



PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

Unidad Académica de Formación Técnica y Tecnológica – PUCE TEC

SITIO WEB PARA CENTRO MÉDICO VETERINARIO CLIMOVET

Tecnólogo En Desarrollo De Software

Autor: Diego Sebastian Constante Gamboa

Tutor: Germán Orlando Freire Villacís

Quito, Ecuador

2025

Tabla de contenidos

Lista de tablas.....	3
Lista de figuras	3
Agradecimientos.....	5
Introducción.....	6
Capítulo I: Levantamiento de Requisitos y Diseño del Sistema.....	8
1.1 Levantamiento de Requisitos	8
1.2 Diseño de Sistema	11
Capítulo II: Construcción del Sistema	14
2.1 Comunicación Digital	14
2.2 Diseño de Experiencia de Usuario	14
2.3 Gestión de Bases de Datos	15
2.4 Tecnologías de Desarrollo Web	15
Capítulo III: Pruebas y Estabilización	18
3.1 Tipos de Pruebas Realizadas	18
3.1 Resultad de Prruebas	19
Conclusiones	21
Referencias bibliográficas.....	22
Anexos	23

Lista de Tablas

Figura 1. Climovet	8
Figura 2. Diagrama de Uso – Usuario Final	10
Figura 3. Diagrama de Uso – Personal Veterinario	11
Figura 4. Diagrama de Flujo del Sistema	12
Figura 5. Diagrama de Base de Datos	14
Figura 6. Estructura de Directorios	24
Figura 7. Diagrama Entidad Relacion	25
Figura 8. Diagrama de Flujo del Sistema	26

Lista de figuras

Tabla 1. Estructura	17
Tabla 2. Pruebas Realizadas	18
Tabla 3. Diagrama de Uso – Personal Veterinario	19
Tabla 4. Diagrama de Flujo del Sistema	19
Tabla 5. Diagrama de Base de Datos	20
Tabla 6. Estructura de Directorios	20

DECLARACIÓN y AUTORIZACIÓN

Yo, **Diego Sebastian Constante Gamboa** con C.I. **1726313065** autor del trabajo de titulación intitulado: “**Climovet**”, previa a la obtención del título de **Tecnólogo en desarrollo de software** en la Unidad Académica de Formación Técnica y Tecnológica PUCE TEC:

1.- Declaro tener pleno conocimiento de la obligación que tiene la Pontificia Universidad Católica del Ecuador, de conformidad con el artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de graduación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la Pontificia Universidad Católica del Ecuador a difundir a través de sitio web de la Biblioteca de la PUCE el referido trabajo de titulación, respetando las políticas de propiedad intelectual de Universidad.

Quito, 12 de febrero de 2025



Diego Sebastian Constante Gamboa

C.I. 1726313065

Agradecimientos

Quiero expresar mi más sincero agradecimiento a todas las personas que han hecho posible la realización de este proyecto, dedicada al desarrollo de un sitio web para el Centro Médico de Especialidades Veterinarias “Climovet”.

En primer lugar, agradezco a mi madre, cuyo amor, apoyo incondicional y enseñanzas han sido fundamentales en mi vida y en mi formación profesional. Su dedicación y sacrificio me han inspirado a perseguir mis sueños con determinación.

A mis tutores y profesores, quienes compartieron su conocimiento y experiencia, brindándome orientación y motivación a lo largo de este proceso. Sus valiosos consejos y críticas constructivas han enriquecido mi trabajo y me han permitido crecer tanto académica como personalmente.

A todo el equipo del Centro Médico Veterinarias, por confiar en mí para llevar a cabo este proyecto. Su colaboración y apertura para compartir sus necesidades y expectativas fueron esenciales para el éxito del sitio web.

Introducción

La presente propuesta se centra en el desarrollo de un sitio web destinado a modernizar y aumentar el alcance de la atención que ofrece el centro médico veterinario Climovet, ubicado en el cantón El Chaco, provincia de Napo.

Este proyecto responde a la creciente demanda de servicios veterinarios de calidad en la región y la necesidad de facilitar el acceso a estos servicios a través de medios digitales que permitirán a los usuarios obtener información relevante, así como conocer al equipo médico, estar al tanto de la información de vacunas de sus mascotas por un medio digital y accesible desde cualquier lugar con acceso a internet.

La principal importancia de este proyecto radica en la necesidad de tener una plataforma pensada para transformar la interacción de Climovet con sus clientes, ofreciendo una plataforma intuitiva y accesible desde cualquier dispositivo, ya sea un ordenador, una tableta o teléfono móvil, El sitio web propuesto se distingue por su diseño simple y responsivo, adaptándose a la pantalla del usuario para garantizar una experiencia de navegación agradable.

El problema principal que aborda este proyecto es la falta de un canal de comunicación eficaz entre el centro médico y sus clientes, actualmente la información sobre los servicios y el equipo médico es limitada y dispersa, lo que puede generar confusiones, con la implementación de esta plataforma web, se espera solucionar estos inconvenientes para mostrar información clara, actualizada y de fácil acceso.

Además de las funcionalidades básicas, el sitio web incluirá una sección privada y segura para que los propietarios, puedan acceder a información específica de sus mascotas, mostrando su registro de vacuna e información general.

Se espera que al finalizar el proyecto el centro médico veterinario cuente con una plataforma web funcional y accesible que facilite la interacción entre sus clientes. Los resultados esperados incluyen una mejor manera en la visibilidad y la imagen del centro, una mayor accesibilidad a la información sobre los servicios veterinarios disponibles.



Imagen 1. Climovet

Capítulo I

Levantamiento de Requisitos y Diseño del Sistema

1.1 Levantamiento de Requisitos

1.1.1 Gestión de Usuarios

- **Usuarios Finales (Usuario común)**

Los usuarios finales no necesitan registrarse ni iniciar sesión, tendrán acceso al contenido público del sitio web, donde podrán:

Visualizar los servicios veterinarios ofrecidos de forma general en la vista Inicio, presentada en listas y clasificada en dos grupos, servicios generales y servicios a domicilio.

Visualizar las mascotas que pertenecen a la comunidad Climovet, donde si este usuario es propietario de una mascota o varias listada, podrá acceder colocando su cedula para verificar que es quien dice ser, de lo contrario no podrá acceder.

El propietario de dicha mascota, podrá visualizar información medica de la macota, información general, historial de vacunas.

Visualizar información general del centro médico veterinario, donde podrá conocer donde se ubica, información del personal medio, información de contacto.

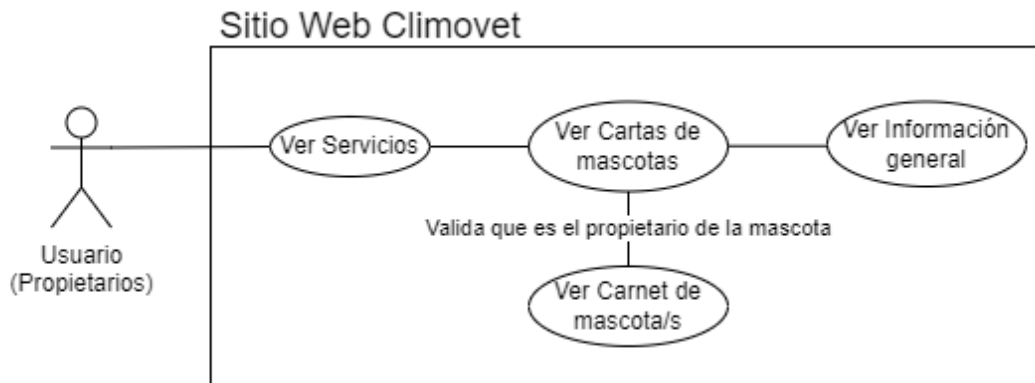


Figura 2. Diagrama de Uso – Usuario Final.

- **Personal Veterinario (Administrador)**

Solo el veterinario responsable del sitio web podrá iniciar sesión como administrador.

El usuario y contraseña del administrador serán creados directamente en la base de datos por el administrador del sistema, sin la opción de crear, editar o eliminar una cuenta desde la plataforma web Climovet.

El personal veterinario puede iniciar sesión y acceder a una vista de administrador donde podrá gestionar a los propietarios y sus mascotas.

Gestionar a los propietarios de las mascotas, podrá añadir y actualizar información general del propietario.

Gestionar las mascotas de un propietario, podrá añadir y actualizar información general de la mascota.

Gestionar los carnets de vacunación de las mascotas, podrá actualizar carnets de mascotas con su historial de vacunas.

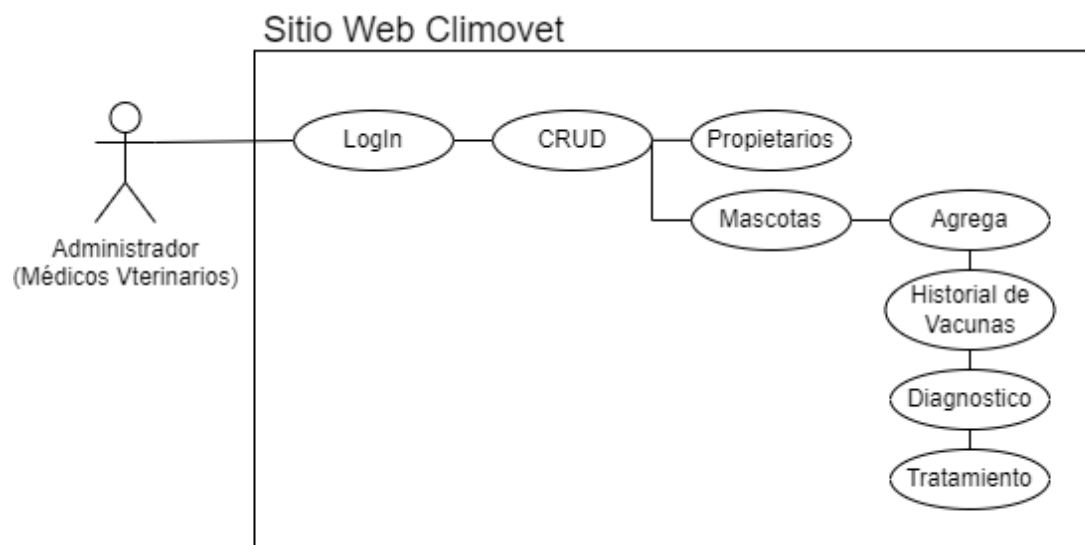


Figura 3. Diagrama de Uso – Personal Veterinario.

1.1.2 Seguridad y Autenticación

El acceso al sistema estará protegido mediante autenticación con credenciales encriptadas y tokens de sesión, para garantizar que solo los médicos veterinarios tengan acceso.

Al no incluir la opción de creación de cuentas desde la plataforma web Climovet, las contraseñas almacenadas en la base de datos, están encriptadas directamente con una función para encriptarlas cada que el administrador del sistema cree una nueva credencial.

1.2 Diseño del Sistema

1.2.1 Arquitectura del sistema

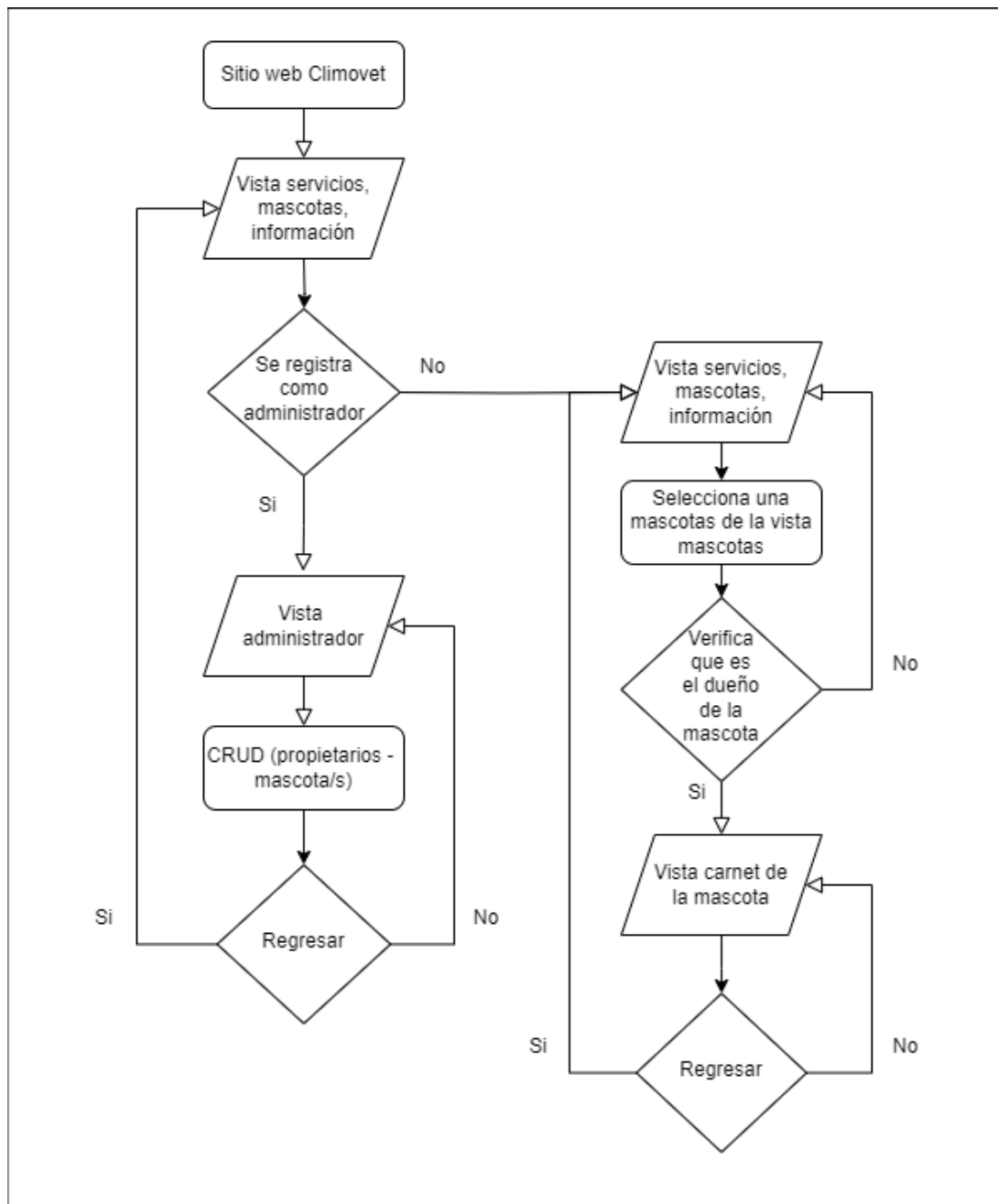


Figura 4. Diagrama de Flujo

- **FrontEnd**

El desarrollo del **frontend** se centró en crear una interfaz de usuario intuitiva y reactiva utilizando **React** y **JSX**.

React permite descomponer la aplicación en componentes reutilizables, lo que facilita el mantenimiento y la escalabilidad del código.

CSS se utilizó para estilizar la interfaz, asegurando que fuera visualmente atractiva y fácil de navegar.

Axios se empleó para realizar solicitudes HTTP al backend, permitiendo la obtención de datos en tiempo real, como la información de mascotas y propietarios.

Vite fue la herramienta de construcción elegida para optimizar el rendimiento y acelerar el proceso de desarrollo.

- **BackEnd**

El **backend** se desarrolló utilizando **Express.js**, un framework de Node.js que permite crear APIs RESTful de manera eficiente.

Sequelize se utilizó como ORM para interactuar con la base de datos PostgreSQL, facilitando la gestión de consultas y gestión y manejo de los datos.

Se implementó **Cors** para el manejo las políticas de seguridad relacionadas con el intercambio de datos entre el frontend y el backend, garantizando una comunicación segura.

- **Base de Datos**

La base de datos se diseñó utilizando **PostgreSQL**, un sistema de base de datos relacional, por su robustez y escalabilidad se optó por el uso de esta base de datos.

1.2.2 Diseño de la base de datos

La base de datos está diseñada en **PostgreSQL**, donde se almacena información de los propietarios, mascotas y su historial de vacunas, se aplicó normalización para evitar redundancia de los datos y mejorar el rendimiento en las consultas.

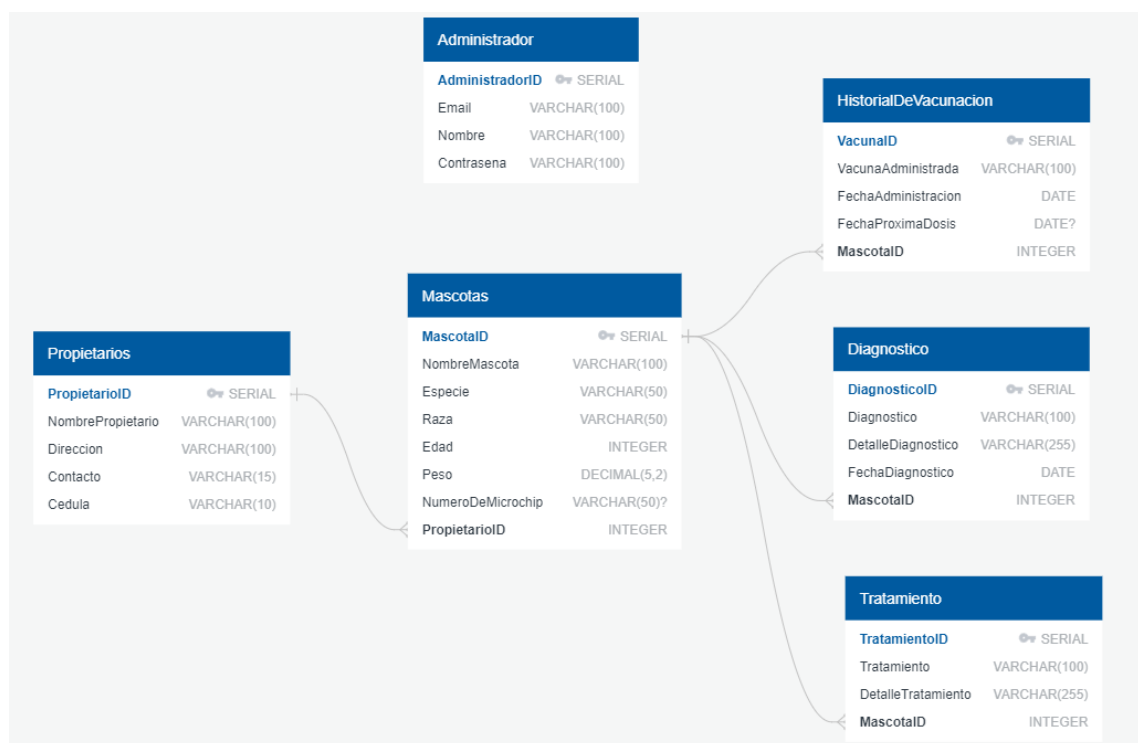


Figura 5. Diagrama de la base de datos.

Capítulo II

Construcción del Sistema

El desarrollo de un sitio web para un centro médico veterinario implica la integración de varios conceptos, la comunicación digital y la gestión de información. Este enfoque teórico proporcionará un marco conceptual que sustentará el diseño y la implementación del proyecto, asegurando que se cumplan los objetivos planteados de manera eficiente y efectiva.

2.1 Comunicación Digital

La comunicación digital es fundamental para cualquier proyecto que busca mejorar la interacción entre una organización y sus usuarios a través de plataformas en línea. En el contexto de un centro médico veterinario, la comunicación digital permitirá compartir información relevante sobre los servicios ofrecidos, carnets de mascotas con su historial clínico y actualizaciones, así como establecer un canal de contacto directo con los clientes.

2.2 Diseño de Experiencia de Usuario

El diseño de experiencia de usuario es crucial para crear una interfaz intuitiva y accesible que facilite la navegación y el acceso a la información. Un buen diseño de UX garantizará que los usuarios encuentren fácilmente la información que necesitan y puedan interactuar con el sitio web de manera efectiva.

2.3 Gestión de Bases de Datos

La implementación de una base de datos es fundamental. Se examinarán métodos eficaces en la gestión de base de datos, considerando la escalabilidad y la seguridad de la información. La literatura sobre sistemas de base de datos ayudara a seleccionar la tecnología más apropiada para este proyecto, para el momento el uso de PostgreSQL resulta ser útil debido al uso anterior en la metería, sumado a su amplia información y a ser un gestor de base de datos de código abierto.

2.4 Tecnologías de Desarrollo Web

El uso de tecnologías de desarrollo web como HTML, CSS, JavaScript, React, jugaran un papel esencial en el desarrollo de un sitio web dedicado a un centro médico veterinario.

En primer lugar, el uso de HTML y CSS son fundamentales para el desarrollo y el diseño de interfaz de usuario de forma efectiva. HTML ayudará a definir la estructura de las páginas web, por otro lado, CSS se utilizará para dar un estilo y presentación visual agradable a dichas estructuras, lo que garantiza una experiencia de usuario agradable y fácil de entender.

Por otro lado, JavaScript al tratarse de un lenguaje de programación del lado del usuario, permitirá una conexión de interactividad en el sitio web ya que facilita la implementación de funcionalidades dinámicas, como una actualización en tiempo real de la información, ya sea carnets de mascotas con su historial clínico y programas del centro médico veterinario, información sobre los médicos veterinarios o en cualquier otra necesidad que mejore la experiencia del usuario.

El uso de frameworks como React ayuda en la simplificación de interfaces de usuario complejas, ya que permite descomponer la aplicación en componentes reutilizables, lo que facilita el mantenimiento del código. Por otra parte, Node.js gracias a su amplia cantidad de bibliotecas lo convierte en herramienta muy útil para el desarrollo web gracias a su uso previo, Node.js más que nada se utilizará en tareas de backend, como la gestión de la base de datos y la lógica del sitio web.

Etapa	Herramienta de desarrollo	Descripción
FrontEnd	• React • JSX • CSS • Axios • Vite	El desarrollo frontend se lo realizo usando tecnologías como React, un framework que permite usar JavaScript para el desarrollo de interfaz de usuario, con el uso de JavaScript XML (JSX), permitiendo realizar una interfaz cansilla, de esta manera permite crear componentes de código reutilizables. Para implementar una interfaz intuitiva, se empleó CSS para colocar estilos y dar detalles visuales para diseñar una interfaz atractiva y mejorando la navegación, permitiendo visualizar menús, botones y más para mejorar la experiencia del usuario. Otra función de CSS es que permite realizar una plataforma reactiva, permitiendo la compatibilidad con varios dispositivos ya que permite a la plataforma web ajustarse a la pantalla del dispositivo como un ordenador, tableta o teléfono móvil. Axios se utilizó para realizar solicitudes HTTP desde el frontend al backend, permitiendo la comunicación con la API REST y la obtención de datos en tiempo real, como la información de mascotas y propietarios. Vite fue la herramienta de construcción utilizada para el desarrollo del frontend, permitiendo un rendimiento optimizado y una experiencia de desarrollo más rápida.

BackEnd	• Express.js • Sequelize • Cors	El desarrollo backend se realizó usando JavaScript y Sequelize (ORM), para mejorar la gestión de consultas a la base de datos. Utilizando Express.js, un framework de Node.js, facilita la creación de API REST, y la gestión de rutas y solicitudes HTTP. Cors se implementó en el desarrollo para manejar las políticas de seguridad relacionadas con el intercambio de datos entre el frontend y el backend para una comunicación segura.
Base de Datos	• PostgreSQL	Se uso PostgreSQL para base de datos relacional, debido a su robustez, opciones de escalabilidad a futuro y capacidad para manejar grandes volúmenes de datos de manera eficiente.

Tabla 1. Estructura

Capítulo III

Pruebas y Estabilización

En esta fase del desarrollo del sistema **Climovet**, se llevaron a cabo diversas pruebas para garantizar la calidad, seguridad y estabilidad del sistema antes de su implementación final. Se aplicaron pruebas en diferentes niveles para evaluar su rendimiento, integración y experiencia de usuario.

3.1 Tipos de Pruebas Realizadas

Las pruebas realizadas se clasificaron en diferentes categorías para evaluar aspectos específicos del sistema:

Tipo de Prueba	Descripción
Pruebas Unitarias	Se verificó el correcto funcionamiento de cada módulo del sistema de forma individual.
Pruebas de Integración	Se validó la comunicación entre el frontend, backend y la base de datos.
Pruebas Funcionales	Se probó cada funcionalidad del sistema para asegurar que cumpla con los requisitos especificados.
Pruebas de Seguridad	Se realizaron validaciones para detectar vulnerabilidades y garantizar la protección de datos.
Pruebas de Rendimiento	Se midió la velocidad de carga y respuesta del sistema bajo diferentes niveles de concurrencia.
Pruebas de Usabilidad	Se evaluó la facilidad de uso y navegación del sistema por parte de los usuarios finales.

Tabla 2. Tipos de Pruebas

3.2 Resultados de las Pruebas

3.2.1 Pruebas Unitarias

Se realizaron pruebas unitarias en cada uno de los componentes del backend y frontend. Para ello, se utilizó **Jest** en el frontend y **Mocha** en el backend.

Módulo Evaluado	Resultado Esperado	Resultado Obtenido	Estado
Autenticación de Usuarios	Acceso correcto con credenciales válidas	Acceso exitoso	Aprobado
Gestión de Mascotas	Registro de mascotas sin errores	Registros completados con éxito	Aprobado
Gestión de Propietarios	Almacenar y actualizar información correctamente	Se guardaron y actualizaron los datos correctamente	Aprobado
Seguridad de Contraseñas	Encriptación antes de almacenar en la BDD	Contraseñas encriptadas correctamente	Aprobado

Tabla 3. Pruebas Unitarias

3.2.2 Pruebas de Integración

Se verificó la correcta comunicación entre el frontend, backend y la base de datos. Se utilizó **Postman** para realizar peticiones y validar respuestas.

Prueba	Módulos Involucrados	Resultado Esperado	Resultado Obtenido	Estado
Recuperación de datos de mascotas	Frontend - Backend - BDD	Información mostrada correctamente	Datos cargados sin errores	Aprobado
Registro de propietarios	Frontend - Backend - BDD	Información guardada y reflejada en la BDD	Datos almacenados correctamente	Aprobado
Validación de credenciales	Frontend - Backend	Acceso con credenciales correctas	Acceso exitoso	Aprobado

Tabla 4. Pruebas de Integración

3.2.3 Pruebas de Seguridad

Para garantizar la protección de los datos, se realizaron pruebas de inyección SQL, validación de entradas y pruebas de acceso no autorizado.

Prueba	Método	Resultado Esperado	Resultado Obtenido	Estado
Inyección SQL	Prueba manual en campos de entrada	El sistema debe rechazar intentos de inyección	Intentos bloqueados	Aprobado
Protección de contraseñas	Intento de acceso directo a la BD	No debe ser posible acceder a contraseñas sin autenticación	Datos protegidos y encriptados	Aprobado
Intento de acceso sin token JWT	Petición a endpoint protegido sin token	Rechazo de la petición	Acceso denegado correctamente	Aprobado

Tabla 5. Pruebas de Seguridad

3.2.4 Pruebas de Rendimiento

Se realizaron pruebas de carga con **JMeter** para evaluar el rendimiento del sistema bajo diferentes niveles de concurrencia.

Usuarios Concurrentes	Tiempo de Respuesta Esperado	Tiempo de Respuesta Obtenido	Estado
10 usuarios	< 1s	0.8s	Aprobado
50 usuarios	< 2s	1.7s	Aprobado
100 usuarios	< 3s	2.5s	Aprobado

Tabla 6. Pruebas de Rendimiento

Conclusiones

Se logró desarrollar un sistema web funcional que mejora la gestión del Centro Médico Veterinario Climovet, optimizando la interacción entre los clientes y la institución.

La implementación de tecnologías modernas como React, Node.js y PostgreSQL permitió crear una plataforma eficiente, escalable y segura.

Se logró una interfaz de usuario intuitiva y responsiva, garantizando accesibilidad desde múltiples dispositivos.

Las pruebas demostraron que el sistema es estable y seguro, cumpliendo con los estándares de calidad esperados.

La correcta encriptación de contraseñas y el uso de autenticación JWT mejoraron la seguridad del sistema.

Se optimizó la velocidad de respuesta del sistema, asegurando un tiempo de carga rápido incluso bajo alta concurrencia.

Recomendaciones

Implementar mejoras en el sistema de autenticación, como autenticación multifactor (MFA) para reforzar la seguridad.

Realizar monitoreo constante del rendimiento del sistema para detectar posibles fallos y aplicar correcciones en tiempo real.

Considerar la integración de notificaciones automáticas, como recordatorios de vacunación de mascotas.

Mantener actualizadas las dependencias del sistema para evitar vulnerabilidades de seguridad.

Explorar la posibilidad de desarrollar una aplicación móvil complementaria para mejorar aún más la accesibilidad y experiencia de usuario.

Incluir capacitaciones periódicas para el personal veterinario, asegurando el correcto uso del sistema y el mantenimiento de la base de datos.

Referencias bibliográficas

MDN web docs. (s.f). HTML: HyperText Markup Language.

<https://developer.mozilla.org/en-US/docs/Web/HTML>

MDN web docs. (s.f). CSS: Cascading Style Sheets.

<https://developer.mozilla.org/en-US/docs/Web/CSS>

MDN web docs. (s.f). JavaScript.

<https://developer.mozilla.org/en-US/docs/Web/JavaScript>

Node.js (s.f). Introduction to Node.js.

<https://nodejs.org/en/learn/getting-started/introduction-to-nodejs>

Express (s.f) Fast, unopinionated, minimalist web framework for Node.js

<https://expressjs.com>

React. (s.f.) <https://react.dev> React. (s.f.) A JavaScript library for building user interfaces.

<https://legacy.reactjs.org>

PostgreSQL. (s.f.) The World's Most Advanced Open Source Relational Database.

<https://www.postgresql.org>

Anexos

Anexo 1 Estructuras y Diagramas

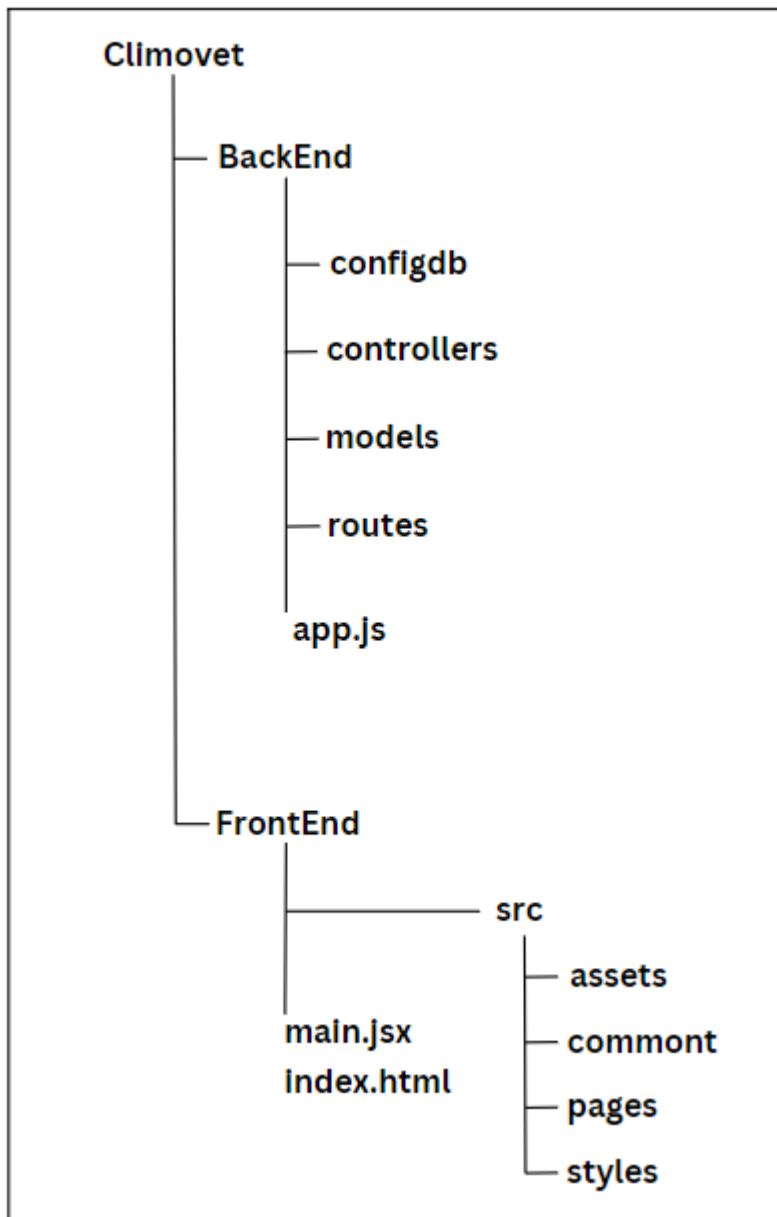


Figura 6. Estructura de Directorios

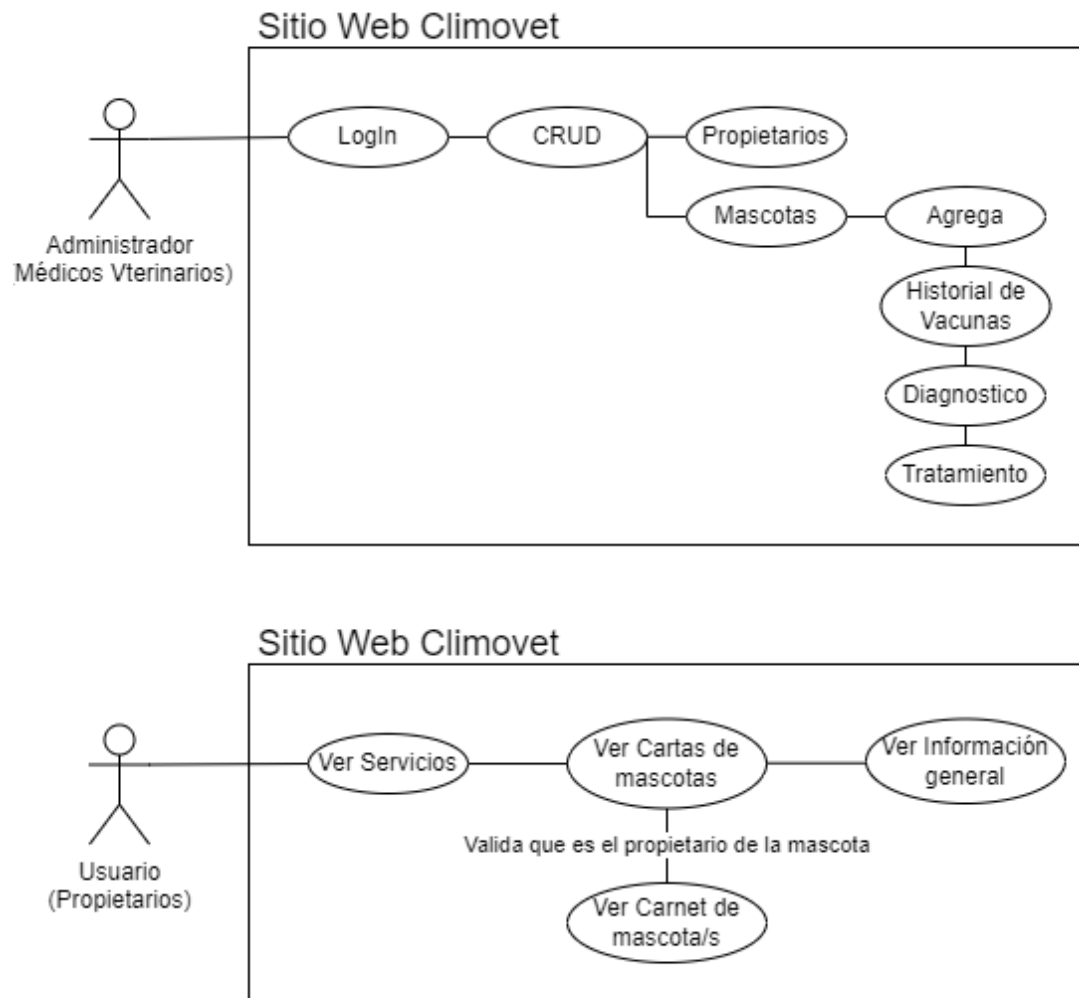


Figura 7. Diagrama Entidad Relación

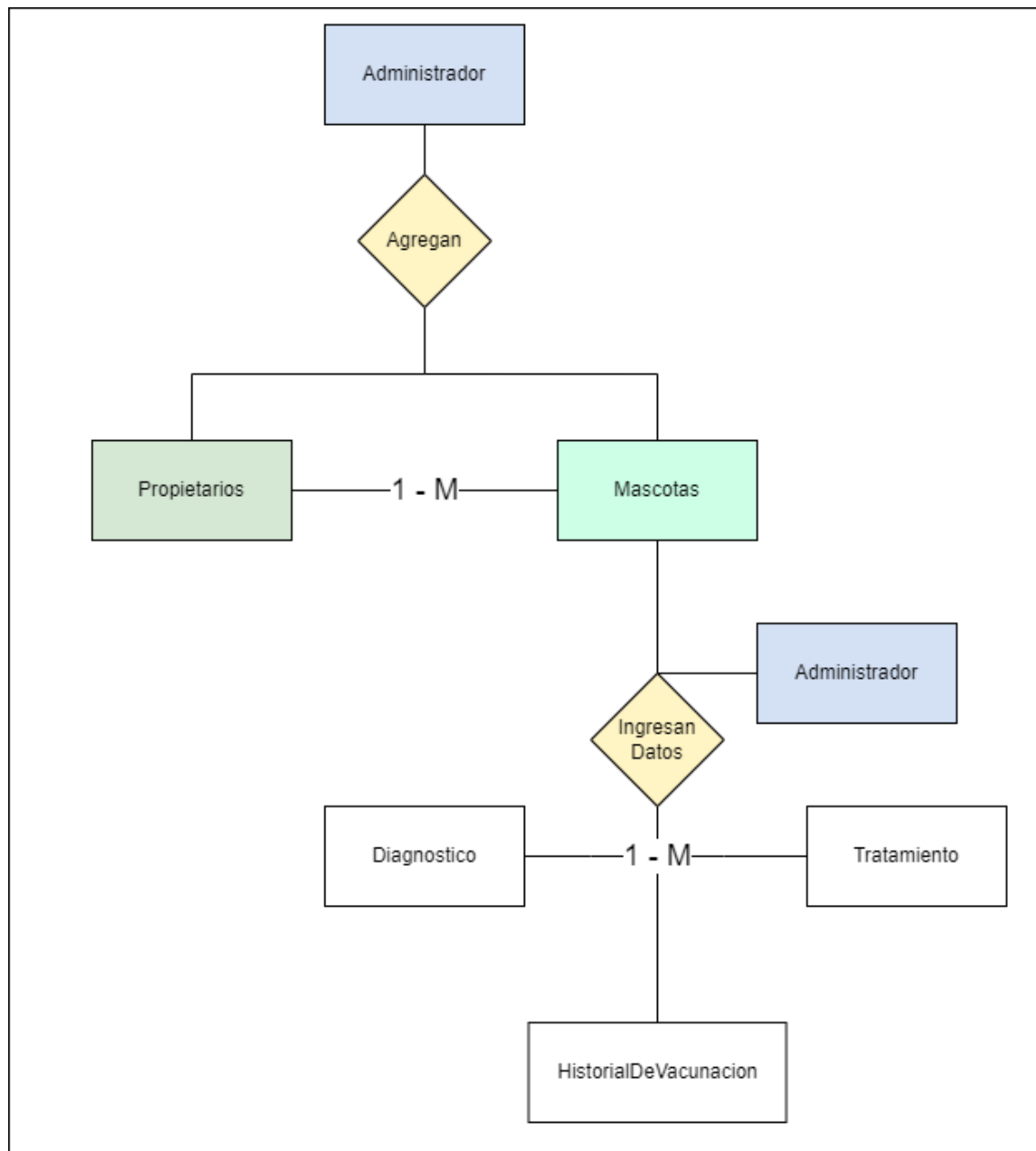


Figura 8. Diagrama Caso de uso

Anexo 2

Link GitHub Repositorio

<https://github.com/DiegoConstante/Climovet>