nivel de confianza le genera el uso de inteligencia artificial para la detección de enfermedades?

Tabla 2. Variable independiente machine learning

Operacionalización de la variable dependiente historial médico

Conceptualización	Dimensión	Indicadores	Preguntas	Herramientas
Según ACLAM Medical Records Committee et al (2007) las historias clínicas veterinarias son consideradas un elemento clave en un programa de atención veterinaria adecuada para animales utilizados en investigación, encuestas y pruebas.	Gestión de historial médico	Historial médico digital	¿El hospital cuenta con un sistema de gestión para historiales médicos de las mascotas? ¿Qué tan importante es para usted acceder con facilidad al historial médico de sus mascotas?	Reunión de planificación con veterinario
		Historial médico físico	¿Considera seguro la administración del historial médico de su mascota? ¿Con qué frecuencia accede a información médica de su mascota sin ayuda de personal médico? ¿Con qué frecuencia ha experimentado problemas para acceder al historial médico de su mascota?	Encuesta a clientes
		Frecuencia de ingreso de datos	¿Con qué frecuencia actualiza la información médica de su mascota? ¿Cómo se respalda la información de los historiales médicos de la mascota respecto al servicio hospital?	Reunión de planificación con veterinario
Actualización de historiales médicos	Actualización de historiales médicos	Frecuencia de ingreso de datos	¿Con qué frecuencia se actualizan los datos de los pacientes en el veterinario? ¿Con qué frecuencia se siente involucrado/a activamente en el seguimiento de la salud de su mascota?	Reunión de planificación con veterinario
		Encuestas a clientes		Encuesta a clientes

PUCE

permitiendo el acceso de manera remota a servicios o datos de los usuarios.

Extreme Programming (XP)

¿Con qué frecuencia está informado/a sobre citas, vacunas o tratamientos pendientes de su mascota?
¿Considera importante realizar el registro de vacunas de pacientes?

JWT - OAuth2

¿Considera importante que los usuarios tengan roles definidos?

Control de acceso por roles

¿Con qué frecuencia actualiza la información médica de su mascota?

Tabla 1. Variable independiente aplicación web

Operacionalización de la variable independiente machine learning

Conceptualización	Dimensión	Indicadores	Preguntas	Herramientas
El aprendizaje automático, o también conocido como Machine Learning (ML), la podemos denominar como la ciencia que permite a los humanos diseñar algoritmos y enseñar a las computadoras aprender patrones de grandes cantidades de datos, para poder usar estos patrones para la toma de decisiones o predicciones automáticas (Pan, F. 2023).	Tipos de aprendizaje	Aprendizaje supervisado	¿Considera eficaz que una aplicación web tenga la capacidad de hacer posibles predicciones sobre la salud de su mascota basándose en el historial clínico de la misma?	Encuesta a clientes
		Aprendizaje no supervisado	¿Considera que hacer una relación entre síntomas y enfermedades puede ayudar a los diagnósticos tempranos de la mascota?	Reunión de planificación con veterinario
Algoritmos de machine learning	Algoritmos de machine learning	Regresión Logística	¿Cuál es su opinión respecto al uso de robots quirúrgicos en el diagnóstico temprano de la mascota?	Reunión de planificación con veterinario
		Redes Neuronales	¿El hospital cuenta con un proceso para predicción de enfermedades?	Reunión de planificación con veterinario
		Árbol de decisiones	¿Le gustaría poder predecir comportamientos o detectar anomalías en la salud de una mascota?	Reunión de planificación
Predicciones				

PUCE

Presuntivo

¿De qué forma administra los tratamientos de su mascota?

Definitivo

¿Con qué frecuencia le resulta sencillo entender los diagnósticos y tratamientos que recibe su mascota?

Diagnóstico Médico

¿Considera importante llevar un control digital de los tratamientos de su mascota?
¿Qué tan fácil le resulta registrar posibles reacciones a los medicamentos suministrados?

Tratamiento

¿Considera necesario registrar información de posibles reacciones de los medicamentos suministrados a las mascotas?
¿Le gustaría conocer el proceso o reacciones de medicamentos de los tratamientos asignados a las mascotas?

Componentes de historial médico

Clinico

¿Le parece importante poder visualizar el historial de tratamientos que llevan las mascotas?

Laboratorio

Clasificación de diagnóstico

CIE-10

¿Le gustaría que un sistema cuente con la clasificación de diagnósticos?

SNOMED CT

Tabla 3. Variable dependiente historial médico

PUCE

Preguntas

Baremo:
Claridad: Se refiere si la pregunta está comprendida por los destinatarios.
Pertinencia: Se refiere si la pregunta corresponde con lo que se quiere indagar
 Las preguntas en cuanto a su claridad y pertinencia tienen una escala valorativa Likert del 1 al 5 (donde 1 es el menor valor y 5 el mayor). Podrá añadir una formulación alternativa y observación, en caso que considere necesario.
 Marque con una cruz (X) el tramo del baremo que exprese mejor su juicio "Claridad" y "Pertinencia" sobre los ítems propuestos:

Reunión de planificación con el gerente propietario del hospital veterinario Santa Martha

Variable dependiente: Gestión de historiales médicos (Categoría: Historiales médicos)

1. ¿Cuál es el procedimiento para gestión de historiales médicos de las mascotas?

CLARIDAD					PERTINENCIA					FORMULACIÓN ALTERNATIVA	OBSERVACIÓN
1	2	3	4	5	1	2	3	4	5		

Variable dependiente: Gestión de historiales médicos (Categoría: Historiales médicos)

2. ¿Considera que digitalizar los historiales médicos de sus mascotas mejora el servicio del hospital?

CLARIDAD					PERTINENCIA					FORMULACIÓN ALTERNATIVA	OBSERVACIÓN
1	2	3	4	5	1	2	3	4	5		

Variable dependiente: Gestión de pacientes (Categoría: Pacientes)

3. ¿Con qué frecuencia se actualizan los datos de los pacientes en la veterinaria?

CLARIDAD					PERTINENCIA					FORMULACIÓN ALTERNATIVA	OBSERVACIÓN
1	2	3	4	5	1	2	3	4	5		

Variable dependiente: Gestión de tratamientos (Categoría: Tratamientos)

4. En el proceso de asignación de tratamientos ¿Cuál es el método para el control de los mismos?

CLARIDAD					PERTINENCIA					FORMULACIÓN ALTERNATIVA	OBSERVACIÓN
1	2	3	4	5	1	2	3	4	5		

6

PUCE

Variable dependiente: Gestión de tratamientos (Categoría: Tratamientos)

5. ¿Le gustaría tener acceso a posibles reacciones de medicamentos de los tratamientos asignados a las mascotas?

CLARIDAD					PERTINENCIA					FORMULACIÓN ALTERNATIVA	OBSERVACIÓN
1	2	3	4	5	1	2	3	4	5		

Variable independiente: Predicción de enfermedades (Categoría: Análisis de datos con IA)

6. ¿Considera que hacer una relación entre síntomas y enfermedades podría ayudar a posibles diagnósticos tempranos de la mascota?

CLARIDAD					PERTINENCIA					FORMULACIÓN ALTERNATIVA	OBSERVACIÓN
1	2	3	4	5	1	2	3	4	5		

Variable independiente: Predicción de enfermedades (Categoría: Análisis de datos con IA)

7. ¿Cuál es su opinión respecto a la automatización en el diagnóstico temprano de la mascota?

CLARIDAD					PERTINENCIA					FORMULACIÓN ALTERNATIVA	OBSERVACIÓN
1	2	3	4	5	1	2	3	4	5		

Variable independiente: Predicción de enfermedades (Categoría: Análisis de datos con IA)

8. ¿El hospital cuenta con un plan para una posible predicción de enfermedades?

CLARIDAD					PERTINENCIA					FORMULACIÓN ALTERNATIVA	OBSERVACIÓN
1	2	3	4	5	1	2	3	4	5		

Variable independiente: Aplicación web (Categoría: Información de la mascota)

9. ¿Considera que el manejo de aplicaciones web es una alternativa más fácil de acceder a información de una mascota?

CLARIDAD					PERTINENCIA					FORMULACIÓN ALTERNATIVA	OBSERVACIÓN
1	2	3	4	5	1	2	3	4	5		

Variable independiente: Aplicación web (Categoría: Información de la mascota)

10. ¿Cómo se respaldaría la información de los historiales médicos en el hospital veterinario?

7

PUCE

CLARIDAD					PERTINENCIA					FORMULACIÓN ALTERNATIVA	OBSERVACIÓN
1	2	3	4	5	1	2	3	4	5		

Variable independiente: Aplicación web (Categoría: Tecnología)

11. ¿Está familiarizado con el uso de tecnologías de computación en la web?

CLARIDAD					PERTINENCIA					FORMULACIÓN ALTERNATIVA	OBSERVACIÓN
1	2	3	4	5	1	2	3	4	5		

Variable independiente: Aplicación web (Categoría: Agendación de citas)

12. ¿Cuál es el procedimiento para agendación de citas en el hospital veterinario?

CLARIDAD					PERTINENCIA					FORMULACIÓN ALTERNATIVA	OBSERVACIÓN
1	2	3	4	5	1	2	3	4	5		

Variable independiente: Aplicación web (Categoría: Agendación de citas)

13. ¿Considera importante agilizar el proceso de registro de pacientes?

CLARIDAD					PERTINENCIA					FORMULACIÓN ALTERNATIVA	OBSERVACIÓN
1	2	3	4	5	1	2	3	4	5		

8

PUCE

Encuesta a dueños de mascotas

Tema del Trabajo de Titulación de grado: Aplicación web con machine learning para la gestión de historial médico de las mascotas en el hospital veterinario Santa Martha en el cantón santo domingo.

Objetivo: Recolectar información para validar propuesta de aplicación web para la gestión de historial médico en las mascotas.

Instrucciones al público objetivo: La encuesta está dirigida a los dueños de las mascotas del hospital veterinario Santa Martha.

PREGUNTAS

Variable independiente: Aplicación web (Categoría: Interfaz de usuario)

14. ¿Cree necesario que una aplicación web cuente con un diseño entendible? (PRE)

a) Muy necesario
 b) Necesario
 c) Medianamente necesario
 d) Poco necesario
 e) Innecesario

Relevancia	Claridad	Formulación alternativa:	Observación:								
1	2	3	4	5	1	2	3	4	5		

Variable independiente: Aplicación web (Categoría: Información de la mascota)

15. ¿Estás de acuerdo en visualizar los datos médicos de su mascota desde un dispositivo con acceso a internet? (PRE Y POST)

a) Totalmente de acuerdo
 b) Bastante de acuerdo
 c) Neutral
 d) Poco de acuerdo
 e) Totalmente en desacuerdo

Relevancia	Claridad	Formulación alternativa:	Observación:								
1	2	3	4	5	1	2	3	4	5		

Variable independiente: Aplicación web (Categoría: Tecnología)

16. ¿Con qué frecuencia tiene acceso a internet? (PRE)

9

PUCE

a) Muy frecuente
b) Frecuentemente
c) Ocasionalmente
d) Raramente
e) Nunca

Relevancia	Claridad	Formulación alternativa:	Observación:
1 2 3 4 5	1 2 3 4 5		

Variable independiente: Aplicación web (Categoría: Información de la mascota)

4. ¿Qué dispositivo electrónico utiliza regularmente? (PRE)

a) Teléfono móvil
b) Laptop
c) Tablet
d) Computadora de escritorio

Variable independiente: Aplicación web (Categoría: Tecnologías)

5. ¿Con qué frecuencia utiliza navegadores web, como Google Chrome, Safari o Firefox? (PRE)

a) Muy frecuentemente
b) Frecuentemente
c) Ocasionalmente
d) Raramente
e) Nunca

Relevancia	Claridad	Formulación alternativa:	Observación:
1 2 3 4 5	1 2 3 4 5		

Variable dependiente: Aplicación web (Categoría: Tecnologías)

6. ¿Qué tan importante es para usted digitalizar el historial clínico de su mascota? (PRE Y POST)

a) Muy importante
b) Importante
c) Moderadamente importante
d) Poco importante
e) No es importante

Relevancia	Claridad	Formulación alternativa:	Observación:
1 2 3 4 5	1 2 3 4 5		

Variable dependiente: Aplicación web (Categoría: Tecnologías)

10

PUCE

7. ¿Qué tan importante es para usted que una aplicación web le permita obtener información de la salud de su mascota de manera organizada y segura? (PRE Y POST)

f) Muy importante
g) Importante
h) Moderadamente importante
i) Poco importante
j) No es importante

Relevancia	Claridad	Formulación alternativa:	Observación:
1 2 3 4 5	1 2 3 4 5		

Variable dependiente: Aplicación web (Categoría: Tecnologías)

8. ¿Con qué frecuencia actualiza la información médica de su mascota? (PRE Y POST)

a) Muy frecuentemente
b) Frecuentemente
c) Ocasionalmente
d) Raramente
e) Nunca

Relevancia	Claridad	Formulación alternativa:	Observación:
1 2 3 4 5	1 2 3 4 5		

Variable independiente: Aplicación web (Categoría: Citas médicas)

9. ¿Con qué frecuencia está informado/a sobre citas, vacunas o tratamientos pendientes de su mascota? (PRE Y POST)

a) Muy frecuentemente
b) Frecuentemente
c) Ocasionalmente
d) Raramente
e) Nunca

Relevancia	Claridad	Formulación alternativa:	Observación:
1 2 3 4 5	1 2 3 4 5		

Variable independiente: Predicción de enfermedades (Categoría: Análisis de datos con IA)

11

PUCE

10. ¿Considera eficaz que una aplicación web tenga la capacidad de hacer posibles predicciones sobre la salud de su mascota basándose en el historial clínico de la misma? (PRE Y POST)

f) Completamente efectivo
g) Altamente efectivo
h) Parcialmente efectivo
i) Poco efectivo
j) Nada efectivo

Relevancia	Claridad	Formulación alternativa:	Observación:
1 2 3 4 5	1 2 3 4 5		

Variable dependiente: Gestión de historial médico (Categoría: Historial médico)

11. ¿Qué tan importante es para usted acceder con facilidad al historial médico de sus mascotas? (PRE)

a) Muy importante
b) Importante
c) Moderadamente importante
d) Poco importante
e) No es importante

Relevancia	Claridad	Formulación alternativa:	Observación:
1 2 3 4 5	1 2 3 4 5		

Variable dependiente: Gestión de historial médico (Categoría: Historial médico)

12. ¿Considera seguro la administración del historial médico de su mascota? (PRE Y POST)

a) Totalmente seguro
b) Seguro
c) Moderadamente seguro
d) Poco seguro
e) Totalmente inseguro

Relevancia	Claridad	Formulación alternativa:	Observación:
1 2 3 4 5	1 2 3 4 5		

Variable dependiente: Gestión de historial médico (Categoría: Historial médico)

13. ¿Con qué frecuencia accede a información médica de su mascota sin ayuda de

12

PUCE

personal médico? (PRE Y POST)

a) Siempre
b) Casi siempre
c) A veces
d) Casi nunca
e) Nunca

Relevancia	Claridad	Formulación alternativa:	Observación:
1 2 3 4 5	1 2 3 4 5		

Variable dependiente: Gestión de historial médico (Categoría: Historial médico)

14. ¿Con qué frecuencia ha experimentado problemas para acceder al historial médico de su mascota? (PRE Y POST)

a) Siempre
b) Casi siempre
c) A veces
d) Casi nunca
e) Nunca

Relevancia	Claridad	Formulación alternativa:	Observación:
1 2 3 4 5	1 2 3 4 5		

Variable dependiente: Gestión de tratamientos (Categoría: Tratamientos)

15. ¿Con qué frecuencia le resulta sencillo entender los diagnósticos y tratamientos que recibe su mascota? (PRE Y POST)

a) Siempre
b) Casi siempre
c) A veces
d) Casi nunca
e) Nunca

Relevancia	Claridad	Formulación alternativa:	Observación:
1 2 3 4 5	1 2 3 4 5		

Variable dependiente: Gestión de tratamientos (Categoría: Tratamientos)

13

PUCE

16. ¿De qué forma administra los tratamientos de su mascota? (PRE)

a) Manual
b) Digital
c) No administro

Relevancia	Claridad	Formulación alternativa:	Observación:
1 2 3 4 5	1 2 3 4 5		

Variable dependiente: Gestión de tratamientos (Categoría: Tratamientos)

17. ¿Con qué frecuencia se siente involucrado/a activamente en el seguimiento de la salud de su mascota? (PRE Y POST)

a) Siempre
b) Casi siempre
c) A veces
d) Casi nunca
e) Nunca

Relevancia	Claridad	Formulación alternativa:	Observación:
1 2 3 4 5	1 2 3 4 5		

Variable dependiente: Gestión de tratamientos (Categoría: Tratamientos)

18. ¿Considera importante llevar un control digital de los tratamientos de su mascota? (PRE)

a) Muy importante
b) Importante
c) Moderadamente importante
d) Poco importante
e) No es importante

Relevancia	Claridad	Formulación alternativa:	Observación:
1 2 3 4 5	1 2 3 4 5		

Variable dependiente: Gestión de tratamientos (Categoría: Tratamientos)

19. ¿Qué tan fácil le resulta registrar posibles reacciones a los medicamentos suministrados? (PRE Y POST)

a) Muy fácil
b) Fácil

14

PUCE

c) Normal
d) Difícil
e) Muy difícil

Relevancia	Claridad	Formulación alternativa:	Observación:
1 2 3 4 5	1 2 3 4 5		

Variable dependiente: Gestión de tratamientos (Categoría: Tratamientos)

20. ¿Considera necesario registrar información de posibles reacciones de los medicamentos suministrados a las mascotas? (PRE)

a) Muy necesario
b) Necesario
c) Medianamente necesario
d) Poco necesario
e) Innecesario

Relevancia	Claridad	Formulación alternativa:	Observación:
1 2 3 4 5	1 2 3 4 5		

Datos generales

Edad:
a) 12-14
b) 14-16
c) 16-18
d) 18 o más

Género:
a) Femenino
b) Masculino
c) Prefiero no decirlo

¿Posee alguna discapacidad?
a) Sí
b) No

En caso de ser afirmativa la pregunta anterior, ¿Qué tipo de discapacidad posee?
a) Visual
b) Auditiva
c) Física
d) Intelectual
e) Otra (indique): _____

15

3/7/25, 21:20 Como: PAREDES ROMERO JENNIFER TAHIS - Outlook

PUCE

¡GRACIAS POR SU COLABORACIÓN!


Una vez finalizada su validación, puede realizar comentarios, sugerencias o la aprobación, además, es pertinente que agregues sus datos personales.

Comentarios de validación:

Una vez finalizada la revisión y validación de las preguntas, confirmo que en general están bien elaboradas, habiendo realizado algunas correcciones menores para mejorar la precisión y claridad. Considero que el contenido cumple con los objetivos planteados.

Datos informativos del experto

Nombres y Apellidos: Jonathan Mario Moreno Rivera
Profesión y cargo: Desarrollador de Software
Título universitario: Mg. En Ingeniería de Software y Sistemas Informáticos
Email: jmarmorenor@gmail.com
Fecha y hora de validación: 03/07/2025 - 9:00



PUCE

3/7/25, 21:19 Como: PAREDES ROMERO JENNIFER TAHIS - Outlook

Outlook

VALIDACIÓN DE INSTRUMENTOS

Desde PAREDES ROMERO JENNIFER TAHIS <jparedes@pucesd.edu.ec>
Fecha Jue 03/07/2025 21:14
Para LUIS JAVIER ULLOA MENESES <ljuloo@pucesd.edu.ec>
CC JORDAN STALIN HERRERA CHANCAY <jsherrera@pucesd.edu.ec>; WILLIAN JAVIER OCAMPO PAZOS <wjocampo@pucesd.edu.ec>

1 archivo adjunto (401 KB)
Validación de Instrumentos de recolección de datos_Herrera_Paredes (5).docx

Estimado Mg. Luis Ulloa

Reciba un cordial saludo.

Me dirijo a usted con el fin de solicitarle, muy respetuosamente, su apoyo para la validación de instrumentos de recolección de datos de nuestro trabajo de Titulación, llamado "Aplicación web con machine learning para la gestión de historial médico de las mascotas en el hospital veterinario Santa Martha en el Cantón Santo Domingo", el cual estamos desarrollando.


Como parte de los requisitos para la obtención de mi título en Ing. Tecnologías de la Información, su experiencia y conocimientos en el área de machine learning serían de gran ayuda para identificar posibles mejoras y asegurar la calidad del contenido.

Adjunto a este correo encontrará el documento en su versión preliminar. Estamos abiertos a cualquier observación o recomendación que considere pertinente.

Agradecemos de antemano su tiempo y colaboración.

Sin otro particular, nos despedimos con aprecio y respeto.

Atentamente:
Jordan Herrera
Jennifer Paredes


¡GRACIAS POR SU COLABORACIÓN!


Una vez finalizada su validación, puede realizar comentarios, sugerencias o la aprobación, además, es pertinente que agregue sus datos personales.


Comentarios de validación:


Se ha realizado la revisión y validación de los instrumentos, y se considera que son pertinentes para su trabajo de titulación

Datos informativos del experto

Nombres y Apellidos: Rodolfo Sirilo Córdova Gálvez
Profesión y cargo: Profesor Auxiliar II
Título universitario: Magister en Redes de Comunicaciones
Email: rscordovag@pucesd.edu.ec
Fecha y hora de validación: 10/07/2025 – 18h00



.....
Firma


ANEXOS DE CORREOS
3/7/25, 21:18 Correo: PAREDES ROMERO JENNIFER TAHIS - Outlook

 Outlook

VALIDACIÓN DE INSTRUMENTOS

Desde: PAREDES ROMERO JENNIFER TAHIS <jparedes@pucesd.edu.ec>
Fecha: Jue 03/07/2025 21:16
Para: RODOLFO SIRILO CORDOVA GALVEZ <rscordovag@pucesd.edu.ec>
CC: JORDAN STALIN HERRERA CHANCAY <jherrerac@pucesd.edu.ec>; WILLIAN JAVIER OCAMPO PAZOS <ojcampo@pucesd.edu.ec>

 1 archivo adjunto (401 KB)
Validación de Instrumentos de recolección de datos, Herrera, Paredes (3).docx

Estimado Mg. Rodolfo Cordova.
Reciba un cordial saludo.

Me dirijo a usted con el fin de solicitarle, muy respetuosamente, su apoyo para la validación de instrumentos de recolección de datos de nuestro trabajo de Titulación, llamado "Aplicación web con machine learning para la gestión de historial médico de las mascotas en el hospital veterinario Santa Martha en el Cantón Santo Domingo", el cual estamos desarrollando.


Como parte de los requisitos para la obtención de mi título en Ing. Tecnologías de la Información, su experiencia y conocimientos en el área serían de gran ayuda para identificar posibles mejoras y asegurar la calidad del contenido.

Adjunto a este correo encontrará el documento en su versión preliminar. Estamos abiertos a cualquier observación o recomendación que considere pertinente.

Agradecemos de antemano su tiempo y colaboración.

Sin otro particular, nos despedimos con aprecio y respeto.

Atentamente:
Jordan Herrera
Jenniffer Paredes


¡GRACIAS POR SU COLABORACIÓN!


Una vez finalizada su validación, puede realizar comentarios, sugerencias o la aprobación, además, es pertinente que agregue sus datos personales.

Comentarios de validación:

Se ha revisado lo solicitado, y los instrumentos son pertinentes para su aplicación.

Datos informativos del experto

Nombres y Apellidos: Luis Javier Ulloa Meneses
Profesión y cargo: Profesor Tiempo Completo (Titular Agregado I)
Título universitario: Magister en Big Data y Data Science
Email: ljulloa@pucesd.edu.ec
Fecha y hora de validación: 08/07/2025 – 10h00


.....
Firma

Anexo 5: Historias de Usuario e Historia técnica

Historia de Usuario	
Número: 1	Usuario: Veterinario / secretaria / Doctores asistentes / Dueños de la mascota
Nombre historia: Login	
Prioridad en negocio: 100	Riesgo en desarrollo: Bajo
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Jordan Herrera y Jennifer Paredes	
<p>Descripción:</p> <p>Como Veterinario / secretaria / Doctor / Dueño Quiero ingresar usuario y contraseña Para entrar a la plataforma y tener información de las mascotas.</p>	
<p>Escenario de prueba:</p> <p>Dado el ingreso de usuario o contraseña Cuando se presiona el botón "Iniciar sesión" Entonces se presenta un mensaje diciendo "Bienvenido, (usuario)".</p> <p>Dado el ingreso erróneo de usuario o contraseña Cuando se presiona el botón "Iniciar sesión" Entonces se despliega la alerta diciendo "El usuario o contraseña no existe, intente de nuevo".</p>	

Historia de Usuario	
Número: 2	Usuario: Veterinario / secretaria / Doctores asistentes / Dueños de la mascota
Nombre historia: Cierre de sesión	
Prioridad en negocio: 100	Riesgo en desarrollo: Bajo
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Jordan Herrera y Jennifer Paredes	
<p>Descripción:</p> <p>Como Veterinario / secretaria / Doctor / Dueño Quiero salir de la aplicación Para retirarme del puesto de trabajo.</p>	
<p>Escenario de prueba:</p> <p>Dado el cierre de sesión Cuando se presiona el botón "Cerrar sesión" Entonces se presenta un mensaje diciendo "¿Está seguro que desea cerrar sesión?" Entonces se cierra sesión.</p>	

Historia de Usuario	
Número: 5	Usuario: Veterinario / secretaria
Nombre historia: Eliminación de usuarios	
Prioridad en negocio: 96	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Jordan Herrera y Jennifer Paredes	
<p>Descripción:</p> <p>Como Veterinario / Secretaria Quiero eliminar usuarios Para tener conocimientos de usuarios que ya no son clientes.</p>	
<p>Escenario de prueba:</p> <p>Dado se elimina un usuario Cuando se presiona el botón "Eliminar" Entonces se permita eliminar al usuario seleccionado.</p>	

Historia de Usuario	
Número: 4	Usuario: Veterinario / secretaria
Nombre historia: Edición de usuarios	
Prioridad en negocio: 96	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Jordan Herrera y Jennifer Paredes	
<p>Descripción:</p> <p>Como Veterinario / Secretaria Quiero editar los datos de los usuarios Para mantener actualizado la información de los usuarios.</p>	
<p>Escenario de prueba:</p> <p>Dado se edita un usuario Cuando se presiona el botón "Guardar" Entonces se permita modificar la información del usuario.</p> <p>Dado se edite de manera incorrecta Cuando se presiona el botón "Guardar" Entonces se despliega una alerta diciendo "Error al editar usuario".</p>	

Historia de Usuario	
Número: 3	Usuario: Veterinario / secretaria
Nombre historia: Registro de usuarios	
Prioridad en negocio: 99	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Jordan Herrera y Jennifer Paredes	
<p>Descripción:</p> <p>Como Veterinario / Secretaria Quiero registrar a nuevos usuarios Para que puedan acceder a la información de su mascota.</p>	
<p>Escenario de prueba:</p> <p>Dado se registra un nuevo usuario Cuando se presiona el botón "Guardar" Entonces se permita registrar al nuevo usuario (dueño de la mascota).</p> <p>Dado el registro erróneo de un usuario Cuando se presiona el botón "Guardar" Entonces se despliega una alerta diciendo "Error al guardar usuario, revise los campos".</p>	

Historia de Usuario	
Número: 6	Usuario: Veterinario / secretaria
Nombre historia: Registro de pacientes (mascotas)	
Prioridad en negocio: 99	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Jordan Herrera y Jennifer Paredes	
<p>Descripción:</p> <p>Como Veterinario / Secretaria Quiero registrar a nuevos pacientes Para que la mascota cuente con un historial médico digital.</p>	
<p>Escenario de prueba:</p> <p>Dado se registra un nuevo paciente Cuando se presiona el botón "Guardar" Entonces se permita registrar al nuevo paciente.</p> <p>Dado el registro erróneo de una mascota Cuando se presiona el botón "Guardar" Entonces se despliega una alerta diciendo "Error al guardar paciente, revise los campos".</p>	

Historia de Usuario	
Número: 7	Usuario: Veterinario / secretaria
Nombre historia: Edición de pacientes (mascotas)	
Prioridad en negocio: 96	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Jordan Herrera y Jennifer Paredes	
<p>Descripción:</p> <p>Como Veterinario / Secretaria Quiero editar la información de los pacientes Para que los datos estén actualizados.</p>	
<p>Escenario de prueba:</p> <p>Dado se edita un paciente Cuando se presiona el botón "Guardar" Entonces se permita editar el registro del paciente.</p> <p>Dado se edite de manera errónea a un paciente Cuando se presiona el botón "Guardar" Entonces se despliega una alerta diciendo "Error al editar paciente".</p>	

Historia de Usuario	
Número: 8	Usuario: Veterinario / secretaria
Nombre historia: Eliminación de pacientes (mascotas)	
Prioridad en negocio: 96	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Jordan Herrera y Jennifer Paredes	
<p>Descripción:</p> <p>Como Veterinario / Secretaria Quiero eliminar pacientes Para tener información de pacientes que siguen siendo clientes.</p>	
<p>Escenario de prueba:</p> <p>Dado se elimine un paciente Cuando se presiona el botón "Eliminar" Entonces se permita eliminar el registro del paciente.</p>	

Historia de Usuario	
Número: 9	Usuario: Veterinario / Doctores asistentes
Nombre historia: Creación de historial médico	
Prioridad en negocio: 100	Riesgo en desarrollo: Muy Alto
Puntos estimados: 4	Iteración asignada: 1
Programador responsable: <u>Jordan</u> Herrera y <u>Jennifer</u> Paredes	
Descripción: Como Veterinario/ Doctor asistente Quiero crear historiales médicos para las mascotas registradas Para documentar los datos médicos de la mascota.	
Escenario de prueba: Dado crea un nuevo "historial médico" Cuando se presiona el botón "Guardar" Entonces se permite crear el historial médico de la mascota. Dado la creación errónea del historial médico Cuando se presiona el botón "Guardar" Entonces se despliega una alerta diciendo "Error al crear historial médico, revise los campos".	

Historia de Usuario	
Número: 10	Usuario: Veterinario / Doctores asistentes
Nombre historia: Edición de historial medico	
Prioridad en negocio: 96	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Jordan Herrera y Jennifer Paredes	
Descripción: Como Veterinario/ Doctor asistente Quiero editar historiales médicos para las mascotas registradas Para actualizar los datos médicos de la mascota.	
Escenario de prueba: Dado se actualiza un "historial médico" Cuando se presiona el botón "Editar" Entonces se permite guardar los cambios en el historial médico de la mascota. Dado se edite de manera incorrecta el historial médico Cuando se presiona el botón "Editar" Entonces se despliega una alerta diciendo "Error al editar historial médico".	

Historia de Usuario	
Número: 11	Usuario: Veterinario
Nombre historia: Eliminación de historial médico	
Prioridad en negocio: 96	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Jordan Herrera y Jennifer Paredes	
Descripción: Como veterinario Quiero eliminar un historial médico Para mantener actualizada mi lista de pacientes.	
Escenario de prueba: Dado se desee eliminar el historial Cuando se presiona el botón "Eliminar" Entonces se despliega una alerta diciendo "¿Está seguro de eliminar permanentemente el historial médico?" Entonces se despliega una alerta diciendo "Historial médico eliminado correctamente"	

Historia de Usuario	
Número: 12	Usuario: secretaria / Dueño de mascota
Nombre historia: Visualización de historial médico	
Prioridad en negocio: 90	Riesgo en desarrollo: Bajo
Puntos estimados: 1	Iteración asignada: 11
Programador responsable: Jordan Herrera y Jennifer Paredes	
Descripción: Como Secretaria Quiero visualizar los historiales médicos de los pacientes Para tener información por si es requerido por algún dueño de una mascota Como Dueño de mascota Quiero visualizar el historial médico de mi mascota Para tener información de la salud de la misma.	
Escenario de prueba: Dado se desee visualizar el historial médico de alguna mascota Cuando se presiona el botón "Historial médico" Entonces se permita visualizar el historial médico completo de la mascota	

Historia de Usuario	
Número: 13	Usuario: secretaria / Veterinario
Nombre historia: Agendamiento de citas	
Prioridad en negocio: 99	Riesgo en desarrollo: Muy Alto
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Jordan Herrera y Jennifer Paredes	
Descripción: Como Secretaria Quiero agendar citas Para mantener informado a los doctores y dueños de mascota. Como veterinario Quiero agendar citas Para tener organizado el tiempo que me demoro en cada paciente.	
Escenario de prueba: Dado se desee agendar una nueva cita Cuando se presiona el botón "Guardar" Entonces se permita agendar una nueva cita. Dado se ingrese erróneamente los datos para agendar una nueva cita Cuando se presiona el botón "Guardar" Entonces se despliegue un aviso diciendo "No se puede agendar cita"	

Historia de Usuario	
Número: 14	Usuario: secretaria / Veterinario
Nombre historia: Eliminación de citas	
Prioridad en negocio: 96	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Jordan Herrera y Jennifer Paredes	
Descripción: Como Secretaria Quiero eliminar citas cuando se requiera Para poder agendar otros pacientes Como veterinario Quiero eliminar citas Para tener disponibilidad para otros pacientes	
Escenario de prueba: Dado se desee eliminar una cita Cuando se presiona el botón "Eliminar" Entonces se permita eliminar la cita seleccionada. Dado se desee eliminar la cita Cuando se presiona el botón "Eliminar" Entonces se despliega una alerta diciendo "¿Está seguro de eliminar la cita médica?" Entonces se despliegue una alerta diciendo "Cita médica eliminada correctamente"	

Historia de Usuario	
Número: 15	Usuario: Dueño de mascota
Nombre historia: Visualización de citas	
Prioridad en negocio: 93	Riesgo en desarrollo: BAJO
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Jordan Herrera y Jennifer Paredes	
Descripción: Como Dueño Quiero visualizar las citas próximas de mi mascota Para estar pendiente y organizarme con mi tiempo personal	
Escenario de prueba: Dado se desee visualizar la cita próxima de la mascota Cuando se presiona el botón "Visualizar cita" Entonces se permita visualizar la siguiente cita que tiene agendada la mascota	

Historia de Usuario	
Número: 16	Usuario: Veterinario/ Doctor asistente
Nombre historia: Registro de tratamientos	
Prioridad en negocio: 99	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Jordan Herrera y Jennifer Paredes	
Descripción: Como Veterinario/ Doctor asistente Quiero crear un nuevo tratamiento Para tener en cuenta la medicina recomendada y el diagnóstico del paciente.	
Escenario de prueba: Dado se desee crear un nuevo tratamiento Cuando se presiona el botón "Guardar" Entonces se permita realizar la acción requerida. Dado se ingresa erróneamente los datos para crear un tratamiento Cuando se presiona el botón "Guardar" Entonces se despliega una alerta diciendo "No se puede crear tratamiento, revise los campos"	

Historia de Usuario	
Número: 17	Usuario: Veterinario/ Doctor asistente
Nombre historia: Edición de tratamientos	
Prioridad en negocio: 96	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Jordan Herrera y Jennifer Paredes	
Descripción:	
Como Veterinario/ Doctor asistente Quiero editar un tratamiento Para cambiar la medicación de la mascota en caso no reaccione de la manera esperada.	
Escenario de prueba:	
Dado se desee finalizar un tratamiento Cuando se presiona el botón "Finalizar" Entonces se permita realizar la acción requerida.	

Historia de Usuario	
Número: 18	Usuario: Veterinario/ Doctor asistente
Nombre historia: Finalización de tratamientos	
Prioridad en negocio: 96	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Jordan Herrera y Jennifer Paredes	
Descripción:	
Como Veterinario/ Doctor asistente Quiero finalizar un tratamiento Para culminar con la medicación de la mascota	
Escenario de prueba:	
Dado se desee finalizar un tratamiento Cuando se presiona el botón "Finalizar" Entonces se permita realizar la acción requerida	

Historia de Usuario	
Número: 19	Usuario: Dueño de mascota
Nombre historia: Visualizar tratamientos	
Prioridad en negocio: 93	Riesgo en desarrollo: Bajo
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Jordan Herrera y Jennifer Paredes	
Descripción:	
Como Dueño Quiero visualizar el tratamiento que está siguiendo mi mascota Para poder llevar un control de la medicina recomendada	
Escenario de prueba:	
Dado se visualizar el tratamiento Cuando se presiona el botón "Visualizar" Entonces se permita realizar la acción requerida.	


Historia de Usuario	
Número: 20	Usuario: Dueño de mascota
Nombre historia: Creación de Notas	
Prioridad en negocio: 96	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Jordan Herrera y Jennifer Paredes	
Descripción:	
Como Dueño Quiero poder escribir reacciones que tuvo mi mascota Para tener información relevante para que el médico pueda detallar un mejor diagnostico o cambio de tratamientos.	
Escenario de prueba:	
Dado se desee crear una nueva nota Cuando se presiona el botón "Nueva" Entonces se permita realizar la acción requerida Entonces se despliegue un botón diciendo "Nota guardada".	

Historia de Usuario	
Número: 21	Usuario: Dueño de mascota
Nombre historia: Edición de Notas	
Prioridad en negocio: 96	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Jordan Herrera y Jennifer Paredes	
Descripción:	
Como Dueño Quiero editar mis notas Para tener actualizar la información de la salud de mi mascota durante algún tratamiento.	
Escenario de prueba:	
Dado se desee editar una nota Cuando se presiona el botón "Editar" Entonces se permita realizar la acción requerida.	

Historia de Usuario	
Número: 22	Usuario: Dueño de mascota
Nombre historia: Eliminación de Notas	
Prioridad en negocio: 96	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Jordan Herrera y Jennifer Paredes	
Descripción:	
Como Dueño Quiero eliminar mis notas Para no tener notas innecesarias o información que ya comenté con el doctor.	
Escenario de prueba:	
Dado se elimine una nota Cuando se presiona el botón "Eliminar" Entonces se permita realizar la acción requerida.	



Historia de Usuario	
Número: 23	Usuario: Veterinario /Doctor Asistente
Nombre historia: Análisis del historial médico	
Prioridad en negocio: 100	Riesgo en desarrollo: Muy Alto
Puntos estimados: 8	Iteración asignada: 1
Programador responsable: Jordan Herrera y Jennifer Paredes	
Descripción:	
Como Veterinario /Doctor Quiero que el sistema analice los datos del historial médico y las notas ingresadas por el usuario, para obtener futuros diagnósticos.	
Escenario de prueba:	
Dado se analice los datos Cuando se presiona el botón "Analizar" Entonces aparecerá un mensaje de "Historial analizado con éxito".	
Dado el análisis incompleto del historial médico Cuando se presiona el botón "Analizar" Entonces se aparecerá un mensaje de error "Error historial incompleto".	

Historia de Usuario	
Número: 24	Usuario: Veterinario /Doctor Asistente
Nombre historia: Predicción de enfermedad	
Prioridad en negocio: 100	Riesgo en desarrollo: Muy Alto
Puntos estimados: 8	Iteración asignada: 1
Programador responsable: Jordan Herrera y Jennifer Paredes	
Descripción:	
Como Veterinario /Doctor Quiero que el sistema genere un posible diagnóstico médico en base al análisis previamente realizado.	
Escenario de prueba:	
Dado el posible diagnostico medico generado Cuando se presiona el botón "Generar diagnostico" Entonces aparecerá un mensaje con la enfermedad más probable que la mascota tendrá en el futuro.	
Dado que el modelo no puede estar seguro en ciertos casos Cuando se presiona el botón "Generar diagnostico" Entonces se mostrará un mensaje de error "No se ha podido determinar la enfermedad".	

 Pontificia Universidad Católica del Ecuador <small>Seréis mis testigos</small>		SANTO DOMINGO <small>TECNOLOGÍAS DE LA INFORMACIÓN</small>	
Historia Técnica			
Nombre historia: Seguridad e integridad de la aplicación			
Puntos estimados: 8		Iteración asignada: 1	
Programador responsable: Jordan Herrera y Jennifer Paredes			
Descripción: Esta fase se orienta a la implementación de mecanismos de seguridad en las diferentes capas de la aplicación Medical Pets , abarcando el frontend , el backend y la base de datos. Para la autenticación de los usuarios se emplea JSON Web Token (JWT) , lo que permite validar la identidad en cada solicitud realizada a los recursos protegidos del sistema. Asimismo, las contraseñas se almacenan utilizando bcrypt , técnica de cifrado que garantiza la protección de las credenciales frente a accesos no autorizados. Adicionalmente, se implementa un esquema de control de acceso basado en roles y permisos , el cual permite restringir las operaciones sensibles según el tipo de usuario.			
Criterio de aceptación: <ul style="list-style-type: none"> ✓ Generar y verificar tokens JWT en cada petición protegida del backend ✓ Hashear todas las contraseñas con bcrypt antes de almacenarlas en la base de datos ✓ Permitir el acceso a módulos definidos por roles ✓ Establecer conexión SSL entre el servidor y la base de datos en producción 			
Beneficios de justificación: Asegurar que la aplicación proteja adecuadamente la información personal de los usuarios, así como sus credenciales de acceso y los datos asociados a las mascotas. Para ello, se implementan buenas prácticas de seguridad en cada una de las capas del sistema, con el propósito de prevenir accesos no autorizados, la interceptación de información durante la transmisión de datos y la generación de registros fraudulentos dentro de la plataforma.			
Impacto en el proyecto: En ausencia de estas medidas de seguridad, la aplicación podría quedar expuesta a múltiples vulnerabilidades. Entre ellas, el acceso no autorizado a información privada de los usuarios y a los datos de las mascotas, la posible exposición de contraseñas en caso de una brecha de seguridad, y la ejecución de operaciones administrativas por parte de usuarios sin los privilegios correspondientes.			
Dependencias: <ul style="list-style-type: none"> ✓ Librerías de Python: passlib, Python-dotenv ✓ Variables de entorno: SECRET_KEY, ALGORITHM ✓ Base de datos MySQL: se gestiona mediante el ORM SQLAlchemy con tablas, user, roles ✓ Librerías: Vue Router y Pinia para el control de sesiones del frontend 			
Tareas: <ul style="list-style-type: none"> ✓ Autenticación basada en JWT (JSON Web Tokens) con FastAPI ✓ Encriptación de contraseñas usando hashing unidireccional ✓ Middleware de Autorización para los Roles y Permisos ✓ Protección de rutas por medio de Vue-Router ✓ Conexión segura a la base de datos utilizando el aislamiento de credenciales mediante SQLAlchemy y .env 			

Anexo 6: Pruebas de aceptación

 Pontificia Universidad Católica del Ecuador <small>Seréis mis testigos</small>		SANTO DOMINGO <small>DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN</small>	
CASO DE PRUEBA 01		Fecha: 15/07/2025	
Nombre caso de prueba: Inicio de sesión	Sprint: 1		
Módulo/sección a evaluar: Login	Historia de usuario asociada: 1		
Técnica de prueba: Caja Negra <input checked="" type="checkbox"/> Caja Blanca <input type="checkbox"/>	Tipo: Prueba de Aceptación		
Descripción:			
Dado el ingreso de usuario o contraseña Cuando se presiona el botón "Iniciar sesión" Entonces se presenta un mensaje diciendo "Bienvenido, (usuario)".			
Dado el ingreso erróneo de usuario o contraseña Cuando se presiona el botón "Iniciar sesión" Entonces se despliega la alerta diciendo "El usuario o contraseña no existe, inténtelo de nuevo".			
Pre-condiciones			
<ul style="list-style-type: none"> • Tener acceso a internet • Ser paciente del hospital veterinario 			
Pasos y condiciones de ejecución			
<ul style="list-style-type: none"> • Dirijirse a la aplicación web "Medical Pets" • Ingresar usuario y contraseña • Pulsar el botón "Iniciar sesión" 			
Resultado esperado			
<ul style="list-style-type: none"> • Se presenta la pagina principal 			
Estado de prueba		Éxito	Falló
		SI	
Errores asociados:			
 Doc. Patricio Intrigo PRODUCT OWNER			
<small> Dirección: Vía a Chino Km. 2 Código postal: 23030 / Teléfono: (593-99320421) Santo Domingo, Ecuador - https://www.puc.edu.ec/ </small>			

 Pontificia Universidad Católica del Ecuador <small>Seréis mis testigos</small>		SANTO DOMINGO <small>DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN</small>	
CASO DE PRUEBA 02		Fecha: 15/07/2025	
Nombre caso de prueba: Cierre de sesión	Sprint: 1		
Módulo/sección a evaluar: Cierre de sesión	Historia de usuario asociada: 2		
Técnica de prueba: Caja Negra <input checked="" type="checkbox"/> Caja Blanca <input type="checkbox"/>	Tipo: Prueba de Aceptación		
Descripción:			
Dado el cierre de sesión Cuando se presiona el botón "Cerrar sesión" Entonces se presenta un mensaje diciendo "¿Está seguro que desea cerrar sesión?" Entonces se cierra sesión.			
Pre-condiciones			
<ul style="list-style-type: none"> • Tener acceso a internet • Estar logueado en la aplicación 			
Pasos y condiciones de ejecución			
<ul style="list-style-type: none"> • Dirijirse al botón "Cerrar sesión" 			
Resultado esperado			
<ul style="list-style-type: none"> • Se cierra la sesión • Se despliega un aviso diciendo "¿Está seguro que desea cerrar sesión?" • Si es positivo se cierra la sesión. 			
Estado de prueba		Éxito	Falló
		SI	
Errores asociados:			
 Doc. Patricio Intrigo PRODUCT OWNER			
<small> Dirección: Vía a Chino Km. 2 Código postal: 23030 / Teléfono: (593-99320421) Santo Domingo, Ecuador - https://www.puc.edu.ec/ </small>			

Pontificia Universidad Católica del Ecuador **SANTO DOMINGO** DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN

CASO DE PRUEBA 03 Fecha: 15/07/2025

Nombre caso de prueba: Registro de usuarios Sprint: 1
 Módulo/sección a evaluar: Registro de usuarios Historia de usuario asociada: 3
 Técnica de prueba: Caja Negra Caja Blanca Tipo: Prueba de Aceptación

Descripción:
 Dado se registra un nuevo usuario Cuando se presiona el botón "Guardar" Entonces se despliega un mensaje diciendo "Usuario registrado exitosamente" Entonces se registra al nuevo usuario (dueño de la mascota).
 Dado el registro erróneo de un usuario Cuando se presiona el botón "Guardar" Entonces se despliega una alerta diciendo "Error al guardar usuario, revise los campos".

Pre-condiciones
 • Tener acceso a internet
 • Estar logado en la aplicación

Pasos y condiciones de ejecución
 • Dirigirse a el módulo de usuarios
 • Dirigirse al botón "Nuevo"
 • Ingresar la información para crear el usuario

Resultado esperado
 • Poder registrar un nuevo usuario
 • Se guarde exitosamente el usuario

Estado de prueba	Éxito	Falló
	SI	

Errores asociados:


 Doc. Patricio Intriago
 PRODUCT OWNER

Dirección: Vía a Chone Km. 2
 Código postal: 23020 / Teléfono: (05) 49020425
 Santo Domingo, Ecuador / <https://pucsd.edu.ec/>

Pontificia Universidad Católica del Ecuador **SANTO DOMINGO** DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN

CASO DE PRUEBA 04 Fecha: 15/07/2025

Nombre caso de prueba: Edición de usuarios Sprint: 1
 Módulo/sección a evaluar: Edición de usuarios Historia de usuario asociada: 4
 Técnica de prueba: Caja Negra Caja Blanca Tipo: Prueba de Aceptación

Descripción:
 Dado se edita un usuario Cuando se presiona el botón "Guardar" Entonces se permite modificar la información del usuario.
 Dado se edita de manera incorrecta Cuando se presiona el botón "Guardar" Entonces se despliega una alerta diciendo "Error al editar usuario".


Pre-condiciones
 • Tener acceso a internet
 • Estar logado en la aplicación

Pasos y condiciones de ejecución
 • Dirigirse a el módulo de usuarios
 • Dirigirse al botón "Editar"
 • Ingresar la información para editar el usuario

Resultado esperado
 • Poder editar la información del usuario
 • Se edite exitosamente el usuario

Estado de prueba	Éxito	Falló
	SI	

Errores asociados:


 Doc. Patricio Intriago
 PRODUCT OWNER

Dirección: Vía a Chone Km. 2
 Código postal: 23020 / Teléfono: (05) 49020425
 Santo Domingo, Ecuador / <https://pucsd.edu.ec/>

Pontificia Universidad Católica del Ecuador **SANTO DOMINGO** DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN

CASO DE PRUEBA 05 Fecha: 15/07/2025

Nombre caso de prueba: Eliminación de usuarios Sprint: 1
 Módulo/sección a evaluar: Eliminación de usuarios Historia de usuario asociada: 5
 Técnica de prueba: Caja Negra Caja Blanca Tipo: Prueba de Aceptación

Descripción:
 Dado se elimina un usuario Cuando se presiona el botón "Eliminar" Entonces se despliega un mensaje diciendo "Está seguro que desea eliminar el usuario? Entonces permita eliminar al usuario seleccionado.


Pre-condiciones
 • Tener acceso a internet
 • Estar logado en la aplicación

Pasos y condiciones de ejecución
 • Dirigirse a el módulo de usuarios
 • Dirigirse al botón "Eliminar"

Resultado esperado
 • Poder eliminar al usuario
 • Se elimine el usuario

Estado de prueba	Éxito	Falló
	SI	

Errores asociados:


 Doc. Patricio Intriago
 PRODUCT OWNER

Dirección: Vía a Chone Km. 2
 Código postal: 23020 / Teléfono: (05) 49020425
 Santo Domingo, Ecuador / <https://pucsd.edu.ec/>

Pontificia Universidad Católica del Ecuador **SANTO DOMINGO** DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN

CASO DE PRUEBA 06 Fecha: 15/07/2025

Nombre caso de prueba: Registro de pacientes Sprint: 1
 (MASC0145)
 Módulo/sección a evaluar: Registro de pacientes Historia de usuario asociada: 6
 (MASC0145)
 Técnica de prueba: Caja Negra Caja Blanca Tipo: Prueba de Aceptación

Descripción:
 Dado se registra un nuevo paciente Cuando se presiona el botón "Guardar" Entonces se despliega un mensaje diciendo "Paciente registrado exitosamente" Entonces se registra al nuevo paciente.
 Dado el registro erróneo de un usuario Cuando se presiona el botón "Guardar" Entonces se despliega una alerta diciendo "Error al guardar paciente, revise los campos".

Pre-condiciones
 • Tener acceso a internet
 • Estar logado en la aplicación

Pasos y condiciones de ejecución
 • Dirigirse a el módulo de pacientes
 • Dirigirse al botón "Nuevo"
 • Ingresar la información para crear el paciente

Resultado esperado
 • Poder registrar un nuevo paciente
 • Se guarde exitosamente el nuevo paciente

Estado de prueba	Éxito	Falló
	SI	

Errores asociados:


 Doc. Patricio Intriago
 PRODUCT OWNER

Dirección: Vía a Chone Km. 2
 Código postal: 23020 / Teléfono: (05) 49020425
 Santo Domingo, Ecuador / <https://pucsd.edu.ec/>

Pontificia Universidad Católica del Ecuador | **SANTO DOMINGO** | DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN

CASO DE PRUEBA 07 Fecha: 15/07/2025

Nombre caso de prueba: Edición de pacientes Sprint: 1
 Módulo/sección a evaluar: Edición de pacientes Historia de usuario asociada: 7
 Técnica de prueba: Caja Negra Caja Blanca Tipo: Prueba de Aceptación

Descripción:
 Dado se edita un paciente Cuando se presiona el botón "Guardar" Entonces se permite modificar la información del paciente.
 Dado se edita de manera incorrecta Cuando se presiona el botón "Guardar" Entonces se despliega una alerta diciendo "Error al editar paciente".

Pre-condiciones

- Tener acceso a internet
- Estar logeado en la aplicación

Pasos y condiciones de ejecución


- Dirigirse a el módulo de pacientes
- Dirigirse al botón "Editar"
- Ingresar la información para editar el paciente

Resultado esperado

- Poder editar la información del paciente
- Se edite exitosamente el paciente

Estado de prueba	Éxito	Falló
	SI	

Errores asociados:


 Doc. Patricio Intrigo
 PRODUCT OWNER

Dirección: Vía a Chone Km. 2
 Código postal: 23030 / Teléfono: (051-91520421)
 Santo Domingo, Ecuador / <https://www.uc.edu.ec/>

Pontificia Universidad Católica del Ecuador | **SANTO DOMINGO** | DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN

CASO DE PRUEBA 08 Fecha: 15/07/2025

Nombre caso de prueba: Eliminación de pacientes Sprint: 1
 (mascotas)
 Módulo/sección a evaluar: Eliminación de pacientes Historia de usuario asociada: 8
 (mascotas)
 Técnica de prueba: Caja Negra Caja Blanca Tipo: Prueba de Aceptación

Descripción:
 Dado se elimina un paciente Cuando se presiona el botón "Eliminar" Entonces se despliega un mensaje diciendo "Está seguro que desea eliminar el paciente? Entonces permite eliminar al paciente seleccionado".

Pre-condiciones

- Tener acceso a internet
- Estar logeado en la aplicación

Pasos y condiciones de ejecución


- Dirigirse a el módulo de pacientes
- Dirigirse al botón "Eliminar"

Resultado esperado

- Poder eliminar al paciente
- Se elimine al paciente

Estado de prueba	Éxito	Falló
	SI	

Errores asociados:


 Doc. Patricio Intrigo
 PRODUCT OWNER

Dirección: Vía a Chone Km. 2
 Código postal: 23030 / Teléfono: (051-91520421)
 Santo Domingo, Ecuador / <https://www.uc.edu.ec/>

Pontificia Universidad Católica del Ecuador | **SANTO DOMINGO** | DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN

CASO DE PRUEBA 09 Fecha: 15/07/2025

Nombre caso de prueba: Creación de historial médico Sprint: 2
 Módulo/sección a evaluar: Creación de historial médico Historia de usuario asociada: 9
 Técnica de prueba: Caja Negra Caja Blanca Tipo: Prueba de Aceptación

Descripción:
 Dado crea un nuevo "historial médico" Cuando se presiona el botón "Guardar" Entonces se despliega un mensaje diciendo "Historial médico creado exitosamente"
 Dado la creación errónea del historial médico Cuando se presiona el botón "Guardar" Entonces se despliega una alerta diciendo "Error al crear historial médico, revise los campos".

Pre-condiciones

- Tener acceso a internet
- Estar logeado en la aplicación

Pasos y condiciones de ejecución


-

Resultado esperado

- Poder crear un historial médico

Estado de prueba	Éxito	Falló
	SI	

Errores asociados:


 Doc. Patricio Intrigo
 PRODUCT OWNER

Dirección: Vía a Chone Km. 2
 Código postal: 23030 / Teléfono: (051-91520421)
 Santo Domingo, Ecuador / <https://www.uc.edu.ec/>

Pontificia Universidad Católica del Ecuador | **SANTO DOMINGO** | DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN

CASO DE PRUEBA 10 Fecha: 15/07/2025

Nombre caso de prueba: Edición de historial médico Sprint: 2
 Módulo/sección a evaluar: Edición de historial médico Historia de usuario asociada: 10
 Técnica de prueba: Caja Negra Caja Blanca Tipo: Prueba de Aceptación

Descripción:
 Dado se edita un historial médico Cuando se presiona el botón "Guardar" Entonces se permite modificar la información del historial médico.
 Dado se edita de manera incorrecta Cuando se presiona el botón "Guardar" Entonces se despliega una alerta diciendo "Error al editar historial médico".

Pre-condiciones

- Tener acceso a internet
- Estar logeado en la aplicación

Pasos y condiciones de ejecución


-

Resultado esperado


- Poder editar el historial médico

Estado de prueba	Éxito	Falló
	SI	

Errores asociados:


 Doc. Patricio Intrigo
 PRODUCT OWNER

Dirección: Vía a Chone Km. 2
 Código postal: 23030 / Teléfono: (051-91520421)
 Santo Domingo, Ecuador / <https://www.uc.edu.ec/>



Pontificia Universidad Católica del Ecuador
Servicio más íntegro

SANTO DOMINGO
DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN

CASO DE PRUEBA 11 Fecha: 15/07/2025

Nombre caso de prueba: Eliminación de historial médico	Sprint: 2
Módulo/sección a evaluar: Eliminación de historial médico	Historia de usuario asociada: 11
Técnica de prueba: Caja Negra <input checked="" type="checkbox"/> Caja Blanca <input type="checkbox"/>	Tipo: Prueba de Aceptación

Descripción:

Dado se elimina un historial médico Cuando se presiona el botón "Eliminar" Entonces se despliega un mensaje diciendo "Está seguro que desea eliminar el historial médico? Entonces permito eliminar el historial médico seleccionado."

Pre-condiciones

- Tener acceso a internet
- Estar logado en la aplicación

Pasos y condiciones de ejecución


-

Resultado esperado

- Poder eliminar el historial médico

Estado de prueba	Éxito	Falló
	SI	


Errores asociados:




 Dr. Patricio Intriago
 PRODUCT OWNER

Dirección: Vía a Chone Km. 2
 Código postal: 23030 / Teléfono: (05)-91230425
 Santo Domingo, Ecuador / <https://www.puc.edu.ec/>







Pontificia Universidad Católica del Ecuador
Servicio más íntegro

SANTO DOMINGO
DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN

CASO DE PRUEBA 12 Fecha: 15/07/2025

Nombre caso de prueba: Visualización de historial médico	Sprint: 2
Módulo/sección a evaluar: Visualización de historial médico	Historia de usuario asociada: 12
Técnica de prueba: Caja Negra <input checked="" type="checkbox"/> Caja Blanca <input type="checkbox"/>	Tipo: Prueba de Aceptación

Descripción:

Dado se desea visualizar el historial médico de alguna mascota Cuando se presiona el botón "Historial médico" Entonces se permita visualizar el historial médico completo de la mascota.

Pre-condiciones

- Tener acceso a internet
- Estar logado en la aplicación

Pasos y condiciones de ejecución

-

Resultado esperado

- Poder visualizar el historial médico

Estado de prueba	Éxito	Falló
	SI	


Errores asociados:




 Dr. Patricio Intriago
 PRODUCT OWNER

Dirección: Vía a Chone Km. 2
 Código postal: 23030 / Teléfono: (05)-91230425
 Santo Domingo, Ecuador / <https://www.puc.edu.ec/>







Pontificia Universidad Católica del Ecuador
Servicio más íntegro

SANTO DOMINGO
DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN

CASO DE PRUEBA 13 Fecha: 15/07/2025

Nombre caso de prueba: Agendamiento de citas	Sprint: 2
Módulo/sección a evaluar: Agendamiento de citas	Historia de usuario asociada: 13
Técnica de prueba: Caja Negra <input checked="" type="checkbox"/> Caja Blanca <input type="checkbox"/>	Tipo: Prueba de Aceptación

Descripción:

Dado se desea agendar una nueva cita Cuando se presiona el botón "Guardar" Entonces se permite agendar una nueva cita.

Dado se ingrese correctamente los datos para agendar una nueva cita Cuando se presiona el botón "Guardar" Entonces se despliega un aviso diciendo "No se puede agendar cita"

Pre-condiciones

- Tener acceso a internet
- Estar logado en la aplicación

Pasos y condiciones de ejecución


-

Resultado esperado

- Poder agendar citas.


Estado de prueba	Éxito	Falló
	SI	


Errores asociados:




 Dr. Patricio Intriago
 PRODUCT OWNER

Dirección: Vía a Chone Km. 2
 Código postal: 23030 / Teléfono: (05)-91230425
 Santo Domingo, Ecuador / <https://www.puc.edu.ec/>







Pontificia Universidad Católica del Ecuador
Servicio más íntegro

SANTO DOMINGO
DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN

CASO DE PRUEBA 14 Fecha: 15/07/2025

Nombre caso de prueba: Eliminación de citas	Sprint: 2
Módulo/sección a evaluar: Eliminación de citas	Historia de usuario asociada: 14
Técnica de prueba: Caja Negra <input checked="" type="checkbox"/> Caja Blanca <input type="checkbox"/>	Tipo: Prueba de Aceptación

Descripción:

Dado se desea eliminar una cita Cuando se presiona el botón "Eliminar" Entonces se permite eliminar la cita seleccionada.

Dado se desea eliminar la cita Cuando se presiona el botón "Eliminar" Entonces se despliega una alerta diciendo "¿Está seguro de eliminar la cita médica?" Entonces se despliega un alerta diciendo "Cita médica eliminada correctamente"

Pre-condiciones

- Tener acceso a internet
- Estar logado en la aplicación

Pasos y condiciones de ejecución


-

Resultado esperado

- Poder eliminar citas.


Estado de prueba	Éxito	Falló
	SI	


Errores asociados:



 Dr. Patricio Intriago
 PRODUCT OWNER

Dirección: Vía a Chone Km. 2
 Código postal: 23030 / Teléfono: (05)-91230425
 Santo Domingo, Ecuador / <https://www.puc.edu.ec/>





Pontificia Universidad Católica del Ecuador | **SANTO DOMINGO** | DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN

CASO DE PRUEBA 15 Fecha: 15/07/2025

Nombre caso de prueba: Visualización de citas Sprint: 2
 Módulo/sección a evaluar: Visualización de citas Historia de usuario asociada: 15

Técnica de prueba: Caja Negra Caja Blanca Tipo: Prueba de Aceptación

Descripción:
 Dado se desee visualizar la cita próxima de la mascota Cuando se presiona el botón "Visualizar cita" Entonces se permita visualizar la siguiente cita que tiene agendada la mascota

Pre-condiciones
 • Tener acceso a internet
 • Estar logeado en la aplicación

Pasos y condiciones de ejecución
 •

Resultado esperado
 • Poder visualizar las citas

Estado de prueba	Éxito	Falló
	SI	

Errores asociados:

[Firma]
 Doc. Patricio Intrigo
 PRODUCT OWNER

Dirección: Vía a Chone Km. 2
 Código postal: 23020 / Teléfono: (05) 49330421
 Santo Domingo, Ecuador / <https://www.ucde.edu.ec/>

30 años de existencia **fe, razón y corazón**

Pontificia Universidad Católica del Ecuador | **SANTO DOMINGO** | DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN

CASO DE PRUEBA 16 Fecha: 15/07/2025

Nombre caso de prueba: Registro de tratamientos Sprint: 2
 Módulo/sección a evaluar: Registro de tratamientos Historia de usuario asociada: 16

Técnica de prueba: Caja Negra Caja Blanca Tipo: Prueba de Aceptación

Descripción:
 Dado se desee crear un nuevo tratamiento Cuando se presiona el botón "Guardar" Entonces se permita realizar la acción requerida.

Dado se ingresa erróneamente los datos para crear un tratamiento Cuando se presiona el botón "Guardar" Entonces se despliega una alerta diciendo "No se puede crear tratamiento, revise los campos"

Pre-condiciones
 • Tener acceso a internet
 • Estar logeado en la aplicación

Pasos y condiciones de ejecución
 •

Resultado esperado
 • Poder registrar los tratamientos.

Estado de prueba	Éxito	Falló
	SI	

Errores asociados:

[Firma]
 Doc. Patricio Intrigo
 PRODUCT OWNER

Dirección: Vía a Chone Km. 2
 Código postal: 23020 / Teléfono: (05) 49330421
 Santo Domingo, Ecuador / <https://www.ucde.edu.ec/>

30 años de existencia **fe, razón y corazón**

Pontificia Universidad Católica del Ecuador | **SANTO DOMINGO** | DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN

CASO DE PRUEBA 17 Fecha: 15/07/2025

Nombre caso de prueba: Edición de tratamientos Sprint: 2
 Módulo/sección a evaluar: Edición de tratamientos Historia de usuario asociada: 17

Técnica de prueba: Caja Negra Caja Blanca Tipo: Prueba de Aceptación

Descripción:
 Dado se desee finalizar un tratamiento Cuando se presiona el botón "Finalizar" Entonces se permita realizar la acción requerida.

Pre-condiciones
 • Tener acceso a internet
 • Estar logeado en la aplicación

Pasos y condiciones de ejecución
 •

Resultado esperado
 • Poder editar los tratamientos.

Estado de prueba	Éxito	Falló
	SI	

Errores asociados:

[Firma]
 Doc. Patricio Intrigo
 PRODUCT OWNER

Dirección: Vía a Chone Km. 2
 Código postal: 23020 / Teléfono: (05) 49330421
 Santo Domingo, Ecuador / <https://www.ucde.edu.ec/>

30 años de existencia **fe, razón y corazón**

Pontificia Universidad Católica del Ecuador | **SANTO DOMINGO** | DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN

CASO DE PRUEBA 18 Fecha: 15/07/2025

Nombre caso de prueba: Finalización de tratamientos Sprint: 2
 Módulo/sección a evaluar: Finalización de tratamientos Historia de usuario asociada: 18

Técnica de prueba: Caja Negra Caja Blanca Tipo: Prueba de Aceptación

Descripción:
 Dado se desee finalizar un tratamiento Cuando se presiona el botón "Finalizar" Entonces se permita realizar la acción requerida.

Pre-condiciones
 • Tener acceso a internet
 • Estar logeado en la aplicación

Pasos y condiciones de ejecución
 •

Resultado esperado
 • Poder finalizar los tratamientos.


Estado de prueba	Éxito	Falló
	SI	

Errores asociados:

[Firma]
 Doc. Patricio Intrigo
 PRODUCT OWNER

Dirección: Vía a Chone Km. 2
 Código postal: 23020 / Teléfono: (05) 49330421
 Santo Domingo, Ecuador / <https://www.ucde.edu.ec/>

30 años de existencia **fe, razón y corazón**



Pontificia Universidad Católica del Ecuador
SANTO DOMINGO
Serán mis testigos

DIRECCIÓN DE INVESTIGACIÓN
 VINCULACIÓN E INNOVACIÓN

CASO DE PRUEBA 19 Fecha: 15/07/2025

Nombre caso de prueba: Visualizar tratamientos Sprint: 2
 Módulo/sección a evaluar: Visualizar tratamientos Historia de usuario asociada: 19

Técnica de prueba: Caja Negra Caja Blanca Tipo: Prueba de Aceptación

Descripción:

Dado se visualizar el tratamiento Cuando se presiona el botón "Visualizar" Entonces se permita realizar la acción requerida.

Pre-condiciones

- Tener acceso a internet
- Estar logado en la aplicación

Pasos y condiciones de ejecución


-

Resultado esperado



- Poder visualizar el historial médico


Estado de prueba	Éxito	Falló
	SI	

Errores asociados:


 Doc. Patricio Intriago
 PRODUCT OWNER

Dirección: Vía a Chone Km. 2
 Código postal: 23003 / Teléfono: (05) 91281421
 Santo Domingo, Ecuador / <https://pucsd.edu.ec/>



Pontificia Universidad Católica del Ecuador
SANTO DOMINGO
Serán mis testigos

DIRECCIÓN DE INVESTIGACIÓN
 VINCULACIÓN E INNOVACIÓN

CASO DE PRUEBA 20 Fecha: 15/07/2025

Nombre caso de prueba: Creación de Notas Sprint: 2
 Módulo/sección a evaluar: Creación de Notas Historia de usuario asociada: 20

Técnica de prueba: Caja Negra Caja Blanca Tipo: Prueba de Aceptación

Descripción:

Dado se desea crear una nueva nota Cuando se presiona el botón "Nueva" Entonces se permita realizar la acción requerida Entonces se despliegue un botón diciendo "Nota guardada".

Pre-condiciones

- Tener acceso a internet
- Estar logado en la aplicación

Pasos y condiciones de ejecución


-

Resultado esperado



- Poder crear notas.


Estado de prueba	Éxito	Falló
	SI	

Errores asociados:


 Doc. Patricio Intriago
 PRODUCT OWNER

Dirección: Vía a Chone Km. 2
 Código postal: 23003 / Teléfono: (05) 91281421
 Santo Domingo, Ecuador / <https://pucsd.edu.ec/>



Pontificia Universidad Católica del Ecuador
SANTO DOMINGO
Serán mis testigos

DIRECCIÓN DE INVESTIGACIÓN
 VINCULACIÓN E INNOVACIÓN

CASO DE PRUEBA 21 Fecha: 15/07/2025

Nombre caso de prueba: Edición de Notas Sprint: 2
 Módulo/sección a evaluar: Edición de Notas Historia de usuario asociada: 21

Técnica de prueba: Caja Negra Caja Blanca Tipo: Prueba de Aceptación

Descripción:

Dado se desea editar una nota Cuando se presiona el botón "Editar" Entonces se permita realizar la acción requerida.

Pre-condiciones

- Tener acceso a internet
- Estar logado en la aplicación

Pasos y condiciones de ejecución


-

Resultado esperado



- Poder editar las notas.


Estado de prueba	Éxito	Falló
	SI	

Errores asociados:


 Doc. Patricio Intriago
 PRODUCT OWNER

Dirección: Vía a Chone Km. 2
 Código postal: 23003 / Teléfono: (05) 91281421
 Santo Domingo, Ecuador / <https://pucsd.edu.ec/>



Pontificia Universidad Católica del Ecuador
SANTO DOMINGO
Serán mis testigos

DIRECCIÓN DE INVESTIGACIÓN
 VINCULACIÓN E INNOVACIÓN

CASO DE PRUEBA 22 Fecha: 15/07/2025

Nombre caso de prueba: Eliminación de notas Sprint: 2
 Módulo/sección a evaluar: Eliminación de notas Historia de usuario asociada: 22

Técnica de prueba: Caja Negra Caja Blanca Tipo: Prueba de Aceptación

Descripción:

Dado se elimine una nota Cuando se presiona el botón "Eliminar" Entonces se permita realizar la acción requerida.

Pre-condiciones

- Tener acceso a internet
- Estar logado en la aplicación

Pasos y condiciones de ejecución


-

Resultado esperado



- Poder eliminar las notas.


Estado de prueba	Éxito	Falló
	SI	

Errores asociados:


 Doc. Patricio Intriago
 PRODUCT OWNER

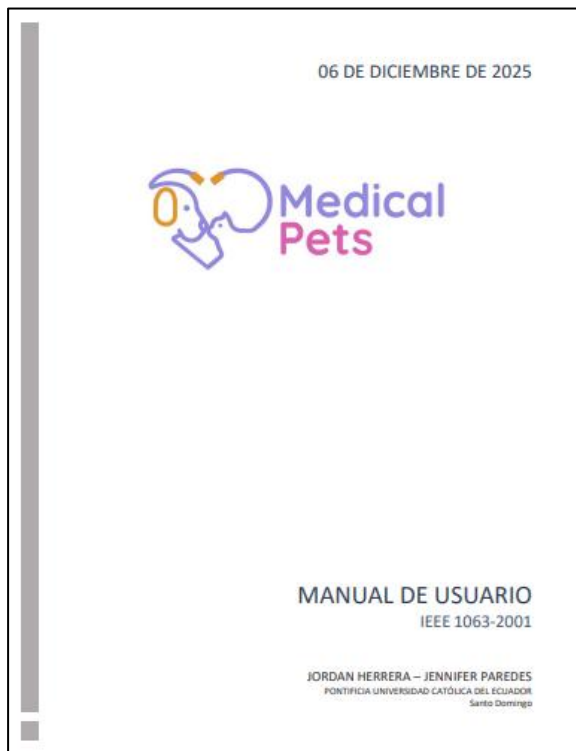
Dirección: Vía a Chone Km. 2
 Código postal: 23003 / Teléfono: (05) 91281421
 Santo Domingo, Ecuador / <https://pucsd.edu.ec/>

 Pontificia Universidad Católica del Ecuador Seréis mis testigos	SANTO DOMINGO	DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN	
		CASO DE PRUEBA 23 Fecha: 15/07/2025	
Nombre caso de prueba: Análisis del historial médico		Sprint: 3	
Módulo/sección a evaluar: Predicción con IA		Historia de usuario asociada: 23	
Técnica de prueba: Caja Negra <input checked="" type="checkbox"/> Caja Blanca <input type="checkbox"/>		Tipo: Prueba de Aceptación	
Descripción: Dado se ingresa al módulo "Predicción con IA" Cuando se filtre la búsqueda con el nombre del dueño Entonces se presentan las mascotas que posee el dueño.			
Pre-condiciones <ul style="list-style-type: none"> Tener acceso a internet Ser paciente del hospital veterinario Tener al dueño creado en la aplicación web 			
Pasos y condiciones de ejecución <ul style="list-style-type: none"> Dirigirse a la aplicación web "Medical Pets" Ingresar usuario y contraseña Pulsar el botón "Iniciar sesión" Tener acceso al módulo de predicción con IA Filtrar dueño de mascota Escoge mascota 			
Resultado esperado <ul style="list-style-type: none"> Se presenta los animales asociados al dueño que tenga un historial médico registrado 			
Estado de prueba		Éxito	Falló
Errores asociados:		SI	
			
Doc. Patricio Intriago PRODUCT OWNER			

 Pontificia Universidad Católica del Ecuador Seréis mis testigos	SANTO DOMINGO	DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN	
		CASO DE PRUEBA 24 Fecha: 15/07/2025	
Nombre caso de prueba: Predicción de enfermedad		Sprint: 3	
Módulo/sección a evaluar: Predicción con IA		Historia de usuario asociada: 24	
Técnica de prueba: Caja Negra <input checked="" type="checkbox"/> Caja Blanca <input type="checkbox"/>		Tipo: Prueba de Aceptación	
Descripción: Dado se ingresa al módulo "Predicción con IA" Cuando se filtre la búsqueda con el nombre del dueño Entonces se presentan las mascotas que posee el dueño Entonces se escoge la mascota deseada Entonces de presiones el botón "Hacer predicción" Entonces se visualice una posible predicción.			
Pre-condiciones <ul style="list-style-type: none"> Tener acceso a internet Ser paciente del hospital veterinario 			
Pasos y condiciones de ejecución <ul style="list-style-type: none"> Dirigirse a la aplicación web "Medical Pets" Ingresar usuario y contraseña Pulsar el botón "Iniciar sesión" Tener acceso al módulo de predicción con IA Filtrar dueño de mascota Escoge mascota Hacer click en "Hacer predicción" 			
Resultado esperado <ul style="list-style-type: none"> Se presente la posible predicción de enfermedad de la mascota 			
Estado de prueba		Éxito	Falló
Errores asociados:		SI	
			
Doc. Patricio Intriago PRODUCT OWNER			

Anexo 7: Manual de Usuario



MEDICAL PETS

REGISTRO DE CAMBIOS

FECHA	USUARIO	VERSIÓN	ACCIONES
06-DIC-25	JENNIFER	1.0	CREACIÓN

Manual de usuario | IEEE 1063 – 2001 1

MEDICAL PETS

Tabla de Contenidos

- 1 INTRODUCCIÓN 3
- 2 CONCEPTO DE LAS OPERACIONES..... 3
- 3 PROCEDIMIENTOS..... 3
 - 3.1 Acceso a la aplicación web..... 3
 - 3.2 Ventana Principal..... 4
 - 3.3 Modulo Usuario..... 4
 - 3.3.1 Agregar nuevo usuario..... 5
 - 3.3.2 Edición de usuario..... 6
 - 3.3.3 Eliminación de Usuario..... 6
 - 3.4 Modulo Pacientes..... 7
 - 3.4.1 Agregar nueva mascota..... 8
 - 3.4.2 Edición de Mascota..... 8
 - 3.4.3 Eliminar mascota..... 9
 - 3.5 Módulo de Historial Médico..... 9
 - 3.5.1 Agregar Historial Médico..... 10
 - 3.5.2 Edición Historial Médico..... 10
 - 3.5.3 Eliminar Historial Médico..... 11
 - 3.6 Modulo Citas Medicas..... 11
 - 3.6.1 Agendar nueva cita..... 12
 - 3.6.2 Visualizar Cita..... 12
 - 3.6.3 Edición de Citas..... 13
 - 3.6.4 Eliminación de Citas..... 13
 - 3.7 Módulo de Notas..... 14
 - 3.7.1 Nueva Nota..... 14
 - 3.7.2 Edición de Notas..... 15
 - 3.7.3 Eliminación de Nota..... 15
 - 3.8 Módulo de Tratamientos..... 16

Manual de usuario | IEEE 1063 – 2001 2

MEDICAL PETS

- 3.8.1 Nuevo tratamiento..... 16
- 3.8.2 Edición de Tratamiento..... 17
- 3.8.3 Eliminación de Tratamiento..... 17
- 3.9 Módulo de Predicción con IA..... 18
- 3.10 Filtro de Búsqueda..... 20
- 3.11 Cerrar Sesión..... 20
- 4 MENSAJES DE ERROR Y RESOLUCIÓN DE PROBLEMAS..... 21
- 5 GLOSARIO..... 21
- 6 REFERENCIAS..... 22
- 7 CARACTERÍSTICAS DE NAVEGACIÓN..... 22

Manual de usuario | IEEE 1063 – 2001 3

MEDICAL PETS

1 INTRODUCCIÓN

Con el constante avance de la tecnología muchas áreas de la ciencia han presenciado avances increíblemente exponenciales que las ha ayudado a crecer e irse mejorando poco a poco en los últimos años. Uno de esas áreas en la cual nos hemos enfocado es en el área de la salud, específicamente enfocado en la salud de las mascotas, por lo cual se ha creado la aplicación 'Medical Pets' para la creación de historiales médicos para las mascotas.

Con la aplicación web tanto dueños como médicos veterinarios tendrán un fácil acceso para ver el estado actual y evolutivo de los más peludos de la casa.

2 CONCEPTO DE LAS OPERACIONES

Los requerimientos mínimos para que la aplicación BONES 3D funcione correctamente, son los siguientes:

- Teléfono o computadora inteligente.
- Poseer un navegador web
- Procesador: 2Vcpu.
- Memoria RAM: 8GB.
- Almacenamiento: 50 GB.
- Acceso al enlace de la aplicación web.

3 PROCEDIMIENTOS

3.1 Acceso a la aplicación web

Para poder acceder a la aplicación web, lo primero que tendremos que hacer será acceder al enlace de la misma, una vez ingresado nos aparecerá la ventana del Login en donde tendremos que ingresar nuestro usuario y contraseña.

Manual de usuario | IEEE 1063 – 2001 3

MEDICAL PETS



3.2 Ventana Principal

Una vez iniciada la sesión en "Medical Pets" se nos mostrara la ventana principal con los respectivos módulos para poder navegar (los módulos que se muestran pueden variar dependiendo del rol del usuario)



3.3 Modulo Usuario

Al momento de darle click en el módulo de usuarios si el usuario con el cual se accedió tiene los permisos necesarios se podrán visualizar todos los usuarios que se encuentran registrados en la aplicación y sus mascotas. En el caso que no se tengan los respectivos permisos el usuario lo único que podrá ver será su usuario con sus datos y su

Manual de usuario | IEEE 1063 – 2001 4

MEDICAL PETS

respectiva mascota (en caso de tener varias se mostrara la primera que se asoció con este usuario).



3.3.1 Agregar nuevo usuario

Si queremos crear un nuevo usuario lo que tenemos que hacer es darle click EN EL símbolo '+' que se encuentra en la barra de navegación izquierda, esto abrirá un pequeño formulario donde tendremos que llenar los campos correspondientes, una vez llenados y confirmada la información del nuestro nuevo usuario lo que tendremos que hacer es darle click en guardar lo que provocara que automáticamente se muestre el nuevo usuario registrado en pantalla.

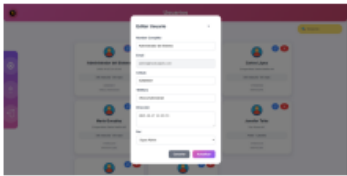


Manual de usuario | IEEE 1063 – 2001 5

MEDICAL PETS

3.3.2 Edición de usuario

Si necesitamos editar alguno de los datos del usuario ingresado lo que tendremos que hacer es localizar al usuario al cual queremos modificar sus datos y darle click en el botón de editar, esto provocara que se abra el formulario con los datos actuales del usuario los cuales podrán ser modificados, una vez cambiados y revisados podrán darle click en el botón de 'Actualizar' para que se actualicen los nuevos datos.




3.3.3 Eliminación de Usuario

En el caso que nosotros necesitemos de eliminar un usuario, lo que tendremos que hacer es localizar al usuario que se desea eliminar y darle click en la 'X', esto provocara que se muestre un mensaje en pantalla que nos preguntara si estamos seguros de eliminar el usuario seleccionado, si damos click en 'Aceptar' lo que sucederá es que el usuario eliminado ya no se mostrara mas en la pantalla de Usuarios.


Manual de usuario | IEEE 1063 – 2001 6

MEDICAL PETS



3.4 Módulo Pacientes

En este módulo si contamos con los permisos necesarios podremos ver todas las mascotas que se han registrado en la aplicación. En caso de ingresar con el rol de usuario nuestra vista en este módulo se limitará únicamente a la mascota o mascotas las cuales tengamos asociadas a nuestro usuario.




Manual de usuario | IEEE 1063 – 2001 7

MEDICAL PETS

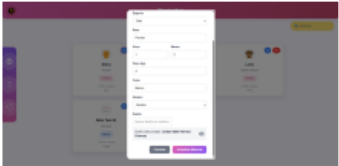
3.4.1 Agregar nueva mascota

Muy parecido al método de agregar un nuevo usuario, las nuevas mascotas se agregarán igualmente por medio de un formulario, pero con la novedad que en este mismo existe un campo en donde por medio de un filtro de búsqueda se puede buscar al dueño de la mascota que se esta registrando. Una vez se comprueben que están todos los datos en el formulario se procederá a dar click en el botón 'Guardar Mascota'.



3.4.2 Edición de Mascota

Para poder editar una mascota se le tendrá que dar click en el botón de edición lo que desplegara un formulario con los datos de la mascota seleccionado, una vez corregido los datos se le tiene que dar click en 'Actualizar Mascota' para guardar los cambios hechos.



Manual de usuario | IEEE 1063 – 2001 8

MEDICAL PETS


3.4.3 Eliminar mascota

En el caso que nosotros necesitemos de eliminar una mascota, lo que tendremos que hacer es localizar a la mascota que se desea eliminar y darle click en la 'X', esto provocara que se muestre un mensaje en pantalla que nos preguntara si estamos seguros de eliminar la mascota seleccionado, si damos click en 'Aceptar' lo que sucederá es que la mascota eliminada ya no se mostrara más en la pantalla de Pacientes.



3.5 Módulo de Historial Médico

En este módulo si contamos con los permisos necesarios podremos ver todos los historiales médicos que se han registrado en la aplicación.




Manual de usuario | IEEE 1063 – 2001 9

MEDICAL PETS


3.5.1 Agregar Historial Médico

Bastante similar al método de agregar una nueva mascota, los nuevos historiales se agregarán igualmente por medio de un formulario, en donde se tendrá que llenar los respectivos campos de acuerdo con los datos obtenidos de la mascota seleccionada, una vez llenado los datos se procederá a dar click en 'Guardar Historial'.



3.5.2 Edición Historial Médico

Para poder editar un historial médico se tendrá que dar click en el botón de edición lo que desplegara un formulario con los datos de la mascota seleccionado, una vez corregido los datos se le tiene que dar click en 'Actualizar Historial' para guardar los cambios realizados.



Manual de usuario | IEEE 1063 – 2001 10

3.5.3 Eliminar Historial Médico

En el caso que nosotros necesitemos de eliminar un historial médico, lo que tendremos que hacer es localizar a el historial correcto que se desea eliminar y darle click en la 'X', esto provocara que se muestre un mensaje en pantalla que nos preguntara si estamos seguros de eliminar el historial seleccionado, si damos click en 'Aceptar' lo que sucederá es que el historial eliminado ya no se mostrara más en la pantalla de Historial Médico.



3.6 Módulo Citas Médicas

En este modulo si se dispone de los permisos correspondientes se podrá agendar una nueva cita médica, en caso de que el rol sea de un usuario simplemente se mostrara la cita que se tiene agendada (en caso de tener una).



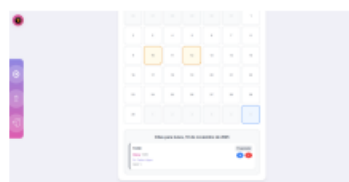
3.6.1 Agendar nueva cita

Para poder agendar una nueva cita se tienen disponibles dos opciones, se le puede dar al botón '+' o simplemente seleccionar la fecha que se quiere agendar, cualquiera de estas opciones desplegara un pequeño formulario con los datos requeridos para el agendamento de la cita. Una vez llenado los respectivos campos se procedera en dar click en 'Guardar Cita' para que la cita quede registrada.



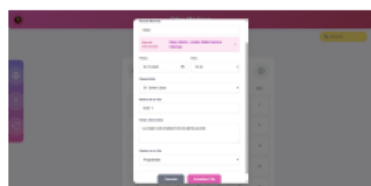
3.6.2 Visualizar Cita

Para poder visualizar la cita medica bastara con dar click en el día exacto del calendario que se muestra en pantalla, lo que desplegara una lista con todas las citas que se tengan registradas para ese día.



3.6.3 Edición de Citas

Para poder editar una cita médica se tendrá que dar click en el botón de edición lo que desplegara un formulario con los datos correspondientes a la cita seleccionada, una vez corregido los datos se le tiene que dar click en 'Actualizar Cita' para guardar los cambios realizados.



3.6.4 Eliminación de Citas

En el caso que nosotros necesitemos de eliminar una cita médica, lo que tendremos que hacer es localizar la cita que se desea eliminar y darle click en la 'X', esto provocara que se muestre un mensaje en pantalla que nos preguntara si estamos seguros de eliminar la cita seleccionada, si damos click en 'Aceptar' lo que sucederá es que la cita eliminada ya no se mostrara más en la pantalla de Citas Médicas.



3.7 Módulo de Notas

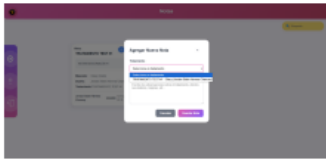
En este módulo, el usuario (dueño de mascota) podrá crear notas relacionadas al tipo de tratamiento que está siguiendo su mascota. Estas servirán para mantener al día al médico Veterinario sobre la evolución de su mascota con respecto al mal a tratar.



3.7.1 Nueva Nota

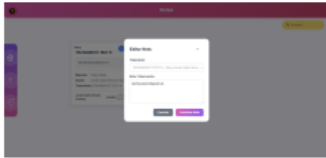
Para la creación de una nueva nota es aun mucho más fácil que los casos anteriores, pues aquí el usuario simplemente tendrá que darle click en el botón '+', seleccionar el tratamiento de su mascota y escribir la observación. Una vez realizado esto tendrá que darle click en el botón 'Guardar Nota'.

MEDICAL PETS



3.7.2 Edición de Notas

La edición podrá realizarla el usuario dando click en el botón de editar de la nota de su mascota. Una vez tenga los datos modificados deberá de darle click en el botón 'Actualizar Nota' para guardar dichos cambios.




3.7.3 Eliminación de Nota

En el caso que el usuario quiera eliminar una nota, lo que tiene que hacer es localizar la nota que se desea eliminar y darle click en la 'X', esto provocara que se muestre un mensaje en pantalla que nos preguntara si estamos seguros de eliminar la nota seleccionada, si damos click en 'Aceptar' lo que sucederá es que la nota eliminada ya no se mostrara más en la pantalla de Notas.


Manual de usuario | IEEE 1063 – 2001 15

MEDICAL PETS



3.8 Módulo de Tratamientos

En este módulo si se tienen los permisos necesarios se permitirá especificar el tipo de tratamiento que deberá de seguir la mascota (paciente).




3.8.1 Nuevo tratamiento

Para poder crear un nuevo tratamiento lo que se tienen que hacer es dar click al botón '+' lo que desplegara un pequeño formulario con los datos requeridos para el registrar un nuevo tratamiento. Una vez llenado los respectivos campos se procederá en dar click en 'Guardar Tratamiento para que la cita quede registrada.


Manual de usuario | IEEE 1063 – 2001 16

MEDICAL PETS



3.8.2 Edición de Tratamiento

La edición podrá realizarla el usuario dando click en el botón de editar del tratamiento de su mascota. Una vez tenga los datos modificados deberá de darle click en el botón 'Actualizar Tratamiento para guardar dichos cambios.

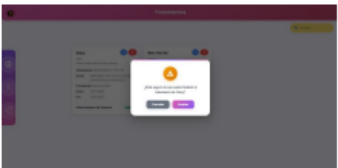


3.8.3 Eliminación de Tratamiento

En el caso que se requiera eliminar un tratamiento, lo que se tiene que hacer es localizar el tratamiento que se desea eliminar y darle click en la 'X', esto provocara que se muestre un mensaje en pantalla que nos preguntara si estamos seguros de eliminar el tratamiento seleccionado, si damos click en 'Aceptar' lo que sucederá es que el tratamiento quedara eliminado y ya no se mostrara más en la pantalla de Tratamientos.


Manual de usuario | IEEE 1063 – 2001 17

MEDICAL PETS



3.9 Módulo de Predicción con IA


Por parte de la innovación se creó el módulo de predicción con IA, este módulo existe con el único fin de sugerirle al veterinario de posibles enfermedades que puede tener la mascota a analizar.




Para poder hacer uso de dicho modulo de predicción, lo primero que tendremos que hacer es utilizar el filtro de búsqueda ubicado en la parte superior derecha, en este filtro lo que haremos será ingresar el nombre del dueño de la mascota, una vez lo identifiquemos tendremos que darle click al mismo para poder avanzar.

Manual de usuario | IEEE 1063 – 2001 18

MEDICAL PETS



A continuación, se mostrarán todas las mascotas que el usuario tenga registradas, de las cuales seleccionaremos únicamente la que nosotras precisáramos hacer la predicción, una vez seleccionada dicho paciente procederemos a darle click en el botón de "Hacer predicción".



Después de un rápido análisis se nos mostrarán por pantalla las 3 posibles enfermedades que pueda tener la mascota seguido de donde se están sacando la base para dicho análisis y una rápida recomendación de que se puede hacer en este tipo de situaciones.

Manual de usuario | IEEE 1063 – 2001 19

MEDICAL PETS



3.10 Filtro de Búsqueda

El filtro de búsqueda se encuentra presente en los módulos de 'Usuarios', 'Pacientes', 'Historial Médico', 'Citas Médicas', 'Notas', 'Tratamientos'. Tiene con fin facilitar la búsqueda de información por parte de los usuarios.




3.11 Cerrar Sesión

Este botón nos acompañará en la navegación de toda la aplicación y como su nombre nos indica nos permitirá cerrar la sesión de nuestro usuario. Cuando le demos click al icono correspondiente este nos mostrará por pantalla un mensaje para estar seguros que queremos culminar nuestra sesión y regresar a la ventana del Login.

Manual de usuario | IEEE 1063 – 2001 20


G

MEDICAL PETS



4 MENSAJES DE ERROR Y RESOLUCIÓN DE PROBLEMAS

Todos los errores que pueden producirse dentro de 'Medical Pets', son presentados mediante un cuadro de dialogo, que describe a detalle la causa del error, haciendo fácil su comprensión e implícitamente dándole al usuario la manera de solucionarlo.



5 GLOSARIO

- **Modulo:** Botón que el usuario puede utilizar para navegar en las diferentes páginas que dispone la aplicación.
- **Tratamiento:** Conjunto de medios y acciones que se aplican para curar o aliviar un malestar.
- **Predicción con IA:** Ayuda mediante el uso de la inteligencia arterial para una predicción de posibles enfermedades futuras que pueda tener el paciente.





Manual de usuario | IEEE 1063 – 2001 21

MEDICAL PETS

6 REFERENCIAS

- IEEE (2001). 1063-2001 - IEEE Standard for Software User Documentation [Internet]. Recuperado de: <https://ieeexplore.ieee.org/document/974401>


7 CARACTERÍSTICAS DE NAVEGACIÓN

Botón	Funcionalidad
	Oculta y despliega la barra de navegación.
	Botón de agregar.
	Regresa a la ventana principal.
	Botón de cierre de sesión.

Manual de usuario | IEEE 1063 – 2001 22

Anexo 8: Manual Técnico

30 DE NOVIEMBRE DE 2025



MANUAL TÉCNICO
IEEE 1063-2001

JORNDAN HERRERA – JENNIFER PAREDES
PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR
Santo Domingo

MEDICAL PETS

REGISTRO DE CAMBIOS

FECHA	USUARIO	VERSIÓN	ACCIONES
30-nov.-25	JORDAN	1.0	CREACIÓN

Manual TÉCNICO | IEEE 1063 – 2001

1

MEDICAL PETS

Tabla de Contenidos

- 1 INTRODUCCIÓN..... 3
- 2 CONCEPTO DE OPERACIONES 3
 - 2.1 Requerimientos del dispositivo 3
 - 2.2 Requerimientos del sistema..... 3
- 3 Procedimientos..... 4
 - 3.1 Clonar repositorios GitHub 4
 - 3.2 Archivos del Backend..... 4
 - 3.2.1 Archivo de Vistas (routers) 4
 - 3.2.2 Archivo de Serializadores (schemas)..... 5
 - 3.2.3 Archivo de Rutas (Main) 5
 - 3.3 Archivos del Frontend..... 6
 - 3.3.1 Carpeta de Componentes..... 6
 - 3.3.2 Carpeta de Paginas (views)..... 6
 - 3.3.3 Carpeta de Rutas..... 7
 - 3.3.4 Carpeta de Contextos (stores)..... 7
 - 4.1 Base de Datos MySQL..... 11
- 5 GLOSARIO..... 12
- 6 REFERENCIAS..... 13

Manual TÉCNICO | IEEE 1063 – 2001

2

MEDICAL PETS

1 INTRODUCCIÓN

El presente manual técnico proporciona una visión detallada de la estructura de la aplicación web, la utilización de sus frameworks y las tecnologías que se usaron. Lo primero que se mostrará será el proceso de clonación mediante el repositorio oficial alojado en GitHub. Después, se dará la explicación de los archivos del backend y el frontend en la aplicación web, así como la estructura de sus archivos y componentes en la aplicación.

Al ir mencionando la distribución de carpetas y sus funcionalidades, se describirá las rutas que se utilizan para la creación de los datos, la edición o modificación de estos mismo, conexiones a la base de datos, eliminación de los datos (pero solo desde la vista del usuario, de manera interna los datos se mantienen), llamado de los datos para su visualización, entre otros factores. Este manual será de suma importancia para lograr tener una comprensión clara de la arquitectura del software, facilitando así su mantenimiento, escalabilidad y colaboración en futuras actualizaciones o mejoras.

2 CONCEPTO DE OPERACIONES

A continuación, se detallará los requerimientos mínimos para el correcto uso y despliegue de la aplicación web Medical Pets:

- 2.1 Requerimientos del dispositivo**
 - Teléfono o computadora inteligente.
 - Poseer un navegador web
- 2.2 Requerimientos del sistema**
 - **Procesador:** 2Vcpu.
 - **Memoria RAM:** 8GB.
 - **Almacenamiento:** 50 GB.

Manual TÉCNICO | IEEE 1063 – 2001

3


MEDICAL PETS

- Sistema operativo: Windows 11, Ubuntu 22.04

3 Procedimientos

3.1 Clonar repositorios GitHub

Para poder acceder al repositorio de la aplicación se tendrá que acceder al siguiente enlace <https://github.com/MLPShurA/MedicalPetsClean>, en este repositorio se podrá encontrar la aplicación actualizada con su respectivo archivo README.md y las carpetas del proyecto divididas respectivamente en frontend y backend.



Para poder obtener este proyecto en su totalidad, haremos uso de los comandos de GitHub para poder descargar el repositorio. Utilizando el comando "git clone <https://github.com/MLPShurA/MedicalPetsClean>"


3.2 Archivos del Backend

3.2.1 Archivo de Vistas (routers)

Mediante este archivo es donde se cumplen las funciones de vistas/controladores que forman parte de la arquitectura de la aplicación, estos archivos contienen la lógica requerida para recibir y responder las peticiones HTTP REST.


Manual TÉCNICO | IEEE 1063 – 2001 4

MEDICAL PETS



3.2.2 Archivo de Serializadores (schemas)

Los archivos en esta carpeta utilizan la librería Pydantic para la validación y estructuración de los datos. Mediante estos archivos es posible responder a las peticiones HTTP con el respectivo formato JSON requerido junto con la validación de la información entrante antes de procesarla.




3.2.3 Archivo de Rutas (Main)

En esta aplicación web es mediante el archivo main.py la que se utiliza como punto de entrada de la aplicación. Aquí es donde se centralizan y registran todas las rutas definidas en la carpeta routers.

Manual TÉCNICO | IEEE 1063 – 2001 5


MEDICAL PETS



3.3 Archivos del Frontend

3.3.1 Carpeta de Componentes

En la carpeta de componentes se estructuran los elementos que se reutilizan para el correcto funcionamiento de la aplicación, en estos mismos se encuentran elementos como pueden ser tarjetas, modales o elementos de los formularios.




3.3.2 Carpeta de Páginas (views)

Aquí es donde podemos localizar las principales vistas a las cuales podrá acceder el usuario. Cada archivo que se encuentra en esta carpeta representa la navegación que podrá realizar el usuario de acuerdo a su rol y permisos concedidos a los mismos.


Manual TÉCNICO | IEEE 1063 – 2001 6

MEDICAL PETS



3.3.3 Carpeta de Rutas

Heredando la configuración vue-router, aquí es donde se define que componente o página se debe mostrar en cada URL del navegador.



3.3.4 Carpeta de Contextos (stores)

Aquí el archivo que está en esta carpeta utiliza la librería de Pinia para poder realizar una gestión del estado global de la aplicación. Este permite compartir datos entre los diferentes componentes sin necesidad de pasar props manualmente.

Manual TÉCNICO | IEEE 1063 – 2001 7



4 Módulo de predicción con Inteligencia Artificial.

En el módulo de predicción con IA se implementa un sistema avanzado de Aprendizaje Supervisado la cual utiliza una arquitectura basada en modelos para conseguir predecir las posibles enfermedades de las mascotas (pacientes) basándose principalmente en su raza, edad e historial médico.

4.1 Arquitectura

Toda la lógica que se implementó en la inteligencia artificial se encuentra en la siguiente ruta 'backend/ml_logic' en donde separamos el proceso de entrenamiento con la llamada del modelo completado, con el fin de optimizar el rendimiento y no tener que realizar todo el procedimiento por cualquier tipo de falla inesperada o interrupción del sistema en producción.

4.1.1 Entrenamiento (train_model.py)

En el archivo 'train_model.py' es donde se encuentra todo el script con la lógica del modelo para su entrenamiento, es con el fin que después de todo el proceso se genere un nuevo archivo 'disease_prediction_model.pkl' el cual se puede decir es el cerebro encargado de realizar la predicción. Para la generación de este cerebro se utilizaron las

siguientes lógicas para el filtrado, optimización y precisión relacionado a la predicción de la enfermedad de la mascota:

- **Regresión logística:** Nos provee de una base probabilística bastante sólida para empezar a clasificar y descartar los datos en base a la raza y los historiales médicos de las mascotas.
- **Random Forest:** Este algoritmo tiene la capacidad de capturar relaciones complejas y no lineales entre síntomas y diagnósticos que se subdividen entre los 300 árboles de decisión. Se utilizó 'min_samples_leaf=3' para evitar el sobreajuste (overfitting).
- **SGDClassifier (SVM):** Se utiliza un modelo de 'Máquina de Vectores de Soporte' junto con el descenso de gradiente estocástico el cual en base a los dos filtros anteriores actúa como un clasificador de alta confianza para los patrones encontrados.

Una vez realizado todo el entrenamiento utilizamos la librería de Joblib para la generación del archivo 'disease_prediction_model.pkl', el cual en palabras simples se puede decir que es el cerebro encargado de realizar las predicciones. Esta herramienta nos facilita bastante el trabajo ya no solo de la predicción sino también del entrenamiento pues bastara con solo ejecutar el código una vez para tener una IA la cual solo necesitara ser llamada como lo arías con cualquier API.



4.1.2 Predicción (predictor.py)

En este archivo, empieza cargando el modelo previamente entrenado 'disease_prediction_model.pkl' para la generación de predicciones en tiempo real en donde realiza un post-procesamiento para proporcionarle al usuario las 3 posibles enfermedades que pueda tener la mascota acompañada de la probabilidad de dichas enfermedades. Al momento de reutilizar 'VeterinaryPreprocessor' se garantiza que los datos de entrada desde la web reciban el mismo tratamiento de limpieza y formateo que los datos de entrenamiento, esto con el fin, que la predicción se realice de la manera más limpia posible para evitar predicciones erróneas.



4.2 Integración con el Backend

Ubicado en la ruta 'backend/routers/predictions.py' aquí es donde se definen los endpoints REST que realizan la comunicación entre el Frontend y el cerebro de la IA, para aquello lo que realiza es lo siguiente:

- Recibe el ID de la mascota.
- Realiza una consulta a la base de datos para obtener el historial clínico, raza, peso y edad.
- Envía los datos limpios al módulo 'predictor.py'.
- Al final retorna un objeto JSON con el diagnóstico principal junto con el porcentaje de confianza y una pequeña lista de las otras posibilidades.

4.3 Integración con el Frontend


La integración del frontend sigue el mismo patrón que se ha seguido durante toda la aplicación web, incluido el respectivo response para el uso de la misma en diferentes dispositivos. Tiene la capacidad de realizar búsqueda inteligente de los dueños de las mascotas, selección de la o las mascotas del dueño buscado, puede presentar resultados de las posibles enfermedades que pueda padecer la mascota en base a los historiales médicos de las mismas.

5 Acceso a los datos


5.1 Base de Datos MySQL

La persistencia de los datos de la aplicación se maneja por medio de una base de datos relacional MySQL. Esta misma se encuentra alojada en un contenedor de Docker en un servidor de producción.

MEDICAL PETS



Utilizamos el IDE de JupyterLab para realizar la conexión de manera remota y poder disponer de una interfaz gráfica para la gestión de la misma



6 GLOSARIO

- **Aplicación Web:** La aplicación web permite a los usuarios obtener una experiencia interactiva e innovadora, fusionando una interfaz amigable para los usuarios.
- **Backend:** Es la parte de la aplicación que se encarga de la parte lógica de la misma, entre algunos de los ejemplos se puede mencionar la base de datos y la seguridad.
- **Frontend:** Es toda la parte visual de la aplicación que es apreciada por los usuarios, estos mismo pueden interactuar con las funciones que se muestran.

Manual TÉCNICO | IEEE 1063 – 2001 12

MEDICAL PETS

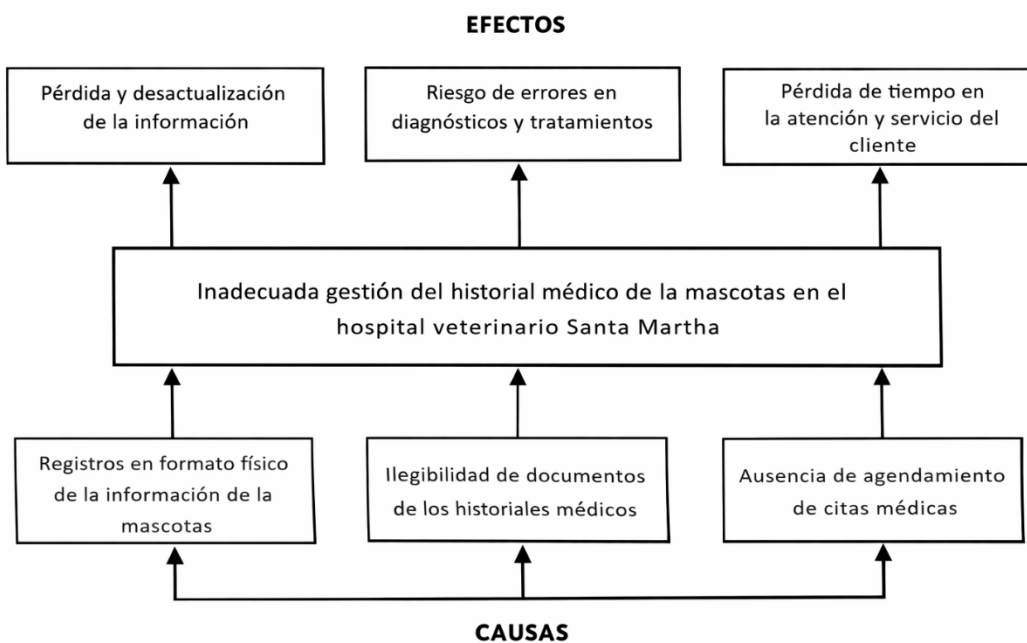
- **Repositorio:** Este es el almacén centralizado que es capaz de guardar y organizar todo el código fuente de la aplicación.
- **Librería:** Se la puede denominar como una colección de código previamente escrito que proporciona funciones y herramientas reutilizables para el desarrollo de software.

7 REFERENCIAS

- IEEE (2001). 1063-2001 - *IEEE Standard for Software User Documentation* [Internet]. Recuperado de: <https://ieeexplore.ieee.org/document/974401>

Manual TÉCNICO | IEEE 1063 – 2001 13

Anexo 9: Árbol del problema



Anexo 10: Evidencia de la reunión de planificación con el Doctor Propietario

<p>Reunión de planificación con el doctor propietario</p> <p>Pregunta 1: ¿Cuál es el procedimiento para gestión de historiales médicos de las mascotas?</p> <p>Respuesta: Generalmente los clientes, es decir los dueños de los pacientes tienen un carnet donde se registran datos generales, vacunas que se le suministran a cada mascota, las visitas médicas que han tenido al hospital, registros de desparasitación, tratamientos y cirugías, adicional nosotros llevamos un registro computacional de las visitas de los animalitos al hospital, el motivo y medicamento recetado.</p> <p>Pregunta 2: ¿Considera que digitalizar los historiales médicos de las mascotas mejora el servicio del hospital?</p> <p>Respuesta: Sí, de hecho, ya he pensado en implementar eso en el hospital, probamos con una herramienta (nos indicó la herramienta llamada Doctor Vet), pero la interfaz es un poco difícil de entender, además no indaga bien en la gestión de los historiales médicos, abarca muchos temas incluso contabilidad. Así que implementar una herramienta para la gestión específicamente de historiales médicos de mis pacientes sería beneficioso para mejorar el servicio de mi hospital.</p> <p>Pregunta 3: ¿Con qué frecuencia se actualizan los datos de los pacientes en la veterinaria?</p> <p>Respuesta: Dependiendo del tiempo de revisión de cada animalito, como mencionaba registramos las visitas al hospital y el motivo. Mi hospital no es una veterinaria común, aquí realizamos exámenes y cirugías de todo tipo y trabajamos las 24 horas del día, hemos tenido emergencias de madrugada de mascotas que no son pacientes regularmente del hospital entonces registramos esa primera o última fecha y el motivo. Es decir, la actualización de los datos depende de cada cliente básicamente.</p>	<p>Pregunta 4: En el proceso de asignación de tratamientos ¿Cuál es el método para el control de los mismos?</p> <p>Respuesta: Por lo general no damos un tratamiento sin antes comprobar que lo que sospechamos sea cierto, entonces cuando un animal llega con alguna sintomatología descartamos o comprobamos enfermedades con exámenes. Después el doctor revisa los resultados y le explica al dueño el tratamiento a seguir, como suministrarlo y fecha de la siguiente revisión.</p> <p>Pregunta 5: ¿Le gustaría tener acceso a posibles reacciones de medicamentos de los tratamientos asignados a las mascotas?</p> <p>Respuesta: Sí claro, me parece importante tener acceso a eso, más que todo porque existen dueños de mascotas que ven algunas de estas reacciones y olvidan avisarle al médico que le asignó esa medicina y optan por cambiar de tratamiento incluso suspenderlo. Además, hay clientes que al inicio del tratamiento ya ven mejoras y dejan de suministrar la medicina argumentando que el animalito "ya está bien", eso a lo largo podría traer consecuencias.</p> <p>Pregunta 6: ¿Considera que hacer una relación entre síntomas y enfermedades podría ayudar a posibles diagnósticos tempranos de la mascota?</p> <p>Respuesta: He escuchado de eso en la medicina de seres humanos y si he pensado que sería genial para el mundo veterinario, sin embargo, no he indagado mucho en eso. Considero que podría ser interesante y de mucha ayuda para mi equipo médico, incluso en algún caso en que no se encuentre el doctor de cabecera de la mascota.</p> <p>Pregunta 7: ¿Cuál es su opinión respecto a la automatización en el diagnóstico temprano de la mascota?</p>
<p>Respuesta: Me parece interesante y de mucha ayuda si se logra implementar bien, de igual manera considero que puede ser un poco complicado que los médicos veterinarios acoplen este tipo de tecnologías en sus rutinas diarias porque los animalitos no explican sus síntomas, se deducen.</p> <p>Pregunta 8: ¿El hospital cuenta con un proceso para predicción de enfermedades?</p> <p>Respuesta: No, hubo casos de investigación de estudiantes para eso, pero no se logró implementar, pero me gustaría contar con tecnologías que ayuden a mejorar la calidad de servicio de mi hospital.</p> <p>Pregunta 9: ¿Considera que el manejo de aplicaciones web es una alternativa más fácil de acceder a la información de una mascota?</p> <p>Respuesta: No sabría si decir más fácil, porque nuevas implementaciones traen cambios y mayor aun si es tecnológicamente, pero si considero que la información estaría más ordenada y al alcance de mis clientes de manera más rápida.</p> <p>Pregunta 10: ¿Cómo se resguarda la información de los historiales médicos en el hospital veterinario?</p> <p>Respuesta: La información médica como tratamientos, vacunas, cirugías la tienen los mismos clientes, las citas médicas al pre agendamos es decir ponemos una fecha establecida, pero si el dueño de la mascota no reporta nada unos días antes pues se cancela y se deja el espacio disponible a otra mascota. Los tratamientos están en manos de los dueños, lamentablemente no tenemos un control de la evolución.</p> <p>Pregunta 11: ¿Está familiarizado con el uso de tecnologías de computación en la nube?</p>	<p>Respuesta: Si he leído un poco sobre, de hecho, utilizamos Google drive para el almacenamiento de alguna información que queremos compartir con el equipo médico del hospital. Sin embargo, no es mucho mi conocimiento sobre el tema.</p> <p>Pregunta 12: ¿Cuál es el procedimiento para agendar citas en el hospital?</p> <p>Respuesta: El doctor que atiende a la mascota designa una fecha próxima, la secretaria se encarga de pre agendar esa cita, pero si el paciente no se comunica unos días antes se cancela la cita y hay disponibilidad para atender a cualquier otra mascotita, se hace de forma manual este proceso.</p> <p>Pregunta 13: ¿Considera importante agilizar el proceso de registro de pacientes?</p> <p>Respuesta: Sí, me parece una excelente idea para administrar los pacientes que asisten de manera regular y los nuevos. Sería una buena gestión del servicio y eficiencia del hospital.</p> <p>Análisis e Interpretación de la entrevista al gerente propietario</p> <p>En base a la entrevista realizada al gerente propietario del Hospital Veterinario Santa Martha, se observó que existen limitantes importantes en la gestión de historiales médicos de las mascotas, ya que no cuentan con un sistema para esto y lo hacen de manera manual. Además, el dueño de la mascota es quien tiene acceso a esta información de manera independiente obteniendo como resultado el no gestionar los tratamientos de los animales, vacunas puestas, citas médicas siguientes incluso exámenes que se la han realizado. En casos de emergencias es bastante complicado poder asignar un medicamento debido a que no se tiene la información completa del paciente.</p> <p>El gerente mostró una actitud positiva e entusiasta hacia la implementación de las tecnologías propuestas, además de mencionar su disponibilidad para el proceso de</p>



Link de video de reunión de planificación:

[VIDEO DE REUNIÓN DE PLANIFICACIÓN CON EL DOCTOR.mp4](#)

Anexo 11: Recodificación del instrumento

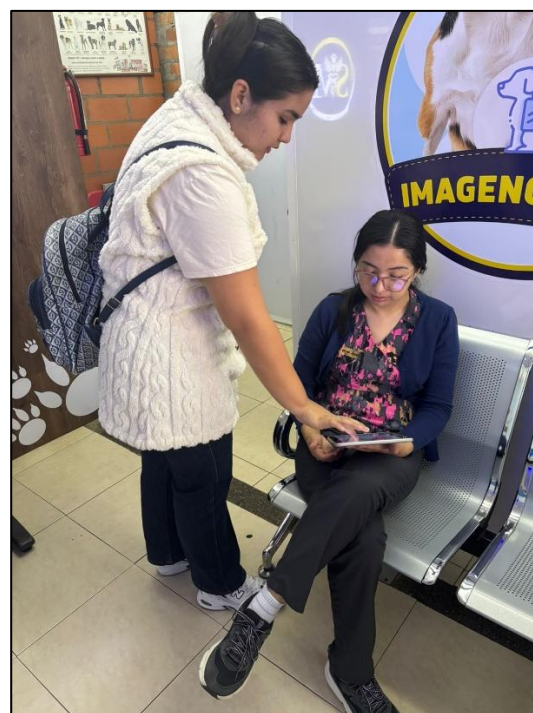
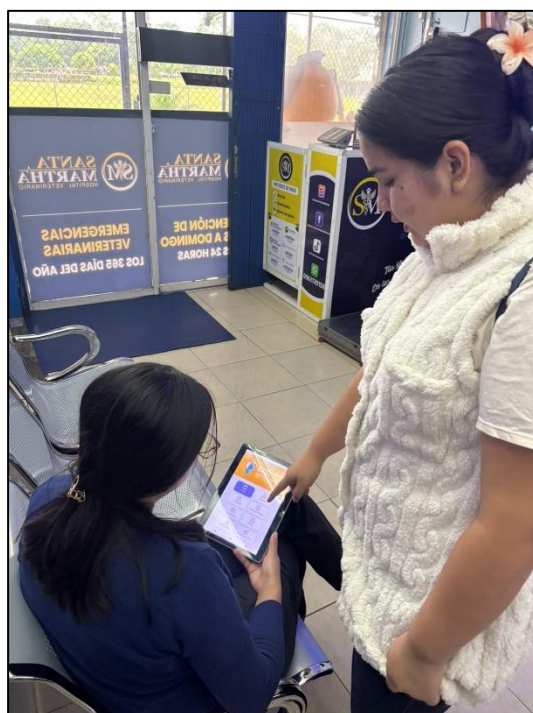
Recodificación escala de Likert	
Valoración	Escala
Familiaridad	
4	Completamente familiarizado
3	Muy familiarizado
2	Algo familiarizado
1	Poco familiarizado
0	Nada familiarizado
Frecuencia	
4	Siempre
3	Casi siempre
2	A veces
1	Casi nunca
0	Nunca
Calidad	
4	Excelente
3	Bueno
2	Regular
1	Deficiente
0	Muy deficiente
Efectividad	
4	Completamente efectivo
3	Altamente efectivo
2	Parcialmente efectivo
1	Poco efectivo
0	Nada efectivo
Importancia	
4	Muy importante
3	Importante

2	Moderadamente importante
1	Poco importante
0	No es importante
Seguridad	
4	Totalmente seguro
3	Seguro
2	Moderadamente seguro
1	Poco seguro
0	Totalmente inseguro
Satisfacción	
4	Totalmente satisfecho
3	Satisfecho
2	Moderadamente satisfecho
1	Poco satisfecho
0	Nada satisfecho
Dificultad	
4	Muy fácil
3	Fácil
2	Normal
1	Difícil
0	Muy difícil
Necesidad	
4	Muy necesario
3	Necesario
2	Medianamente necesario
1	Poco necesario
0	Innecesario
De acuerdo	
4	Totalmente de acuerdo
3	Bastante de acuerdo
2	Neutral
1	Poco de acuerdo
0	Totalmente en desacuerdo

Anexo 12: Evidencia de encuesta dirigida a los clientes



Anexo 13: Evidencia de entrega y explicación de la aplicación web









ENTREGA DE LA APP WEB

 <p>Pontificia Universidad Católica del Ecuador Seréis mis testigos</p>	<p>SANTO DOMINGO</p>	<p>DIRECCIÓN DE INVESTIGACIÓN VINCULACIÓN E INNOVACIÓN</p>
ACTA DE ENTREGA RECEPCIÓN		
<p>En la ciudad de Santo Domingo de la Provincia de Santo Domingo de los Tsáchilas - Ecuador, siendo las 10 horas del día 11 de diciembre del 2023, en las instalaciones del Hospital Veterinario Santa Martha, comparecen:</p>		
<p>El Dr. Patricio Intrago y las señoras Jordan Herrera y Jennifer Paredes, estudiantes de la carrera de Ingeniería de Tecnologías de la Información de la Pontificia Universidad Católica del Ecuador sede Santo Domingo, y una vez culminado el Trabajo de Titulación de Grado "Aplicación Web con Machine Learning para la Gestión de Historial Médico de las mascotas en el Hospital Veterinario Santa Martha en el Cantón Santo Domingo" se procede a la entrega de:</p>		
<ul style="list-style-type: none"> • Aplicación Web con Machine Learning • Manual de usuario • Manual Técnico • Capacitación a los Usuarios 		
<p>Para constancia de lo actuado, en conformidad y aceptación, firman los intervinientes la presente acta de entrega-recepción en dos ejemplares.</p>		
<p>Entrega:</p>		
 _____ Sr. Jordan Herrera Estudiante PUCE-SD	 _____ Srta. Jennifer Paredes Estudiante PUCE-SD	
<p>Recibe:</p>		
 _____ Dr. Patricio Intrago Propietario del Hospital Veterinario Santa Martha		
<p><small>Boleadora, Vía a Clono Km. 2, Código postal: 230201 Teléfono: (099) 903284270 Santo Domingo, Ecuador info@puce.edu.ec</small></p>		
		

Anexo 14: Certificados de participación

 Pontificia Universidad Católica del Ecuador	SANTO DOMINGO
Otorga el siguiente CERTIFICADO	
A: PAREDES ROMERO JENNIFFER TAHIS	
En reconocimiento a su destacada contribución en el ámbito de la Investigación e Innovación, mediante su participación activa con el tema:	
APLICACIÓN WEB CON MACHINE LEARNING PARA LA GESTIÓN DE HISTORIAL MÉDICO DE MASCOTAS EN EL HOSPITAL VETERINARIO SANTA MARTHA EN EL CANTÓN SANTO DOMINGO	
Su compromiso y excelencia representan un ejemplo inspirador para nuestra institución y futuras generaciones de profesionales.	
 FRANKLIN CARRASCO RAMÍREZ <small>COORDINADOR DE LA CARRERA DE SISTEMAS DE INFORMACIÓN</small>	
Mg. Franklin Carrasco Ramírez <small>COORDINADOR DE LA CARRERA DE SISTEMAS DE INFORMACIÓN</small>	
QUITO - AMAZONAS - AMBATO - ESMERALDAS - IBARRA - MANABÍ - SANTO DOMINGO	

 Pontificia Universidad Católica del Ecuador	SANTO DOMINGO
Otorga el siguiente CERTIFICADO	
A: HERRERA CHANCAY JORDAN STALIN	
En reconocimiento a su destacada contribución en el ámbito de la Investigación e Innovación, mediante su participación activa con el tema:	
APLICACIÓN WEB CON MACHINE LEARNING PARA LA GESTIÓN DE HISTORIAL MÉDICO DE MASCOTAS EN EL HOSPITAL VETERINARIO SANTA MARTHA EN EL CANTÓN SANTO DOMINGO	
Su compromiso y excelencia representan un ejemplo inspirador para nuestra institución y futuras generaciones de profesionales.	
 FRANKLIN CARRASCO RAMÍREZ <small>COORDINADOR DE LA CARRERA DE SISTEMAS DE INFORMACIÓN</small>	
Mg. Franklin Carrasco Ramírez <small>COORDINADOR DE LA CARRERA DE SISTEMAS DE INFORMACIÓN</small>	
QUITO - AMAZONAS - AMBATO - ESMERALDAS - IBARRA - MANABÍ - SANTO DOMINGO	

Anexo 15: Informes de Turnitin

Trabajo de titulación Jordan Herrrera-Jennifer Paredes

FINAL v2 .docx

por Jennifer Tahis Paredes Romero

Fecha de entrega: 20-mar-2026 06:12p. m. (UTC-0500)

Identificador de la entrega: 2908518202

Nombre del archivo: Trabajo_de_titulación_Jordan_Herrrera-Jennifer_Paredes_FINAL_v2_.docx (47.83M)

Total de palabras: 29557

Total de caracteres: 169287

Trabajo de titulación Jordan Herrera-Jennifer Paredes FINAL v2 .docx

INFORME DE ORIGINALIDAD

6%

INDICE DE SIMILITUD

5%

FUENTES DE INTERNET

2%

PUBLICACIONES

2%

TRABAJOS DEL
ESTUDIANTE

Jennifer Tahis Paredes Romero

Trabajo de titulación Jordan Herrera-Jennifer Paredes FINAL v2 .docx

TTG - V8
Trabajo Titulación Grado 202501
PUCE SANTO DOMINGO MOODLE

Detalles del documento

Identificador de la entrega

trn:oid:::1:3512632002

181 páginas

Fecha de entrega

20 mar 2026, 5:57 p.m. GMT-5

29.557 palabras

Fecha de descarga

20 mar 2026, 6:21 p.m. GMT-5

169.287 caracteres

Nombre del archivo

Trabajo_de_titulación_Jordan_Herrera-Jennifer_Paredes_FINAL_v2_.docx

Tamaño del archivo

47.8 MB

*% detectado como IA

La detección de IA incluye la posibilidad de que haya falsos positivos. Aunque cierto texto en esta entrega se generó probablemente con IA, los puntajes inferiores al umbral del 20 % no aparecen porque tienen una mayor probabilidad de falsos positivos.

Precaución: Se necesita revisión.

Es esencial comprender los límites de la detección de IA antes de tomar decisiones acerca del trabajo del estudiante. Te alentamos a obtener más información acerca de las funciones de detección de IA de Turnitin antes de usar la herramienta.

Aviso legal

Nuestra evaluación de escritura con IA está diseñada para ayudar a los académicos a identificar texto que podrían haberse preparado mediante una herramienta de IA generativa. Es posible que nuestra evaluación de escritura con IA no siempre sea precisa (existe la posibilidad de que identifique erróneamente redacciones probablemente generadas por humanos como generadas por IA, y redacciones probablemente generadas por IA como generadas por humanos), por lo que no debe usarse como único fundamento para aplicar sanciones a un estudiante. Para determinar si es un caso de deshonestidad académica, se necesita de un escrutinio mayor y el juicio humano, junto con la aplicación de las políticas académicas específicas de la organización.