



UNIVERSIDAD ESTATAL DE BOLÍVAR

**FACULTAD DE CIENCIAS ADMINISTRATIVAS, GESTIÓN
EMPRESARIAL E INFORMÁTICA**

CARRERA DE SOFTWARE

**TRABAJO DE INTEGRACIÓN CURRICULAR
PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN SOFTWARE**

FORMA: PROYECTO TECNOLÓGICO

TEMA:

**SISTEMA WEB PARA LA GESTIÓN Y ADOPCIÓN DE MASCOTAS EN EL
REFUGIO DE ANIMALES DE GUARANDA**

AUTOR(A):

CRISTOFER FABIAN MUYULEMA CUVI

DIRECTOR:

ING. RODRIGO DEL POZO DURANGO, MSC.

GUARANDA – ECUADOR

2026

TEMA DEL PROYECTO TECNOLÓGICO

**SISTEMA WEB PARA LA GESTIÓN Y ADOPCIÓN DE MASCOTAS EN EL REFUGIO
DE ANIMALES DE GUARANDA**

DEDICATORIA

Primero quiero agradecer a Dios, porque en los momentos más difíciles de este camino fue Él quien me dio las fuerzas para no rendirme y seguir adelante hasta llegar aquí. A mis padres y a mis hermanos, gracias, por tanto. Ustedes me enseñaron con su ejemplo lo que significa esforzarse y no rendirse, y siempre confiaron en mí, incluso cuando yo mismo dudaba. Todo lo que soy hoy tiene mucho de lo que ustedes me dieron.

A mi familia y amigos, gracias por estar ahí, por escucharme, por animarme cuando las cosas se ponían difíciles y por alegrarse conmigo con cada avance, por pequeño que fuera.

A mi universidad, porque ahí no solo aprendí de mi carrera, sino que crecí como persona. Fue el lugar donde esas ideas que parecían lejanas se fueron convirtiendo poco a poco en algo real.

A mi tutor, gracias por confiar en este proyecto desde el principio y por guiarme con paciencia. Sus consejos y su experiencia fueron clave para que esta tesis saliera adelante.

Y a mis compañeros, gracias por las conversaciones, las dudas compartidas y por aprender juntos en este proceso. Sin darnos cuenta, cada charla sumó algo a este trabajo.

A todos los que de alguna forma fueron parte de este camino: gracias por creer en mí y por ayudarme a llegar hasta acá.

AGRADECIMIENTO

Al culminar esta etapa fundamental que representa nuestro trabajo de grado, deseamos expresar nuestro más sincero agradecimiento a quienes hicieron posible este logro. En primer lugar, a Dios, por ser la guía y la fortaleza que nos permitió culminar la carrera universitaria y alcanzar esta meta profesional. De igual forma, extendo mi agradecimiento a mi familia, por ser la base y soporte esencial que permitió la culminación de este proyecto. Sin su aliento y comprensión, nada de esto hubiese sido posible.

Agradezco de manera especial al ingeniero Rodrigo del Pozo Durango, director del trabajo de titulación, por su dirección, paciencia e invaluable conocimiento que guió paso a paso en el desarrollo del Sistema web para la gestión y adopción de mascotas en el refugio de animales de Guaranda.

Extiendo un reconocimiento especial a los ingenieros Darwin Carrión y Edgar Rivadeneira, cuya mentoría y enseñanzas fueron fundamentales para el desarrollo metodológico de esta investigación, así como para la formación integral a lo largo de la carrera.

A mis profesores y compañeros, con quienes he compartido años de formación, gracias por formar parte de este proceso, por las valiosas lecciones y por el apoyo mutuo que me permitió convertirme en profesional.

A todos, por su fe, ayuda e inspiración: MUCHAS GRACIAS.

CERTIFICADO DE VALIDACIÓN



FACULTAD DE CIENCIAS
ADMINISTRATIVAS,
GESTIÓN EMPRESARIAL
E INFORMÁTICA

CERTIFICADO DE VALIDACIÓN

Ing. Rodrigo del Pozo Durango, Ing. Darwin Paul Carrion Buenaño, Ing. Edgar Patricio Rivadeneira Ramos, en su orden de director y Pares Académicos del Trabajo de Integración Curricular "SISTEMA WEB PARA LA GESTION Y ADOPCION DE MASCOTAS EN CENTRO DE ATENCIÓN EMERGENTE DE FAUNA URBANA EN GUARANDA (CAEFU)" Desarrollado por los estudiantes Cristofer Fabian Muyulema Cuvi

CERTIFICAN

Que, luego de revisado el Trabajo de Integración Curricular en su totalidad, cumple con las exigencias académicas de la carrera SOFTWARE, por lo tanto, autorizamos su presentación y defensa.

Guaranda, 03 de Julio del 2026

Ing. Rodrigo Humberto
Del Pozo Durango
Director

Ing. Edgar Patricio Rivadeneira
Ramos

Par Académico

Ing. Darwin Paul Carrion Buenaño

Par Académico

DERECHOS DE AUTOR



**BIBLIOTECA
GENERAL**

DERECHOS DE AUTOR

Yo/nosotros Muyulema Cuvi Cristofer Fabian portador/res de la Cédula de Identidad No 0202161063 en calidad de autor/res y titular / es de los derechos morales y patrimoniales del Trabajo de Titulación:

SISTEMA WEB PARA LA GESTIÓN Y ADOPCIÓN DE MASCOTAS EN EL REFUGIO DE ANIMALES DE GUARANDA, modalidad proyecto tecnológico, de conformidad con el Art. 114 del CÓDIGO ORGÁNICO DE LA ECONOMÍA SOCIAL DE LOS CONOCIMIENTOS, CREATIVIDAD E INNOVACIÓN, concedemos a favor de la Universidad Estatal de Bolívar, una licencia gratuita, intransferible y no exclusiva para el uso no comercial de la obra, con fines estrictamente académicos. Conservamos a mi/nuestro favor todos los derechos de autor sobre la obra, establecidos en la normativa citada.

Así mismo, autorizo/autorizamos a la Universidad Estatal de Bolívar, para que realice la digitalización y publicación de este trabajo de titulación en el Repositorio Digital, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

El (los) autor (es) declara (n) que la obra objeto de la presente autorización es original en su forma de expresión y no infringe el derecho de autor de terceros, asumiendo la responsabilidad por cualquier reclamación que pudiera presentarse por esta causa y liberando a la Universidad de toda responsabilidad.

A handwritten signature in blue ink, appearing to read 'Muyulema Cuvi Cristofer Fabian', is written over a faint, circular official stamp.

Muyulema Cuvi Cristofer Fabian
CI: 0202161063

ÍNDICE DE CONTENIDOS

TEMA DEL PROYECTO TECNOLÓGICO.....	ii
DEDICATORIA.....	iii
AGRADECIMIENTO.....	iv
CERTIFICADO DE VALIDACIÓN	v
DERECHOS DE AUTOR.....	vi
INDICE DE CONTENIDOS.....	vii
INDICE DE TABLAS.....	x
INDICE DE FIGURAS	xi
INTRODUCCIÓN.....	1
RESUMEN.....	2
ABSTRACT	3
KICHWA	4
CAPÍTULO I.....	5
FORMULACIÓN GENERAL DEL PROYECTO	5
1.1. Tema.....	5
1.2. Descripción del Problema	5
1.3. Justificación.....	6
1.4. Objetivos: General y Específicos.....	8
CAPÍTULO II	9
MARCO TEÓRICO	9
2.1. Antecedentes.....	9
2.2. Científico	11

	viii
2.2.1. Desarrollo de software	11
2.2.2. Aplicaciones web.....	18
2.3. Conceptual	20
2.3.1. Conceptos de sistemas y tecnología web	20
2.3.2. Desarrollo de software y arquitectura	22
2.3.3. Interacción del sistema	23
2.3.4. Gestión de procesos en el sistema	24
2.4. Legal	25
2.5. Georeferencial.....	26
CAPÍTULO III	28
METODOLOGÍA.....	28
3.1. Metodología de Desarrollo de Software	28
3.2. Fases del proceso de desarrollo	29
3.2.1. Planificación	29
3.2.2. Diseño	30
3.3. Técnicas e Instrumentos de Recopilación de Datos	32
3.3.1. Técnicas	32
3.3.2. Instrumentos.....	33
CAPÍTULO IV	34
INGENIERÍA DEL PROYECTO	34
4.1. Planificación	34
4.1.1 Tabulación y análisis de datos	34
4.1.2. Análisis de la entrevista	35

4.1.3 Historias de usuario	36
4.1.4. Registro de riesgos.....	42
4.1.5. Gestión del trabajo mediante Kanban.....	44
4.2. Diseño.....	46
4.2.1. Arquitectura de software.....	46
4.2.2. Vistas arquitectónicas 4+1	47
4.3. Codificación.....	57
4.3.1. Backend Laravel	58
4.3.2. Frontend React.....	63
4.3.3. Interfaces de pantalla	67
4.4. Pruebas	78
4.4.1 Propósito	78
4.4.2 Diseño y desarrollo de casos de prueba.....	78
4.5 Entrega.....	81
CONCLUSIONES.....	83
RECOMENDACIONES	84
BIBLIOGRAFÍA.....	85
ANEXOS.....	94

ÍNDICE DE TABLAS

Tabla 1. Beneficios de la metodología ágil	16
Tabla 2. Beneficios y limitaciones de MVC.....	17
Tabla 3. Lenguajes de programación.....	23
Tabla 4. Resultados generales de la guía de observación aplicada al CAEFU.....	34
Tabla 5. Hallazgos y requerimientos identificados con la guía de observación	35
Tabla 6. Síntesis de resultados de la entrevista aplicada al técnico del CAEFU	35
Tabla 7. Historia de usuario N° HU-01 — Registro de mascota en el sistema	37
Tabla 8. Historia de usuario N° HU-02 — Registro de cuenta de usuario	37
Tabla 9. Historia de usuario N° HU-03 — Visualización del catálogo de mascotas.	38
Tabla 10. Historia de usuario N° HU-04 — Envío de solicitud de adopción.....	39
Tabla 11. Historia de usuario N° HU-05 — Revisión y resolución de solicitudes de adopción	40
Tabla 12. Historia de usuario N° HU-06 — Generación de reportes estadísticos.....	42
Tabla 13. Registro de riesgos del proyecto.....	43
Tabla 14. Tablero Kanban	45
Tabla 15. Stakeholders.....	79
Tabla 16. Gestor de pruebas	79
Tabla 17. Registro y autenticación de usuarios	80
Tabla 18. Gestión de mascotas	80
Tabla 19. Gestión de solicitudes de adopción	81
Tabla 20. Notificaciones y reportes	81
Tabla 21. Presupuesto estimado - (CAEFU – Guaranda).....	96

ÍNDICE DE FIGURAS

Figura 1. Mapa Geográfico de la ubicación del Centro de Atención Emergente de Fauna Urbana CAEFU en Guaranda	27
Figura 2. Diagrama de patrón arquitectónico de N-capas	46
Figura 3. Diagrama de caso de uso (escenarios)	48
Figura 4. Diagrama de clases (Vista lógica).....	49
Figura 5. Diagrama de secuencia – Proceso de adopción	51
Figura 6. Diagrama de paquetes (Vista de despliegue)	53
Figura 7. Diagrama de actividad — proceso de adopción.....	55
Figura 8. Diagrama de despliegue (Vista física)	57
Figura 9. Logo del CAEFU	67
Figura 10. Interfaz principal (Home)	67
Figura 11. Inicio de sesión para administrador y usuario.....	68
Figura 12. Registro de nuevo usuario.....	68
Figura 13. Panel del usuario adoptante	69
Figura 14. Panel de control	69
Figura 15. Gestión de mascotas del panel administrativo	70
Figura 16. Gestión de solicitudes de adopción del panel administrativo.....	70
Figura 17. Gestión de usuarios del panel administrativo.....	71
Figura 18. Notificaciones inteligentes del panel administrativo.....	71
Figura 19. Pestaña de datos personales del módulo de configuración del administrador	72
Figura 20. Pestaña de seguridad, para el cambio de contraseña, del módulo de configuración del administrador.....	72

INTRODUCCIÓN

El aumento de animales domésticos abandonados afecta la salud pública, bienestar animal y convivencia ciudadana, esta situación impulsa el desarrollo de estrategias de tenencia responsable y adopción (Mota et al., 2021). En Guaranda, el Centro de Atención Emergente de Fauna Urbana (CAEFU) atiende esta problemática a través de la atención veterinaria y gestión de adopciones. Sin embargo, la administración manual de información, falta de seguimiento de solicitudes, ausencia de un catálogo de mascotas y un sistema de notificaciones limitan la eficiencia del proceso.

Este trabajo propone el diseño y desarrollo de un sistema web para gestionar el proceso de adopción de mascotas, utiliza una arquitectura de tres capas con React, Laravel y PostgreSQL, e incorpora un formulario de adopción con verificación de vivienda, ubicación geográfica, agenda de visitas domiciliarias, panel administrativo, seguimiento de solicitudes y notificaciones automáticas mediante la API de WhatsApp.

La propuesta busca fortalecer la tenencia responsable de mascotas y reducir el abandono animal en el cantón Guaranda, mediante un proceso de adopción transparente y eficiente. Centraliza la información institucional, optimiza la gestión administrativa y mejora la trazabilidad de los casos, mediante una arquitectura escalable y adaptable a las necesidades de la institución pública.

Los objetivos específicos son el diagnóstico de los procesos de gestión y adopción, desarrollo de arquitectura del sistema, implementación de módulos de gestión de mascotas, solicitudes, visitas domiciliarias, reportes estadísticos y realizar pruebas de funcionamiento.

El documento se organiza en cuatro capítulos, el primero presenta el problema, justificación y objetivos; el segundo desarrolla el marco teórico; el tercero describe la metodología de investigación y el cuarto presenta el análisis de requerimientos, planificación del proyecto, diseño del sistema bajo el modelo de vistas 4+1, seguido de las conclusiones y recomendaciones.

RESUMEN

El presente proyecto tecnológico tiene como objetivo diseñar un Sistema web para la gestión y adopción de mascotas en el refugio de animales de Guaranda, con el propósito de fortalecer el manejo del centro mediante la digitalización de información, difusión de mascotas y registro de ciudadanos interesados en adoptar. La investigación se fundamenta en las teorías de arquitectura de software, metodologías ágiles de desarrollo y sistemas de información. Para el desarrollo del sistema se emplea la arquitectura de capas (React, Laravel y PostgreSQL); el sistema propuesto permite al personal administrador gestionar el registro de mascotas y su historial médico, revisar y aprobar solicitudes de adopción en etapas, administrar usuarios y notificaciones; mientras que el ciudadano puede explorar el catálogo, registrarse, enviar solicitudes de adopción con información de su hogar y dar seguimiento al estado de su trámite. El proyecto considera la normativa ecuatoriana en materia de protección de datos personales y propiedad intelectual. Como resultado contribuye a la transformación digital del CAEFU, optimizando sus procesos internos y promoción de adopción responsable de fauna urbana en el cantón Guaranda, Ecuador.

Palabras clave: adopción de mascotas, arquitectura n-capas, gestión de información, metodologías ágiles, sistema web, transformación digital.

ABSTRACT

This research project aims to design a Web-based system for pet management and adoption at the Guaranda animal shelter, with the aim of strengthening the center's operations through the digitization of information, the promotion of pets, and the registration of citizens interested in adopting. The research is based on theories of software architecture, agile development methodologies, and information systems. For the development of the system, a layered architecture (React, Laravel y PostgreSQL) is employed; the proposed system allows administrative staff to manage pet records and medical histories, review and approve adoption applications in stages, and manage users and notifications; while citizens can browse the catalog, register, submit adoption applications with information about their household, and track the status of their application. The guidelines established in Ecuadorian regulations regarding the protection of personal data and intellectual property are taken into account. As a result, the project contributes to the digital transformation of CAEFU to optimize its internal processes and promote the responsible adoption of urban wildlife in the canton of Guaranda, Ecuador

Keywords: pet adoption, n-tier architecture, information management, agile methodologies, web system, digital transformation.

KICHWA

Kay rurayka Guaranda llaqtapi tiyak allku kuna, misi kuna shuk shuk wiwa kuna wiñachiy, rikuchiy, chaskiy shinallatak kuyaywan wasiman apay ruraykunata yanapanapak shuk web sistema ruraymi. Kay sistema Cllamkaykunata alli rurachun, willaykunata tantachun, wasiman apanak wiwakunata rikuchichun, shinallatak wiwata wasiman apanapak munak runakunata killkachun nishpa rurashka. Kay rurayka software arquitectura, ágiles metodologías, shinallatak información sistema yachaykunapi sinchiyashkami. Sistema n-capas arquitectura ukupi React, Laravel, PostgreSQL tantanakuywan rurashkami. Sistema yanapashpa, llamkakkuna wiwakunata killkayta, hampi willaykunata waqaychayta, wasiman apanapak mañaykunata rikuyta, alli nishpa mana alli nishpa akllayta, shinallatak usuario kuna, willachiykunata kamachiyta usankakuna. Chaymantami llakta runakunaka wiwakunata rikunkakuna, killkarinkakuna, wasiman apanapak mañayta apachinkakuna, wasipi kawsaymanta willaykunata churankakuna, shinallatak mañaypa imashina rikurishkata katinkakuna. Kay sistema Ecuador llaktapa sapalla willay waqaychay kamachikunata, shinallatak yachay ruraypa hayñikunata yuyarishpa rurashkami. Chayrayku CAEFU llamkaykunata alli rurachin, uku ruraykunata ushaywan paktachin, shinallatak Guaranda kitipi wiwakunata kuyaywan wasiman apayta kallpachin.

Shimikuna: wiwa wasiman apay, arquitectura n-capas, willay kamachiy, ágiles metodologías, web sistema, digital transformación.

CAPÍTULO I

FORMULACIÓN GENERAL DEL PROYECTO

1.1. Tema

Sistema web para la gestión y adopción de mascotas en el refugio de animales de Guaranda.

1.2. Descripción del Problema

El abandono y mala gestión de animales domésticos representan un problema global, entre sus causas destacan la falta de espacio, recursos económicos, rechazo a conductas molestas de las mascotas, pérdida de interés, reemplazo por otro animal, irresponsabilidad y deficiente ética de los propietarios (Mota et al., 2021). Se estima que, en seis años, una perra y su descendencia generan hasta 67.000 cachorros, no obstante, dichos animales muchas veces no alcanzan la etapa reproductiva debido a las duras condiciones de vida que enfrentan (Estepa et al., 2023).

Aunque Ecuador dispone de un marco legal de protección animal, fortalecido desde la Constitución de 2008 que reconoce a los animales como sujetos de derechos, su aplicación efectiva aún presenta dificultades (Guambo et al., 2025). En ese contexto el Centro de Atención Emergente de Fauna Urbana (CAEFU) del cantón Guaranda protege a mascotas en situación de abandono, sin embargo, la gestión actual de sus procesos limita su efectividad.

La tenencia irresponsable de mascotas, falta de control y registro contribuye al aumento de animales en situación de calle, esta situación genera riesgos sanitarios y afecta la dinámica social y ambiental del cantón. En el CAEFU el registro manual de animales y adoptantes provoca pérdida de información, duplicidad de datos y ausencia de control centralizado, escenario que origina retrasos en la validación de solicitudes y generación de reportes confiables.

El estudio de Pata et al. (2024) señala que los procesos de adopción prolongados constituyen un obstáculo para los usuarios y administradores de albergues, además de implicar un alto consumo de tiempo y recursos. En el CAEFU la gestión manual de procedimientos y limitado uso de medios digitales reducen la visibilidad de mascotas en adopción, pues su difusión se restringe a redes sociales con alcance limitado y dificultan la conexión entre el refugio y los adoptantes. Como resultado los animales permanecen largos periodos sin ser adoptados, esto incrementa los costos de mantenimiento y limita la capacidad del refugio para acoger a nuevos animales en situación de vulnerabilidad.

A esta problemática se suma que los procesos actuales en el CAEFU dificultan el seguimiento post adopción y pone en riesgo el bienestar animal y credibilidad del proceso. Además, los procesos manuales consumen tiempo del personal y voluntarios en tareas administrativas, esto limita su capacidad para enfocarse en acciones estratégicas como campañas de sensibilización y educación comunitaria.

Por otra parte, la falta de reportes estadísticos es también un problema ya que, sin datos sobre adopciones, tiempo de permanencia o causas de devolución, la toma de decisiones se ve afectada. En este contexto, la gestión del CAEFU del cantón Guaranda resulta ineficiente y con limitada capacidad de respuesta, por ello, la implementación de soluciones tecnológicas como el desarrollo de un sistema web que optimice la gestión y adopción de mascotas se plantea como una alternativa viable y sustentada en la realidad local.

1.3. Justificación

El desarrollo de un Sistema web para la gestión y adopción de mascotas en el refugio de animales de Guaranda, constituye un aporte relevante en el ámbito social y tecnológico; en el plano social, aborda una problemática local de bienestar animal, donde el uso de herramientas digitales podría mejorar la gestión de los refugios, fomentar la adopción responsable y facilitar el seguimiento de los procesos, optimizando así el uso de recursos.

Desde la perspectiva tecnológica, el sistema permitirá modernizar los procesos administrativos del CAEFU mediante la automatización de registros, monitoreo de adoptantes y generación de reportes estadísticos. Esto contribuye al bienestar animal, fortalece la conexión con la ciudadanía, promueve la transparencia institucional y mejora la toma de decisiones basada en información confiable.

El sistema web generará beneficios directos al personal administrativo y voluntarios, quienes podrán gestionar registros, tramitar adopciones y elaborar reportes de manera más ágil al reducir el tiempo dedicado a tareas repetitivas. De forma indirecta se beneficiará a los animales del refugio, ya que una mayor visibilidad digital y procesos más eficientes aumentarán sus posibilidades de ser adoptados en menor tiempo. Asimismo, los ciudadanos interesados en adoptar contarán con una plataforma accesible para conocer a las mascotas disponibles y gestionar sus solicitudes de manera más organizada.

Para el desarrollo del sistema se propone el uso de una arquitectura moderna basada en Laravel como framework de backend, React para el frontend y PostgreSQL como gestor de base de datos, combinación permitirá la creación de interfaces dinámicas y eficientes. El alcance del sistema incluye módulos para el registro de animales, gestión de solicitudes de adopción, seguimiento post-adopción, generación de reportes estadísticos y una interfaz orientada a la difusión de mascotas disponibles.

El presente proyecto se enmarca en la línea de investigación de Ingeniería de Software, Redes y Telecomunicaciones, en específico en la sublínea de Diseño e Implementación de Sistemas de Información de la carrera de Software de la Universidad Estatal de Bolívar, ya que integra el desarrollo de soluciones tecnológicas orientadas a optimizar procesos administrativos y mejorar la gestión de información en el CAEFU de Guaranda.

1.4. Objetivos: General y Específicos

Objetivo General

Desarrollo de un sistema web para la gestión y adopción de mascotas en el refugio de animales de Guaranda.

Objetivos Específicos

- Diagnosticar la situación actual de los procesos de gestión y adopción de mascotas en el CAEFU de Guaranda mediante el levantamiento de información.
- Desarrollar la arquitectura del sistema web que incluye la estructura de la base de datos e interfaz de usuario.
- Modelar la base de datos para la gestión de la información de solicitudes de adopción y seguimiento post-adopción.
- Realizar las pruebas del sistema web que avale su funcionamiento

CAPÍTULO II

MARCO TEÓRICO

2.1. Antecedentes

De acuerdo con Hidalgo & Usiña (2025), desarrollaron un sistema web orientado a la adopción y localización de mascotas en la Fundación Fauna Urbana de Ibarra, cuyo propósito fue optimizar la gestión de adopciones y recuperación de animales extraviados. Para ello, emplearon la metodología ágil eXtreme Programming, que permitió estructurar una plataforma dinámica que facilita la interacción entre usuarios y administradores; el sistema permite a los usuarios registrarse, buscar mascotas disponibles y reportar animales perdidos; por su parte, los administradores gestionan solicitudes de adopción, actualizan el estado de las mascotas y registran casos de extravío. En este sentido, la plataforma incorpora mecanismos de seguridad, como la encriptación de contraseñas y procesos de autenticación, durante la fase de implementación, se desarrolló la lógica del backend para el procesamiento y almacenamiento de la información, mientras que con MySQL, gestionada a través de Laravel, garantizó la integridad y consistencia de los datos. Finalmente, el sistema se entregó con módulos funcionales para el registro de usuarios, inicio de sesión y visualización inicial, además de presentar características de escalabilidad que permiten su crecimiento futuro. En consecuencia, esta propuesta tecnológica contribuye a mejorar los procesos de adopción y promueve el bienestar animal, de esta manera se fortalece la responsabilidad social en la comunidad.

En la investigación de Mieles & Tipanluisa (2022), desarrollaron un sistema web y una aplicación móvil orientados a la gestión de promociones, adopciones y a la conformación de una red colaborativa para la difusión de mascotas extraviadas para la ciudad de Guayaquil. Para ello, adoptaron la metodología ágil Extreme Programming (XP), la cual prioriza un diseño simple, desarrollo iterativo y una interacción constante con el cliente; el proceso inició con la recopilación de historias de usuario que

estructuraron los casos de uso; posterior el equipo estimó los tiempos de desarrollo y modeló los procesos mediante diagramas que incluyen registro de usuarios, inicio de sesión, gestión de promociones y funciones de socialización; además, el cliente participó en la validación de cada funcionalidad. Para la calidad del software se ejecuta pruebas unitarias de integración y validación de forma continua, desarrollado con HTML5, CSS3, JavaScript, PHP y MySQL. Esta solución facilitó la gestión de promociones, socialización de mascotas, búsqueda de hogares y mascotas extraviadas.

La investigación de Varelas & Castro (2026) sobre el desarrollo de un sistema web para la adopción y registro de mascotas, integra un catálogo, perfiles, solicitudes de adopción, registro de usuarios y recursos informativos. Se empleó la metodología Scrum que facilita la entrega progresiva de soluciones; en fase inicial se definen requerimientos, arquitectura del sistema y tecnologías base; después se desarrolla los módulos de autenticación, gestión de mascotas, adopciones y métricas..

En el trabajo de investigación de Gamarra et al. (2026) presenta el desarrollo de un sistema web orientado a la adopción y donación de mascotas; la metodología de la arquitectura Modelo–Vista–Controlador, que separa la lógica, presentación, control, y mejora la mantenibilidad del software. Para la implementación se empleó PHP, MySQL, HTML5, CSS3 y JavaScript, se integró Composer y PHPMailer para la gestión de dependencias y envío de notificaciones. Se obtuvo una plataforma para la visualización de mascotas disponibles e incorporó Singleton y Factory Method para eficiencia y escalabilidad.

El estudio de Cruz-Hernández & Valencia-Esquivel (2023), desarrollaron un sistema de gestión de adopción animal para la Fundación OPAM, basado en la metodología Scrum, que organiza el trabajo en roles como Product Owner, Scrum Master y equipo de desarrollo. Se emplean Angular (frontend), Node.js (backend), PostgreSQL (base de datos), GitLab y Docker para integración, despliegue y consistencia entre entornos. El funcionamiento del sistema se basa en la comunicación entre frontend y backend a través de HTTP, el servidor procesa la lógica de negocio, interactúa con la base de datos y devuelve respuestas en formato JSON.

Los estudios revisados justifican que los sistemas web orientados a la adopción de mascotas mejoran la gestión, interacción con usuarios y eficiencia de los procesos, se emplea metodologías ágiles como XP y Scrum, dado que facilitan el desarrollo iterativo y la adaptación a requerimientos. En cuanto a la implementación, se observa el uso de arquitecturas cliente-servidor y modelos MVC, que garantizan una separación de responsabilidades, mantenibilidad y escalabilidad del sistema; asimismo, el uso de gestores de bases de datos como MySQL y PostgreSQL que permiten almacenar, organizar y gestionar grandes volúmenes de información. Por tanto, la implementación de plataformas web contribuye a optimizar los procesos administrativos, mejorar la experiencia del usuario y fortalecer la comunicación con potenciales adoptantes.

2.2. Científico

2.2.1. *Desarrollo de software*

La metodología de desarrollo de software constituye un marco estructurado que permite planificar, organizar y controlar la construcción de sistemas de información; su aplicación es importante porque en ella se define el tiempo de ejecución, recursos, presupuesto y las actividades del proyecto. En este sentido, las metodologías se clasifican en tradicionales, ágiles e híbridas, cada una con enfoques específicos según las necesidades del desarrollo (Morales-Carrillo et al., 2022).

En este contexto, las metodologías ágiles adquieren notabilidad a partir del Manifiesto Ágil de 2001, en él se establecen principios orientados a priorizar las interacciones humanas, software funcional, colaboración con el cliente y la capacidad de adaptación al cambio. Por lo tanto, este enfoque promueve un desarrollo iterativo e incremental, con entregas continuas que permiten mejorar progresivamente el producto (Flores-Cerna et al., 2021).

2.2.1.1. Ingeniería de software

Es una disciplina que integra técnicas, herramientas y métodos con el propósito de desarrollar sistemas de alta calidad, confiables y eficientes (Gómez-Álvarez & Zapata-Jaramillo, 2022). Además, requiere competencias técnicas derivadas de las ciencias de

la computación y habilidades sociales como el trabajo en equipo, comunicación y la toma de decisiones. En consecuencia, esta área busca aplicar principios de ingeniería que permitan obtener software rentable, funcional y adaptable a entornos reales (González et al., 2022).

2.2.1.2. Ciclo de vida del desarrollo de software (SDLC)

El desarrollo de software se organiza en fases que conforman el ciclo de vida, abarca la construcción y mantenimiento del sistema; estas etapas incluyen el análisis, diseño, desarrollo, implementación y mantenimiento, cada una con un propósito dentro del proceso (González et al., 2022).

Fases:

- **Análisis:** identifica requisitos, necesidades del usuario y define el alcance del sistema.
- **Diseño:** establece la estructura del sistema y considera los componentes de software y hardware.
- **Desarrollo:** construcción del sistema mediante programación y modelado de procesos.
- **Implementación:** instalación y puesta en funcionamiento del sistema.
- **Mantenimiento:** corrección de errores, incorporación de mejoras para continuidad y eficiencia del software (González et al., 2022).

Seguridad:

- **Análisis:** identificación de riesgos, requisitos de seguridad y políticas normativas, y modelado de amenazas y protección de datos.
- **Diseño:** definición de la arquitectura de seguridad y se establecen configuraciones seguras.
- **Desarrollo:** aplicación de buenas prácticas de programación, uso de frameworks, gestión de errores y ejecución de pruebas de seguridad.
- **Pruebas:** se evalúa el sistema mediante análisis estático y dinámico del código, además de verificar la transferencia segura de datos.

- **Despliegue:** se implementan configuraciones, controles de acceso, monitoreo, auditorías, actualización de parches y gestión de incidentes (Pelaiz & Delgado, 2024).

2.2.1.3. Metodologías ágiles

Las metodologías ágiles se fundamentan en procesos iterativos y de corta duración, cuyo propósito es entregar incrementos funcionales de software en periodos cortos, aquello contribuye a mejorar la satisfacción del cliente (Morales-Carrillo et al., 2022). Además, este enfoque tiene gran aceptación en la industria de desarrollo de software, dado que simplifica las entregas, reduce la carga administrativa y limita la documentación innecesaria, así se incrementa la eficiencia del proceso (Armijos et al., 2024). En este sentido, una de sus características es la colaboración constante y la comunicación continua entre los integrantes del equipo y las partes interesadas (Feriz et al., 2024).

Por otra parte, las metodologías ágiles surgen como una alternativa para transformar la gestión de proyectos, debido a que aportan flexibilidad, iteración y un enfoque centrado en el cliente. Así, mediante la aplicación de sprints, retrospectivas y trabajo interdisciplinario, los proyectos responden con mayor eficiencia en contextos dinámicos, que se caracterizan por cambios en los requisitos, limitaciones presupuestarias y expectativas crecientes de los usuarios (Feriz et al., 2024). En consecuencia, frente a problemas como requisitos variables, planificación inadecuada o insatisfacción del cliente, estas metodologías facilitan la incorporación de cambios de forma controlada (Bautista-Villegas, 2022).

Entre las más utilizadas se encuentran Scrum, Kanban y eXtreme Programming (XP), que favorecen la eficiencia, la calidad del producto y la mejora continua del proceso:

Scrum

Se enfoca en el trabajo de equipos interdisciplinarios autoorganizados que desarrollan el producto en iteraciones cortas denominadas sprints, con una duración máxima de un mes. Cada sprint integra actividades estructuradas como la planificación, reuniones

diarias, desarrollo, revisión y retrospectiva, esto facilita el seguimiento y la mejora continua del proyecto (Armijos et al., 2024). Además, esta metodología se emplea cuando los requisitos no están completamente definidos, ya que promueve la flexibilidad, la innovación y la entrega rápida de valor (Bravo-Huivin et al., 2022). En este sentido, el sprint se concibe como un periodo de trabajo en el que el equipo se compromete a generar resultados funcionales (Feriz et al., 2024). Asimismo, Scrum establece roles definidos que organizan el trabajo del equipo:

- *Product Owner (Dueño del producto)*: Representa a los interesados del proyecto y se encarga de maximizar el valor del producto mediante la gestión y priorización del Product Backlog (Sassa et al., 2023).
- *Scrum Master (Facilitador)*: Garantiza la correcta aplicación de la metodología, promueve la colaboración y elimina impedimentos, sin intervenir directamente en el desarrollo (Haro et al., 2025)
- *Equipo de Desarrollo*: Transforma los requerimientos en incrementos funcionales del producto; se caracteriza por su estructura horizontal y un tamaño óptimo de entre tres y nueve integrantes (Sassa et al., 2023).

Kanban

Se orienta a la gestión visual del trabajo mediante tableros organizados en columnas que representan los estados de las tareas, como “pendiente”, “en progreso” y “finalizado” (Ozkan et al., 2022). Esta metodología basada en principios Lean, busca optimizar el flujo de trabajo al limitar las tareas en curso, permite identificar y eliminar cuellos de botella (Castillo & Guaña, 2024). Sus principios incluyen: visualización del flujo de trabajo, limitación del trabajo en curso, identificación y eliminación de cuellos de botella (Gaete et al., 2021).

Extreming Programming (XP)

Se centra en mejorar la calidad del software mediante prácticas intensivas que abordan la complejidad de los requisitos a lo largo del desarrollo; esta metodología plantea llevar las buenas prácticas al máximo nivel, con el fin de adaptarse a cambios sin afectar

el tiempo ni el presupuesto (Bautista-Villegas, 2022). Entre sus principios más relevantes se destacan:

- *Pruebas constantes*: Desarrollo basado en pruebas unitarias desde etapas tempranas.
- *Programación en parejas*: Trabajo colaborativo entre dos desarrolladores sobre el mismo código.
- *Retroalimentación continua*: Entregas frecuentes que permiten validar y mejorar el producto.

En consecuencia, esta metodología promueve un proceso de desarrollo basado en prácticas como diseño simple, refactorización continua y entregas iterativas, para contribuir a obtener softwares de alta calidad (Armijos et al., 2024).

2.2.1.4. Beneficios de la metodología ágil

Las metodologías ágiles se caracterizan por emplear ciclos de vida adaptativos y estructurados en iteraciones de tiempo definido que funcionan como mini proyectos, dado que este modelo no concentra todo el desarrollo en un único ciclo, sino que lo divide en fases incrementales; por lo tanto, facilita la incorporación de cambios sin necesidad de replantear el proyecto en su totalidad (Guzmán et al., 2024).

En este sentido, cada iteración cumple un rol fundamental, ya que exige la entrega de un incremento funcional del producto; además, estos bloques de tiempo se planifican, ejecutan y evalúan de forma continua, así permite mantener un flujo constante de desarrollo y mejora (Paredes & Páez, 2024). Asimismo, las metodologías ágiles ofrecen múltiples beneficios que fortalecen la gestión de proyectos de software, entre los que destacan la flexibilidad, la mejora continua, la comunicación efectiva y la entrega progresiva de valor (Guzmán et al., 2024). A continuación, se sintetizan sus principales ventajas:

Tabla 1.

Beneficios de la metodología ágil

Metodología	Beneficios principales
SCRUM	Fomenta la comunicación, trabajo colaborativo y permite adaptarse a cambios en los requisitos, tal que reduce el riesgo del proyecto; por consiguiente, mejora la transparencia mediante artefactos como el Product Backlog y Sprint Backlog, y promueve la mejora continua a través de retrospectivas (Armijos et al., 2024).
Extreme Programming (XP)	Prioriza la calidad del software, ya que asegura entregas frecuentes con estándares funcionales, en este sentido, fortalece el trabajo en equipo mediante prácticas como la programación en parejas y facilita la adaptación a cambios (Guanoluisa et al., 2024).
Kanban	Optimiza el flujo de trabajo mediante la visualización de tareas y la limitación del trabajo en curso; además, permite reorganizar prioridades con facilidad. Por lo tanto, mejora la comunicación del equipo y se adapta a distintos tipos de proyectos gracias a su flexibilidad (Gaete et al., 2021).

Nota. Se presentan los beneficios de las metodologías ágiles más utilizadas.

2.2.1.5. Arquitectura cliente-servidor

La arquitectura cliente-servidor constituye un modelo fundamental en el desarrollo de aplicaciones web, ya que organiza el sistema en dos componentes que interactúan de forma coordinada. Por un lado, el cliente representa el punto de acceso del usuario, desde donde se generan las solicitudes mediante navegadores web, aplicaciones de mensajería o clientes de correo. Por otro lado, el servidor recibe las solicitudes, las procesa, gestiona la información requerida y devuelve una respuesta; además, tiene la capacidad de atender múltiples peticiones de manera continua (Rana & Saleh, 2022). En este sentido, esta arquitectura ofrece ventajas para el desarrollo de sistemas web, entre las cuales destacan:

- **Mantenimiento eficiente:** la distribución de funciones en sistemas independientes permite realizar modificaciones en el cliente sin afectar al servidor (Rodríguez et al., 2026).
- **Alta escalabilidad:** facilita la adaptación del sistema ante el crecimiento de usuarios o requerimientos tecnológicos (Rodríguez et al., 2026).
- **Centralización de datos:** permite que múltiples clientes accedan a la información desde diferentes ubicaciones (Rana & Saleh, 2022).

2.2.1.6. Arquitectura de software (MVC)

La arquitectura de software define las estructuras necesarias para el diseño y funcionamiento de un sistema, se considera los requisitos, diseño arquitectónico y organización de sus componentes (García, 2025). El modelo Modelo–Vista–Controlador (MVC) mejora la organización del software ya que separa responsabilidades y facilita su mantenimiento (Bautista-Villegas, 2022). El modelo se encarga de gestionar y almacenar la información del sistema; vista se ocupa de presentar los datos al usuario final de manera comprensible y controlador coordina la lógica de aplicación y gestiona la interacción entre modelo y vista (Enríquez et al., 2023).

Tabla 2.

Beneficios y limitaciones de MVC

Ventajas	Desventajas
Reutilización del código: La separación entre lógica de negocio y presentación permite reutilizar componentes en diferentes partes del sistema.	Complejidad: Requiere mayor conocimiento técnico para su correcta implementación.
Amplia adopción: Su uso extendido en diversas aplicaciones facilita el soporte y la disponibilidad de recursos.	Mayor número de archivos: La separación de componentes incrementa la estructura del proyecto.

Facilidad de pruebas: La división de responsabilidades simplifica la validación del sistema.	Necesidad de diseño previo: Exige una planificación arquitectónica adecuada desde el inicio
--	---

Mantenibilidad: Permite realizar mejoras y actualizaciones de forma más organizada.

Fuente. (Yunanto et al., 2022)

2.2.2. Aplicaciones web

Las aplicaciones web constituyen un tipo de software que se ejecuta a través de navegadores y se encuentra alojado en servidores web, de esta manera permite a los usuarios acceder a sus funcionalidades desde cualquier ubicación mediante una conexión de red. En este contexto, su arquitectura se compone de tres elementos: el servidor web, la red de comunicación y los clientes o navegadores, los cuales interactúan para procesar y visualizar la información (Espíndola et al., 2024).

En este sentido, el servidor web gestiona las solicitudes realizadas por los usuarios mediante el protocolo HTTP, localiza los recursos requeridos en su sistema y envía las respuestas correspondientes al navegador. Además, estas aplicaciones se estructuran bajo modelos de capas, mismas que permiten organizar el procesamiento de la información de forma eficiente (Rodríguez et al., 2026).

Por un lado, el modelo de dos capas conecta directamente la interfaz con la gestión de datos; y el modelo de n-capas, especialmente el de tres capas, incorpora niveles adicionales que mejoran la escalabilidad, mantenibilidad y organización del sistema (Rodríguez et al., 2026). En consecuencia, este tipo de arquitectura favorece el desarrollo de soluciones robustas y accesibles en entornos web (Quezada-Sarmiento et al., 2023).

2.2.2.1. APIs RESTful

Las APIs RESTful representan un enfoque arquitectónico adoptado para el diseño de servicios web, ya que permiten la interoperabilidad entre sistemas distribuidos dentro de la World Wide Web. Este modelo se caracteriza por su naturaleza sin estado y su

capacidad para operar en múltiples plataformas, lo que facilita la integración de aplicaciones en diversos entornos (Ehsan et al., 2022).

Estas interfaces se estructuran por puntos finales que simbolizan funcionalidades dentro de un proceso de negocio; dichos puntos son accesibles a través del protocolo HTTP, con métodos estándar como GET, POST, PUT y DELETE, los cuales permiten realizar operaciones de creación, consulta, actualización y eliminación de datos. Asimismo, cada recurso se identifica mediante una URI, para garantizar una comunicación clara y estructurada entre cliente y servidor (Martin-Lopez et al., 2022). Por consiguiente, las APIs RESTful son un componente fundamental en el desarrollo de servicios web, en especial en arquitecturas basadas en microservicios (Golmohammadi et al., 2024).

2.2.2.2. Seguridad en aplicaciones web

La seguridad en aplicaciones web es esencial para proteger la información y garantizar la confiabilidad de los sistemas; el enfoque de defensa en profundidad propone la implementación de múltiples controles de seguridad distribuidos en diferentes capas, con el propósito de prevenir, detectar y mitigar amenazas de manera integral. Este enfoque considera la aplicación, entorno y componentes asociados para una gestión efectiva de los riesgos (Quezada-Sarmiento et al., 2023; Reyes et al., 2022).

La encriptación sirve para la protección y confidencialidad de datos en reposo almacenada en servidores o en tránsito a través de redes; su implementación se recomienda a quienes utilizan servicios en la nube o intercambian información sensible (Espinoza-Marca & Cieza-Mostacero, 2024).

El análisis de vulnerabilidades identifica y evalúa debilidades en sistemas y redes antes de que sean explotadas para fortalecer la seguridad de la información. Este proceso incluye escaneo de puertos, análisis de código y revisión de configuraciones (Molina, 2026). Las pruebas de intrusión evalúan la resistencia del sistema frente a ataques simulados para la detección de fallos y generación de evidencias para auditorías de seguridad (Cruz et al., 2022).

Análisis personal:

El desarrollo de software integra principios de la ingeniería de software para construir sistemas que resuelvan necesidades concretas; todo el proceso se organiza mediante el ciclo de vida del software y metodologías ágiles que permiten adaptarse a cambios y entregar valor de forma continua. Arquitecturas como cliente-servidor y patrón MVC ayudan a que las aplicaciones queden mejor organizadas y sean fáciles de mantener y escalar. Las aplicaciones web son usadas por su accesibilidad e integración mediante APIs RESTful, su efectividad depende de la implementación de medidas de seguridad que garanticen la protección de información y su correcto funcionamiento.

2.3. Conceptual

2.3.1. Conceptos de sistemas y tecnología web

- **Base de datos:** Una base de datos se define como un conjunto organizado de información almacenada de forma estructurada, lo que permite su acceso rápido y eficiente. Su diseño responde a las necesidades de información de una organización, por lo que facilita la gestión, almacenamiento y recuperación de datos relacionados (Araujo-Inastrilla et al., 2023). En este sentido, la correcta estructuración de la información constituye el elemento esencial que da origen a una base de datos funcional (Pacheco et al., 2023).
- **Gestor de bases de datos:** El gestor de bases de datos es una aplicación que permite definir, crear, administrar y controlar el acceso a la información almacenada (Araujo-Inastrilla et al., 2023). Además, simplifica la complejidad asociada a la gestión de grandes volúmenes de datos, siendo fundamental cuando la información crece de forma acelerada y requiere procesamiento eficiente (J. Pacheco et al., 2023).
- **Gestión de la información:** La gestión de la información representa un proceso estratégico que permite recopilar, analizar y utilizar datos con el fin de comprender el comportamiento pasado y proyectar escenarios futuros. Por lo tanto, facilita la toma de decisiones y el diseño de estrategias organizacionales; asimismo, la integración de distintos tipos de datos fortalece la eficiencia de los procesos y

contribuye al cumplimiento de los objetivos institucionales (Garcés-Giraldo et al., 2022).

- **Hardware:** El hardware comprende el conjunto de componentes físicos de un sistema informático que hacen posible el procesamiento de la información, entre ellos se incluye dispositivos de entrada, salida, almacenamiento y procesamiento, los cuales ejecutan las instrucciones definidas por el usuario. En consecuencia, constituye la base material sobre la que operan los sistemas tecnológicos y permite el funcionamiento de cualquier equipo informático (Monterrubio-Hernández, 2024).
- **Lenguaje SQL:** Constituye el estándar para la definición, manipulación y control de bases de datos relacionales, se caracteriza por su naturaleza declarativa, que implica que el usuario especifica las acciones a realizar sin detallar el procedimiento interno.(Díaz et al., 2023). Asimismo, su sintaxis, cercana al lenguaje natural, facilita su uso y por tanto, se convierte en una herramienta esencial para el desarrollo de sistemas de información y el acceso a diversos sistemas de gestión de bases de datos (Jones et al., 2026).
- **Servidor web:** El servidor web es un componente fundamental en la arquitectura de las aplicaciones, ya que se encarga de distribuir páginas de información a los clientes que las solicitan. Este proceso se realiza a través de conexiones de red mediante el protocolo HTTP; en este sentido, cuando el navegador envía una solicitud, el servidor localiza el recurso en su sistema de archivos y lo devuelve al cliente para su visualización (Espíndola et al., 2024).
- **Sistema tecnológico:** Un sistema tecnológico se define como un conjunto organizado de recursos que permite optimizar procesos y reducir ineficiencias operativas; además, contribuye a eliminar cuellos de botella, retrasos y problemas asociados a la falta de información (Salcedo et al., 2023).
- **Software:** El software engloba los elementos lógicos e intangibles que permiten el funcionamiento de los dispositivos informáticos; esto incluye sistemas operativos, aplicaciones y programas que proporcionan las instrucciones necesarias para que el hardware opere correctamente. Además también actúa como un mediador del

conocimiento, ya que puede facilitar o limitar los procesos según su diseño y contexto de uso (Pacheco et al., 2022).

2.3.2. Desarrollo de software y arquitectura

- **API (Interfaz de Programación de Aplicaciones):** Las API constituyen mecanismos que permiten la comunicación e integración entre diferentes sistemas o aplicaciones. Las API establecen estándares que facilitan la interoperabilidad y la orquestación de servicios, además en organizaciones son una herramienta esencial para la automatización de procesos y la transformación digital, ya que conectan funcionalidades y servicios de manera eficiente (Suescún et al., 2024).
- **Arquitectura cliente-servidor:** La arquitectura cliente-servidor se define como un modelo de computación distribuida en el cual las tareas se dividen entre el cliente, que realiza solicitudes, y el servidor, que procesa dichas solicitudes y responde con la información requerida (Concepción et al., 2022). Este modelo facilita la centralización de recursos, como bases de datos compartidas, y permite escalar la capacidad del sistema mediante la incorporación de nuevos recursos de hardware (Aldalur, 2024).
- **Backend:** El backend abarca los procesos que se ejecutan en el servidor y sostienen la lógica de negocio, gestiona solicitudes de usuarios, procesa información y se comunica con las bases de datos (Molina, 2026). Para su desarrollo se utiliza PHP, Python, Java o .NET y para bases de datos SQL, MySQL o MongoDB (Taj & Farooq, 2023).
- **Framework:** Los frameworks proporcionan herramientas y componentes para enrutamiento, gestión de formularios, autenticación e interacción con bases de datos. Permite optimizar tiempo, costos y promover buenas prácticas de desarrollo; los frameworks más utilizados son Django, Ruby on Rails, Laravel, Express.js y ASP.NET (Quisaguano et al., 2024).
- **Front-End:** El front-end comprende las tecnologías que operan en el lado del cliente y permiten la interacción directa con el usuario a través de navegadores web; estas herramientas transforman la y datos del sistema en interfaces gráficas que

facilita la visualización y manipulación de la información (Aranda, 2018). Entre las tecnologías empleadas destacan HTML, CSS y JavaScript, las cuales posibilitan el diseño de interfaces dinámicas y funcionales (Quisaguano et al., 2024).

- **Laravel:** Laravel es un framework de desarrollo web basado en PHP, creado por Taylor Otwell en 2011, que facilita la construcción de aplicaciones mediante una estructura organizada y eficiente. Este framework adopta el patrón MVC, el cual separa la lógica del negocio, la interfaz de usuario y la gestión de datos, lo que mejora la mantenibilidad del sistema. Asimismo, incorpora herramientas como Artisan para la automatización de tareas, Eloquent ORM para la gestión de bases de datos, un sistema avanzado de rutas, middleware para la seguridad y mecanismos de autenticación y autorización (Varelas & Castro, 2026).
- **Lenguajes de programación:** La interacción entre hardware y software se realiza en código binario que posibilita la ejecución de instrucciones y funcionamiento de la computadora (Molina, 2026). Los lenguajes más utilizados son:

Tabla 3.
Lenguajes de programación

Lenguaje	Características principales
Visual Basic .NET	Facilita el desarrollo de aplicaciones web y servicios en la plataforma .NET.
Visual C# .NET	Permite el desarrollo de aplicaciones a través de .NET Framework.
Visual C++ .NET	Se orienta a aplicaciones de alto rendimiento, con acceso a bibliotecas avanzadas.
Transact-SQL	Gestiona bases de datos relacionales en SQL Server.
Java	Se orienta a objetos independiente de la plataforma y adaptable a diversos entornos.

Fuente. (Molina, 2026).

2.3.3. Interacción del sistema

- **Experiencia de usuario (UX):** La experiencia de usuario se construye a partir de las interacciones que el usuario establece con el sistema, las cuales pueden registrarse y analizarse para generar conjuntos de datos que permitan identificar patrones de comportamiento. En este contexto, el uso de técnicas de machine learning facilita el reconocimiento de necesidades específicas y el diseño de interfaces más intuitivas y adaptativas (Quispe, 2022).

El machine learning contribuye a optimizar la experiencia del usuario al reducir el esfuerzo de navegación y mejorar la interacción con el sistema; para ello, es necesario que las interfaces se diseñen de manera que permitan el seguimiento de las acciones del usuario, con el fin de utilizar dicha información como base para la adaptación dinámica del sistema (Ebenezer et al., 2022).

- **Interfaz de usuario (UI):** La interfaz de usuario es esencial en la interacción entre sistema-usuario, su diseño debe garantizar flexibilidad y adaptabilidad; la recopilación y análisis de datos permiten establecer ciclos de retroalimentación que facilitan el desarrollo de interfaces ajustadas al usuario, se incluye factores culturales y sociales (Quispe, 2022).

Las interfaces tradicionales basadas en el ingreso de texto o comandos generan limitaciones relacionadas con el idioma, cuando los usuarios no dominan el alfabeto o poseen dificultades de alfabetización. Por lo tanto, surgen alternativas como interfaces libres de texto que amplían las formas de interacción mediante el uso de imágenes, símbolos y otros recursos visuales (Ilyas et al., 2022).

2.3.4. Gestión de procesos en el sistema

- **Trazabilidad de procesos:** La trazabilidad es la capacidad de documentar y seguir cada etapa del ciclo de vida, desde la definición de los requerimientos hasta la implementación y el mantenimiento del sistema. En este sentido, permite obtener una visión clara y estructurada de la evolución del software, esto facilita la identificación de inconsistencias y asegura la alineación entre los objetivos del sistema y los requerimientos establecidos (Pocladova et al., 2024).

Además, esta capacidad constituye un elemento esencial dentro de la ingeniería de software, ya que ampara diversos procesos como el análisis del impacto ante cambios, verificación y validación del sistema, así como la evaluación del cumplimiento de los requisitos y la garantía de calidad. Por lo tanto, la trazabilidad contribuye a gestionar la complejidad propia de los sistemas modernos, al permitir una navegación estructurada entre sus componentes (Fucci et al., 2022).

Sin embargo, la ausencia de vínculos de trazabilidad, especialmente entre elementos como los requisitos y los casos de prueba, puede afectar de forma significativa la calidad del software; en consecuencia, esta limitación dificulta la detección de errores y retrasa la mejora continua del sistema (Hossman et al., 2024).

2.4. Legal

Ley Orgánica de Protección de Datos Personales

En esta ley se establece los lineamientos fundamentales para la gestión y protección de datos personales, cuando los sistemas implican tratamiento de información sensible:

Art. 42.-Evaluación de impacto del tratamiento de datos personales.- “El responsable realizará una evaluación de impacto del tratamiento de datos personales cuando se haya identificado la probabilidad de que dicho tratamiento [...] conlleve un alto riesgo para los derechos y libertades del titular [...]. La evaluación de impacto deberá efectuarse previo al inicio del tratamiento de datos personales” (Ley Orgánica de Protección de Datos Personales, 2018, p. 21).

Art. 43.-Notificación de vulneración de seguridad.- “El responsable del tratamiento deberá notificar la vulneración de la seguridad de datos personales [...] a más tardar en el término de cinco (5) días [...]. El encargado del tratamiento deberá notificar [...] dentro del término de dos (2) días” (Ley Orgánica de Protección de Datos Personales, 2018, p. 22).

Art. 51.-Registro Nacional de protección de datos personales.- “El responsable del tratamiento de datos personales deberá reportar y mantener

actualizada la información [...] sobre la identificación de la base de datos, finalidad del tratamiento, naturaleza de los datos, destinatarios, medidas de seguridad y tiempo de conservación” (Ley Orgánica de Protección de Datos Personales, 2018, p. 25).

Código Orgánico de la Economía Social de los Conocimientos, Creatividad e Innovación (INGENIOS)

Este código establece el marco normativo que regula la protección, gestión y difusión del conocimiento en Ecuador.

Artículo 85.- Derechos intelectuales.- se establece que los derechos intelectuales se protegen en todas sus formas, abarca la propiedad intelectual, los conocimientos tradicionales y se orientan a una adecuada gestión del conocimiento, con el propósito de impulsar el desarrollo científico, tecnológico, artístico y cultural (Código Orgánico de La Economía Social de Los Conocimientos, Creatividad e Innovación, 2016, p. 22).

Artículo 87.- De la adquisición y ejercicio de los derechos de la propiedad intelectual.- establece que estos derechos deben mantenerse en equilibrio con otros derechos fundamentales; en consecuencia, su gestión se orienta a promover la innovación social y la difusión del conocimiento, con el fin de generar beneficios compartidos y contribuir al bienestar social y económico (Código Orgánico de La Economía Social de Los Conocimientos, Creatividad e Innovación, 2016, p. 22).

Artículo 104.- Obras susceptibles de protección.- las obras originales de carácter literario, artístico y científico son susceptibles de protección siempre que puedan reproducirse o difundirse por cualquier medio. Dentro de estas se incluye el software (Código Orgánico de La Economía Social de Los Conocimientos, Creatividad e Innovación, 2016, p. 24).

2.5. Georeferencial

Cantón: Guaranda

Provincia: Bolívar

Dirección: Parque Industrial, Guaranda Negroyaco.

Figura 1.

Mapa Geográfico de la ubicación del Centro de Atención Emergente de Fauna Urbana CAEFU en Guaranda



CAPÍTULO III

METODOLOGÍA

3.1. Metodología de Desarrollo de Software

Con el propósito de alcanzar el objetivo de desarrollar un sistema web para la gestión y adopción de mascotas en el refugio de animales de Guaranda, se adoptó la metodología ágil XP (Extreme Programming). Esta elección responde a la naturaleza del proyecto, en el que los requerimientos pueden evolucionar a lo largo del desarrollo y se requiere mantener la calidad del software (Bautista-Villegas, 2022). En el presente estudio, esta metodología facilitó una comunicación continua con el responsable del centro, siendo esencial para levantar, validar y ajustar los requerimientos funcionales conforme avanza el proceso de diseño del sistema

Como complemento, se empleó la metodología Kanban, que permitió organizar y controlar visualmente el flujo de trabajo mediante la distribución de tareas en distintos estados de avance, esta herramienta facilitó la detección oportuna de bloqueos y la optimización continua del proceso de desarrollo (Gaete et al., 2021), así se contribuyó a la planificación y seguimiento de las actividades definidas para el diseño de la arquitectura del sistema, modelación de base de datos y ejecución de pruebas de funcionamiento.

Para cumplir con el diagnóstico de la situación actual de los procesos de gestión y adopción de mascotas, se recopiló información mediante observación directa y entrevistas al administrador del CAEFU y director de Gestión Ambiental del GAD municipal. Esto permitió identificar las necesidades funcionales del sistema, entre las que se destacan:

- Registro de mascotas con información sobre sus características, historial médico, edad, especie y género.

- Administración de usuarios registrados en la plataforma, con datos de identificación, contacto y autenticación.
- Gestión de solicitudes de adopción, se incluye la revisión, aprobación y rechazo de las mismas.
- Control del estado de las mascotas, diferenciado entre aquellas disponibles para adopción y las ya adoptadas.
- Administración de la información del refugio y publicación de mascotas en el catálogo en línea.
- Generación de reportes estadísticos sobre el total de mascotas, adopciones por mes, solicitudes pendientes y usuarios registrados.

El uso combinado de estas metodologías permitió desarrollar un sistema de forma ágil, colaborativa y organizada, de tal manera que se garantiza que el producto final sea funcional, seguro, escalable y fácil de mantener.

3.2. Fases del proceso de desarrollo

3.2.1. Planificación

A partir de la información recopilada mediante entrevistas y observación del manejo del CAEFU, se definió el alcance del sistema y se elaboraron las historias de usuario, las cuales representan los requerimientos funcionales desde la perspectiva de los distintos actores del sistema: el administrador del centro y los usuarios adoptantes.

Durante esta etapa, se elaboró un Release Plan que organizó las funcionalidades en iteraciones cortas, donde se prioriza las de mayor valor para el funcionamiento del sistema, como el registro de mascotas, la visualización del catálogo de animales disponibles y la gestión de solicitudes de adopción. Para cada iteración, se consideró los siguientes elementos

- **Historias de usuario:** Basadas en descripciones de las necesidades del refugio y registro de usuarios (nombres y apellidos, cédula de identidad, número de teléfono, fecha de nacimiento, correo electrónico y contraseña). Las solicitudes de adopción incluirán información sobre el domicilio del solicitante.

- **Prioridad y valor funcional:** Se prioriza las funcionalidades que aportan mayor beneficio al proceso de gestión y adopción, tales como el registro de usuarios, la administración de mascotas y la aprobación de solicitudes.
- **División en iteraciones:** el proyecto se organizó en iteraciones de una a dos semanas, lo que permitió realizar entregas incrementales y obtener retroalimentación continua por parte del personal del CAEFU.

3.2.2. *Diseño*

En esta fase se diseñó la arquitectura del sistema web, misma que incluyó la estructura de la base de datos y la interfaz de usuario, además de modelar la base de datos para la gestión de solicitudes de adopción y el seguimiento post-adopción. Se aplicó los principios de simplicidad, organización y facilidad de mantenimiento.

Patrón Arquitectónico: El sistema adoptó una arquitectura N-Capas, que distribuyó las responsabilidades en niveles diferenciados para garantizar una estructura organizada, escalable y de fácil mantenimiento, las capas utilizadas son:

- *Capa de presentación:* encargada de la interacción entre el usuario y el sistema a través de una interfaz web intuitiva y accesible, que incluyó una página de inicio con la historia del centro, un catálogo de mascotas disponibles (con fotografía, especie, tamaño, edad, género, cirugías, vacunas y desparasitaciones), un apartado de testimonios de usuarios registrados y un formulario de registro de cuentas.
- *Capa lógica de negocio:* donde se implementó los procesos relacionados con la gestión de mascotas, la validación de solicitudes de adopción, la autenticación de usuarios y la administración del sistema.
- *Capa de datos:* responsable del almacenamiento y gestión de la información relativa a usuarios, mascotas, solicitudes y reportes en la base de datos.

Vista Arquitectónica: Para representar la estructura del sistema desde distintas perspectivas, se utilizó las siguientes vistas arquitectónicas.

- Vista lógica para el funcionamiento interno del sistema con representación de diagramas de clases, secuencia y comunicación.

- Vista de procesos para representar el flujo de actividades del proceso de adopción y administración del sistema.
- Vista física para evidenciar la infraestructura tecnológica como servidor, base de datos y equipos de acceso.
- Vista de despliegue para representar la distribución de componentes y módulos del software.
- Vista de escenarios (+1) para describir los casos de principales entre los actores del sistema, administrador y usuario adoptante.

Codificación: Durante esta fase se implementó las funcionalidades definidas en las historias de usuario, se aplicó las prácticas de XP y el tablero Kanban para mantener un flujo de trabajo organizado. Las prácticas aplicadas fueron las siguientes

- *Desarrollo guiado por pruebas (TDD):* antes de implementar determinadas funcionalidades se definió las pruebas de validación de comportamiento esperado.
- *Integración continua:* cada funcionalidad nueva se integró al sistema por parte y con ejecución de pruebas automáticas para prevenir errores o choques entre módulos.
- *Refactorización:* el código se revisaba y ajustaba constantemente para que la estructura estuviese limpia, coherente y escalable.

El desarrollo se organizará en cuatro iteraciones principales:

- Iteración 1: módulo de usuarios y autenticación.
- Iteración 2: módulo de registro y gestión de mascotas.
- Iteración 3: módulo de solicitudes de adopción.
- Iteración 4: módulo de reportes y administración del sistema.

Pruebas: En esta fase se realizó las pruebas del sistema web que avalaron su funcionamiento, éstas se ejecutaron de forma continua durante el desarrollo, para detectar y corregir errores en cada módulo implementado. Los tipos de prueba aplicados fueron:

- *Pruebas unitarias*: verificaron el comportamiento individual de las funciones y componentes del sistema.
- *Pruebas de integración*: comprobaron la correcta interacción entre los distintos módulos del sistema.
- *Pruebas de aceptación*: se realizó en conjunto con el administrador del CAEFU para validar que el sistema cumpla con los requerimientos establecidos en las historias de usuario.
- *Pruebas de regresión*: garantizó que la incorporación de nuevas funcionalidades no afecte los procesos previamente implementados

El tablero Kanban facilitó la identificación oportuna de tareas bloqueadas en esta fase y contribuye a mantener un flujo de desarrollo continuo y controlado.

Entrega: La entrega se realizó al administrador del CAEFU y se puso a disposición todas las funcionalidades desarrolladas y se recogió retroalimentación para la planificación de futuras mejoras. El tablero Kanban permite visualizar el estado de las funcionalidades entregadas, pendientes y mejoras proyectadas, para facilitar la toma de decisiones y la trazabilidad del proyecto

3.3. Técnicas e Instrumentos de Recopilación de Datos

La recopilación de información se realizó a través de entrevistas y observación directa para obtener información sobre el funcionamiento actual del centro CAEFU.

3.3.1. Técnicas

Entrevista: La entrevista estructurada se caracteriza por utilizar preguntas definidas con anterioridad (Haro et al., 2024). Fue aplicada al administrador responsable de la gestión del CAEFU y al director de Gestión Ambiental del GAD de Guaranda, la información obtenida identificó las dificultades presentes en los procesos de registro, seguimiento y adopción de mascotas, así como establecer los requerimientos funcionales y prioridades para el desarrollo del sistema web.

Observación: La observación permite registrar comportamientos, actitudes y situaciones dentro de un contexto específico, con el propósito de comprender el

fenómeno investigado (Ordoñez-Pacheco, 2025). Esta técnica se aplicó durante las visitas al refugio, donde se observó los procesos relacionados con el registro de mascotas, control sanitario, gestión de solicitudes de adopción, seguimiento de animales adoptados y administración de la información institucional. La información recopilada permitió contrastar los datos obtenidos en las entrevistas y conocer de manera objetiva el flujo de trabajo existente.

3.3.2. Instrumentos

Los instrumentos que se utilizarán para registrar y sistematizar la información recopilada son:

Guía de entrevista: Consistió en un documento estructurado con preguntas orientadas a identificar los requerimientos funcionales del sistema, las necesidades institucionales y las problemáticas existentes en los procesos de gestión y adopción de mascotas del CAEFU. Este instrumento fue aplicado al administrador del centro y al director de Gestión Ambiental (Anexo 4).

Guía de observación: Fue un instrumento diseñado para registrar de forma sistemática las actividades, procedimientos y recursos involucrados en la gestión del refugio. Permitted documentar aspectos relacionados con el manejo de la información de las mascotas, el proceso de adopción y las actividades administrativas realizadas por el personal del centro (Anexo 5).

Procesamiento de la información: La información obtenida de las entrevistas fue transcrita y organizada según las categorías de análisis definidas en los objetivos de la investigación. Posterior, se realizó un análisis descriptivo de las respuestas para identificar necesidades, problemas y requerimientos funcionales del sistema.

Los datos recopilados mediante la observación directa fueron sistematizados para contrastar la información obtenida con los resultados de las entrevistas. Finalmente, los hallazgos fueron utilizados como base para la elaboración de las historias de usuario, el diseño de la arquitectura del sistema y la definición de las funcionalidades implementadas en el software.

CAPÍTULO IV

INGENIERÍA DEL PROYECTO

4.1. Planificación

4.1.1 Tabulación y análisis de datos

La guía de observación permitió identificar los procedimientos empleados por el Centro de Atención Emergente de Fauna Urbana (CAEFU) para el registro de mascotas, gestión de adopciones, administración de la información y uso de recursos tecnológicos, con base en estos hallazgos se constituye la base para la definición de los requerimientos funcionales del sistema web propuesto.

Tabla 4.

Resultados generales de la guía de observación aplicada al CAEFU

Bloque	Sí	No
Registro de mascotas	5	1
Proceso de adopción	6	0
Administración de la información	4	2
Recursos tecnológicos y operativos	3	2
Total	18 (78,26%)	5 (21,74%)

Nota. El porcentaje representa los aspectos identificados y no identificados.

Fueron 23 aspectos observados, de los cuales 18 (78,26 %) se encuentran dentro de los procesos que actualmente ejecuta el CAEFU, los 5 restantes (21,74 %) no se verificaron. Con base en esta observación se constató la existencia de procedimientos definidos para el registro de mascotas, la gestión de adopciones y el uso de recursos tecnológicos. Los ausentes corresponden a la digitalización de la información, difusión de mascotas y registro de ciudadanos interesados en adoptar.

Tabla 5.

Hallazgos y requerimientos identificados con la guía de observación

Bloque	Hallazgos observados
Registro de mascotas	Uso de fichas físicas para historial médico, fotografías y clasificación por estado de adopción.
Proceso de adopción	Solicitudes físicas, validación de datos, aprobación y seguimiento.
Administración de la información	No existe difusión digital y registro de interesados.
Recursos tecnológicos	Factibilidad técnica para implementar el sistema web.

Nota. La tabla resume los hallazgos obtenidos durante la observación.

Los resultados evidencian que el CAEFU dispone de procesos establecidos para el manejo de mascotas y la gestión de adopciones; sin embargo la información se administra mediante registros físicos. Existe ausencia de mecanismos digitales para la difusión de mascotas y registro de adoptantes.

4.1.2. Análisis de la entrevista

La entrevista realizada al técnico del CAEFU permitió conocer los procedimientos utilizados para el registro de mascotas, gestión de adopciones y administración de la información institucional, así como identificar las necesidades que deberán considerarse en el desarrollo del sistema web.

Tabla 6.

Síntesis de resultados de la entrevista aplicada al técnico del CAEFU

Categoría	Hallazgo identificado
Registro de mascotas	Uso de fichas individuales con fotografía y registro del estado de ingreso.
Control de mascotas	Existencia de historias clínicas y seguimiento a mascotas adoptadas.

Proceso de adopción		A través de una solicitud, validación documental y aprobación por parte del personal.
Requisitos del adoptante		Es necesario la presentación de la cédula, servicio básico, tiempo y un espacio adecuado.
Seguimiento post-adopción		Se realiza una visita para ver el estado de las mascotas adoptadas.
Limitaciones identificadas		Falta de compromiso ciudadano con el bienestar animal.
Información para la plataforma		Publicación de ordenanzas y sanciones relacionadas con el maltrato animal.
Reportes requeridos		Registro de adoptantes y estadísticas de la población de animales de compañía.
Recursos tecnológicos		Existencia de personal administrativo con conocimientos informáticos.

Nota. Información obtenida mediante la entrevista aplicada al técnico del CAEFU de Guaranda.

Se evidencia que el CAEFU dispone de procedimientos definidos para el registro, seguimiento de mascotas, gestión de adopciones; está presente la necesidad de centralizar la información, administrar registros de adoptantes y difundir información institucional mediante medios digitales.

4.1.3 Historias de usuario

A continuación se presentan las historias de usuario que describen una funcionalidad del sistema desde la perspectiva de los actores involucrados, estas historias guiarán el proceso de desarrollo y validación.

Tabla 7.*Historia de usuario N° HU-01 — Registro de mascota en el sistema*

Historia de usuario			
Número:	HU-01	Usuario:	Administrador
Nombre de historia:	Registro de mascota en el sistema		
Prioridad:	Alta	Riesgo:	Bajo
Programador:	Muyulema Crisrofer		
Precondiciones:	El administrador debe haber iniciado sesión en el sistema con credenciales válidas.		
Descripción:	El administrador puede registrar una nueva mascota con los siguientes datos: nombre, especie, raza, edad, género, tamaño, estado de salud, vacunas, desparasitaciones, cirugías previas y fotografía. Queda como disponible en el catálogo al guardar la información.		
Criterios de aceptación:	1. El sistema valida que los campos obligatorios estén completos. 2. Carga correcta de la fotografía. 3. La mascota aparece en el catálogo como 'Disponible'. 4. Se muestra un mensaje de confirmación al administrador.		

Tabla 8.*Historia de usuario N° HU-02 — Registro de cuenta de usuario*

Historia de usuario			
Número:	HU-02	Usuario:	Usuario adoptante
Nombre de historia:	Registro de cuenta de usuario		
Prioridad:	Alta	Riesgo:	Bajo
Programador:	Muyulema Crisrofer		
Precondiciones:	El usuario no debe tener una cuenta registrada previamente con el mismo correo o cédula.		
Descripción:	Para crear una cuenta debe completar el formulario con: nombres, apellidos, cédula, teléfono, fecha de nacimiento, correo electrónico y		

		contraseña. La contraseña debe tener 8 caracteres entre ellos mayúsculas, números y un carácter especial.
Criterios de aceptación:	de	1. El sistema valida el formato de la cédula y del correo electrónico. 2. Se verifica que la contraseña cumpla los requisitos de seguridad. 3. No se permite el registro con un correo o cédula ya existente. 4. El sistema envía confirmación de registro al correo proporcionado.

Tabla 9.

Historia de usuario N° HU-03 — Visualización del catálogo de mascotas

Historia de usuario			
Número:	HU-03	Usuario:	Usuario adoptante
Nombre de historia:	Visualización del catálogo de mascotas		
Prioridad:	Alta	Riesgo:	Bajo
Programador:	Muyulema Crisrofer		
Precondiciones:	Deben existir mascotas registradas con estado 'Disponible' en el sistema.		
Descripción:	El catálogo de mascotas disponibles cuenta con filtros por especie, género y edad. El usuario puede visualizar el catálogo donde cada tarjeta muestra la información de la mascota.		
Criterios de aceptación:	de	1. Para mostrar una mascota debe tener toda la información visible. 2. Los filtros funcionan correctamente. 3. Solo se presentan mascotas en estado 'Disponible'. 4. La interfaz es responsiva y accesible desde computadoras y dispositivos móviles.	

Tabla 10.*Historia de usuario N° HU-04 — Envío de solicitud de adopción*

Historia de usuario			
Número:	HU-04	Usuario:	Usuario adoptante
Nombre de historia:	Envío de solicitud de adopción		
Prioridad:	Alta	Riesgo:	Medio
Programador:	Muyulema Crisrofer		
Precondiciones:	El usuario debe haber iniciado sesión. La mascota seleccionada debe estar disponible y sin solicitudes activas del mismo usuario.		
Descripción:	El usuario genera una solicitud llenando un formulario con los siguientes datos: Datos de la vivienda Tipo de vivienda: el solicitante elige entre casa propia, casa arrendada, casa familiar, departamento u otro (una sola opción). Foto del espacio: puede adjuntar una imagen del lugar donde vivirá la mascota, no es obligatorio. Dimensiones del espacio: se registra el largo y el ancho en metros, en campos numéricos. Dedicación diaria: el solicitante indica cuántas horas al día podrá destinarse a la mascota, en números enteros. Alimentación: selecciona si le dará croquetas, comida casera preparada, o una combinación de ambas. Ubicación Calle principal: campo obligatorio, que se puede escribir o buscar en el mapa interactivo. Calle secundaria: este dato es opcional. Parroquia, cantón y provincia: se completan solos cuando el usuario marca el punto en el mapa. Identidad y firma Cédula de identidad: hay que subir la foto del documento; obligatorio. Firma del solicitante: adjuntar una imagen de su firma, opcional.		

Visita y confirmación

Fecha y hora de visita: se elige mediante un selector, para coordinar la verificación domiciliaria.

Términos y condiciones: hay que marcar la casilla de aceptación antes de poder enviar la solicitud; sin esto, el envío no procede.

Una vez enviada, la solicitud queda registrada con estado "Enviado".

Desde su perfil el usuario puede seguir el proceso en un tablero que muestra cuatro etapas: Enviado, En revisión, Visita y Aprobado.

Criterios de aceptación:	de	1. Todos los campos obligatorios deben estar completos 2. Los campos opcionales no bloquean el envío. 3. El mapa rellena automáticamente los campos de ubicación. 4. No se puede enviar más de una solicitud para la misma mascota. 5. El sistema muestra un mensaje de confirmación y redirige al tablero de seguimiento tras el envío exitoso. 6. El tablero de seguimiento refleja en tiempo real las cuatro etapas del proceso. 7. El administrador recibe notificación interna inmediata al registrarse una nueva solicitud. 8. El usuario puede consultar en todo momento el estado actualizado de su solicitud desde su perfil.
---------------------------------	-----------	--

Tabla 11.

Historia de usuario N° HU-05 — Revisión y resolución de solicitudes de adopción

Historia de usuario			
Número:	HU-05	Usuario:	Administrador
Nombre de historia:	Revisión y resolución de solicitudes de adopción		
Prioridad:	Alta	Riesgo:	Medio
Programador:	Muyulema Crisrofer		
Precondiciones:	Deben existir solicitudes en estado 'Pendiente'. El administrador debe tener sesión activa.		
Descripción:	El administrador puede revisar y resolver las solicitudes de adopción pendientes a través de un panel estructurado en tres pestañas:		

Pestaña 1 — Resumen de la solicitud

Vista consolidada con los datos generales del solicitante y de la mascota: nombre, cédula, correo, tipo de vivienda, dimensiones del espacio, horas de dedicación, tipo de alimentación, ubicación del domicilio y fecha de visita agendada.

Pestaña 2 — Solicitud (documentos del solicitante)

Visualización del formulario enviado por el usuario.

Pestaña 3 — Visita: checklist de verificación

Formulario de verificación de la visita domiciliaria. Contiene los siguientes ítems de verificación.

1. El espacio declarado coincide con la realidad.
2. El adoptante comprende sus responsabilidades y compromisos.
3. El tiempo que dedicará diariamente a la mascota es suficiente y realista.
4. La vivienda es adecuada y segura para la futura mascota postulada.
5. Las opciones de alimentación propuestas son viables y adecuadas para.

Resultado y notificación

Una vez completado el checklist, el administrador registra su decisión de Aprobar o Rechazar la solicitud e ingresa observaciones opcionales. Se notifica al usuario a través de WhatsApp con el resultado y los pasos a seguir. Actualiza los estados correspondientes.

Criterios de aceptación:	1. El administrador accede al panel de revisión y visualiza tres pestañas: Resumen, Solicitud y Visita. 2. La pestaña "Solicitud" muestra la cédula, firma, foto del espacio y los datos del formulario enviado. 3. El checklist de visita registra el resultado de cada ítem (Cumple / No cumple) y observaciones por ítem. 4. El estado del proceso de adopción avanza por las etapas: Enviado → En revisión → Visita → Aprobado, el tablero del usuario se actualiza en automático. 5. Aprobado: la mascota cambia a estado "Adoptada" y las otras solicitudes pendientes pasan a "Rechazada".
---------------------------------	---

6. Rechazada: Hay espacio para las observaciones registradas.
7. La notificación se envía vía WhatsApp al usuario con el resultado y con las instrucciones para concretar la adopción.
8. El historial de resolución queda registrado con fecha, hora, resultado del checklist y observaciones del administrador..

Tabla 12.

Historia de usuario N° HU-06 — Generación de reportes estadísticos

Historia de usuario			
Número:	HU-06	Usuario:	Administrador
Nombre de historia:	Generación de reportes estadísticos		
Prioridad:	Media	Riesgo:	Bajo
Programador:	Muyulema Crisrofer		
Precondiciones:	Deben existir datos registrados en el sistema (mascotas, solicitudes, usuarios).		
Descripción:	El administrador genera reportes estadísticos como: total de mascotas, disponibles y adoptadas; adopciones por mes; solicitudes pendientes; total de usuarios y resumen de los últimos seis meses de actividad.		
Criterios de aceptación:	1. Los datos presentados son consistentes con la información almacenada. 2. El reporte de los últimos seis meses muestra datos agrupados por mes. 3. La interfaz de reportes es fácil de interpretar. 4. El sistema filtra reportes por período de tiempo.		

4.1.4. Registro de riesgos

El siguiente registro identifica los principales riesgos asociados al desarrollo del sistema, su probabilidad e impacto (valorados en escala de 1 a 5), el nivel de severidad resultante y el plan de mitigación correspondiente. La severidad se obtiene a través de la multiplicación de la probabilidad por impacto: valores de 1 a 7 corresponden a riesgo bajo, de 8 a 14 a riesgo medio, y de 15 a 25 a riesgo alto.

Tabla 13.*Registro de riesgos del proyecto*

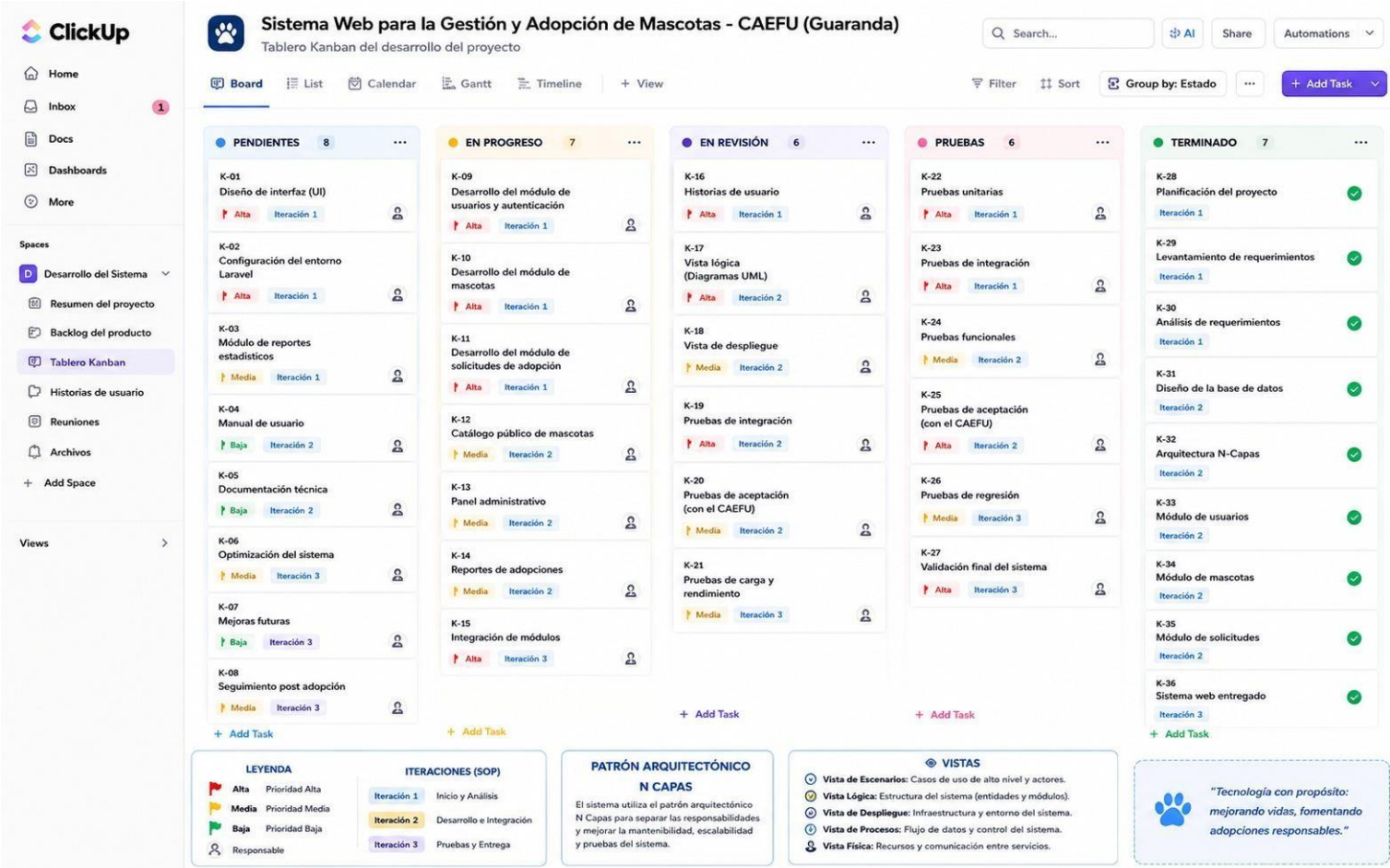
Nº	Riesgo	Prob. (1-5)	Impacto (1-5)	Severidad	Plan de mitigación
1	Cambios frecuentes en los requerimientos por parte del personal del CAEFU durante el desarrollo.	3	4	12 — Media	Mantener comunicación continua con el administrador del CAEFU. Aplicar iteraciones cortas de XP para validar las funcionalidades antes de continuar.
2	Falta de disponibilidad del administrador del CAEFU para validar entregables y realizar pruebas de aceptación.	3	4	12 — Media	Coordinar las sesiones de revisión con anticipación y definir un calendario de entregas acordado desde la planificación.
3	Errores en la integración entre el frontend (React) y el backend (Laravel) durante el desarrollo.	3	3	9 — Media	Aplicar integración continua y ejecución de pruebas de integración automatizadas al cierre de cada iteración.
4	Vulnerabilidades de seguridad en el módulo de autenticación y gestión de datos personales.	2	5	10 — Media	La autenticación es resuelta con tokens JWT, contraseñas cifradas con bcrypt y validación de entradas en el servidor.
5	Fallos en el servicio de la API de WhatsApp para notificaciones automáticas.	3	3	9 — Media	Incorporar el manejo de excepciones junto con un sistema de reintentos.
6	Rendimiento deficiente del sistema ante un número elevado de usuarios concurrentes.	2	4	8 — Media	Optimizar las consultas a la base de datos, agregar paginación en el catálogo y realizar pruebas de carga antes de entregar.
7	Pérdida de datos por fallo en la base de datos PostgreSQL.	1	5	5 — Baja	Configurar respaldos automáticos periódicos de la base de datos y documentar el

8	Dificultad del personal del CAEFU para operar el sistema por falta de conocimientos técnicos.	3	3	9 — Media	procedimiento de restauración. Elaborar un manual de usuario y realizar una sesión de capacitación al personal antes de la entrega definitiva.
---	---	---	---	--------------------------------------	--

4.1.5. Gestión del trabajo mediante Kanban

Para la gestión y seguimiento del avance del proyecto, se implementó un tablero Kanban con la herramienta ClickUp, así se organizó las tareas en cinco columnas que representan los estados del flujo de trabajo: Por hacer, En progreso, En revisión, Pruebas y Terminado. Este esquema permite visualizar en tiempo real el estado de cada historia de usuario, detectar bloqueos y mantener un ritmo de desarrollo constante y controlado.

Tabla 14.
Tablero Kanban



4.2. Diseño

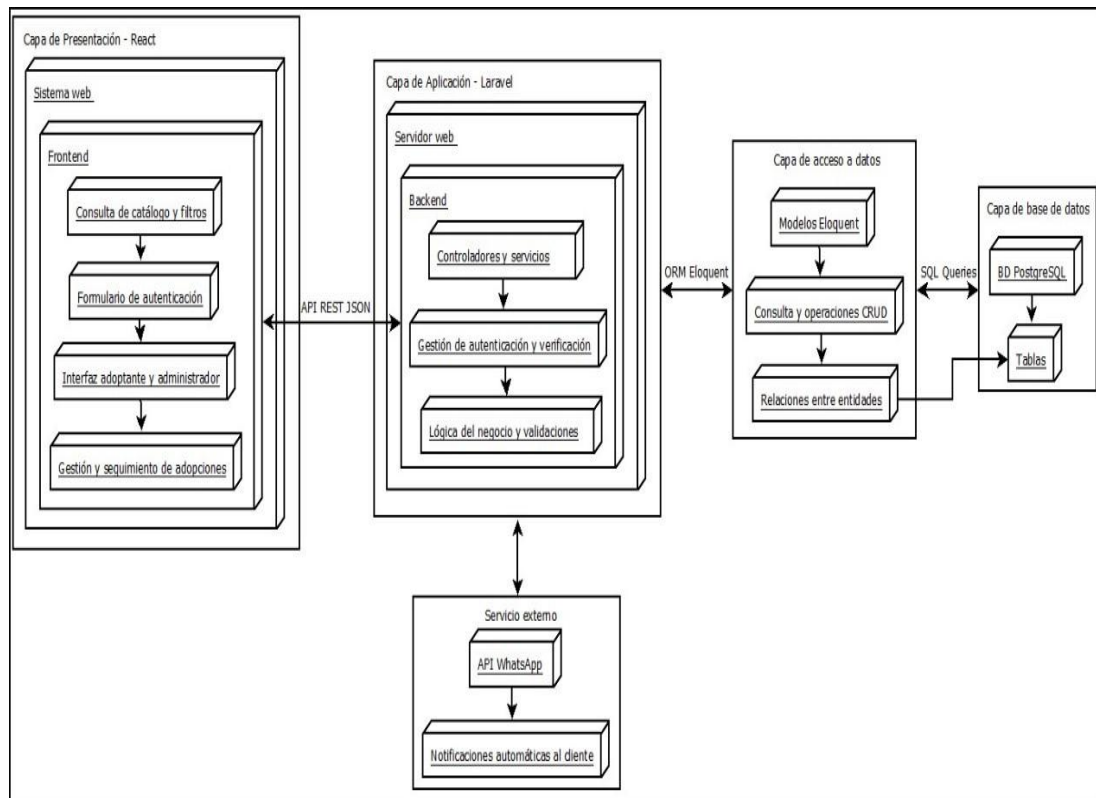
4.2.1. Arquitectura de software

4.2.1.1. Patrón de diseño arquitectónico N-capas

El sistema web del CAEFU adopta una arquitectura N-capas que distribuye las responsabilidades en niveles independientes, de esta manera se facilita el mantenimiento, escalabilidad y seguridad del sistema. La capa de presentación se desarrolla con React, capa de aplicación con el framework Laravel (PHP), capa de acceso a datos utiliza el ORM Eloquent para operaciones con base de datos, y capa de persistencia emplea PostgreSQL. Adicional, se integra un servicio externo de la API de WhatsApp para el envío de notificaciones automáticas a los usuarios adoptantes.

Figura 2.

Diagrama de patrón arquitectónico de N-capas



Nota. Elaboración propia

4.2.2. Vistas arquitectónicas 4+1

El modelo de vistas arquitectónicas 4+1 permite describir la arquitectura del sistema desde perspectivas complementarias, así se facilita la comprensión del diseño para el equipo de desarrollo y para los interesados del proyecto.

4.2.2.1. Vista de escenarios (+1)

El diagrama de casos de uso representa la vista de escenarios del sistema web del CAEFU, en la que intervienen dos actores externos, el Administrador y el Usuario (adoptante). Ambos comparten las funcionalidades de Registrarse, Iniciar sesión, Autenticación y Cerrar sesión, pues el proceso de acceso y validación es el mismo.

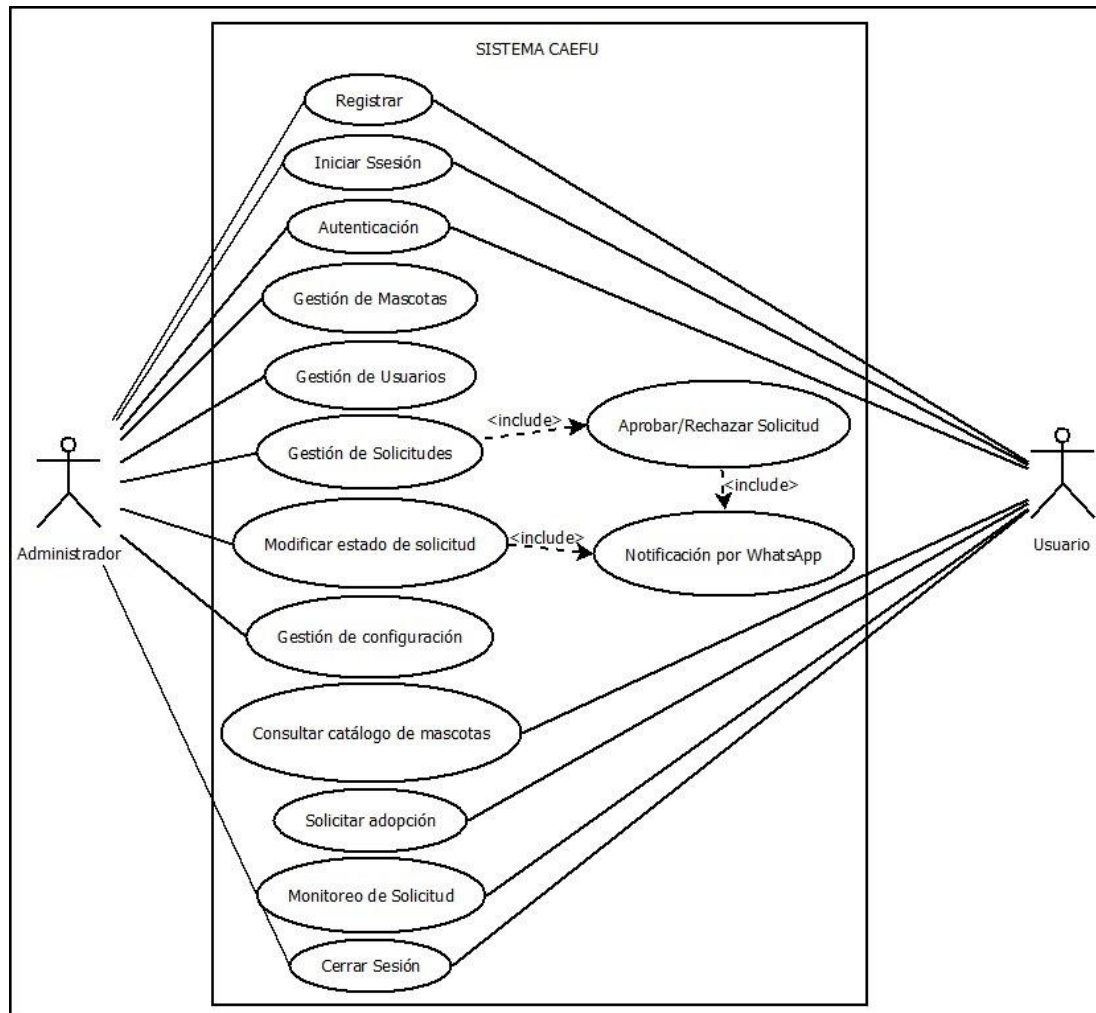
Al Administrador le corresponde la Gestión de Mascotas, Gestión de Usuarios, Gestión de Solicitudes, Modificar Estado de Solicitud y Gestión de Configuración. El caso de uso Gestión de Solicitudes incluye la funcionalidad Aprobar/Rechazar Solicitud, con la que se evalúan y deciden las solicitudes de adopción que presentan los usuarios. Modificar Estado de Solicitud incluye Aprobar/Rechazar Solicitud, porque cambiar el estado de una solicitud requiere una decisión sobre su aprobación o rechazo.

El caso de uso Aprobar/Rechazar Solicitud incluye la funcionalidad Notificación por WhatsApp cada vez que una solicitud se aprueba o se rechaza, el sistema avisa automáticamente al usuario el resultado del proceso.

Las funcionalidades exclusivas del Usuario (adoptante) incluyen Consultar Catálogo de Mascotas, Solicitar Adopción y Monitorear Solicitud. Estas opciones permiten al usuario visualizar las mascotas disponibles, gestionar solicitudes de adopción y realizar el seguimiento del estado de sus trámites dentro de la plataforma.

Figura 3.

Diagrama de caso de uso (escenarios)



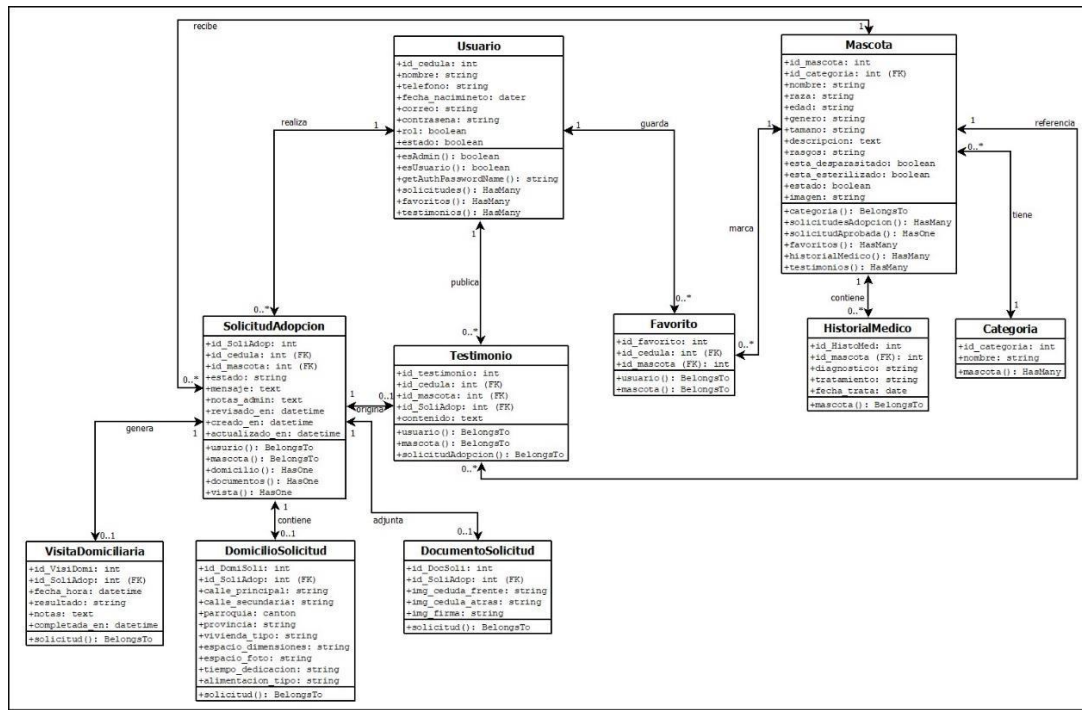
Nota. Elaboración propia

4.2.2.2. Vista de diseño lógico

La vista lógica describe la estructura estática del sistema mediante el diagrama de clases, el cual identifica las entidades principales que participan en el proceso de adopción de mascotas, sus atributos, tipos de datos y los métodos asociados a cada una. Las clases fundamentales del sistema son: Usuario, Mascota, Administrador, Solicitud de Adopción, Testimonio y Reporte.

Figura 4.

Diagrama de clases (Vista lógica)



Nota. Elaboración propia

La clase Usuario representa a las personas que acceden al sistema para explorar el catálogo de mascotas y enviar solicitudes de adopción. Sus atributos principales incluyen identificador, nombre, apellido, cédula, correo, contraseña y rol. Dispone de métodos de autenticación como esAdmin() y esUsuario(), además de las relaciones solicitudes(), favoritos() y testimonios().

La clase Mascota almacena la información de cada animal registrado en el CAEFU, incluye nombre, especie, edad, género, tamaño, descripción, fotografía, estado de adopción y datos de salud (vacunación, desparasitación), con métodos que vinculan cada animal a su categoría, solicitudes de adopción, favoritos e historial médico.

La clase SolicitudAdopcion registra cada petición de adopción, guarda una referencia a un usuario (cédula) y a una mascota, además del estado, el mensaje, las notas administrativas y las fechas de creación, revisión y actualización. Un usuario realiza cero o varias solicitudes, y una mascota recibe cero o varias solicitudes, según las

multiplicidades 1 a 0..* que conectan estas tres clases. A partir de cada solicitud se generan tres elementos complementarios: el domicilio del solicitante (DomicilioSolicitud, con dirección, tipo de vivienda y datos de dedicación), los documentos adjuntos (DocumentoSolicitud, con las imágenes de cédula y firma) y, de forma opcional, una visita (VisitaDomiciliaria, con fecha, resultado y notas de la inspección).

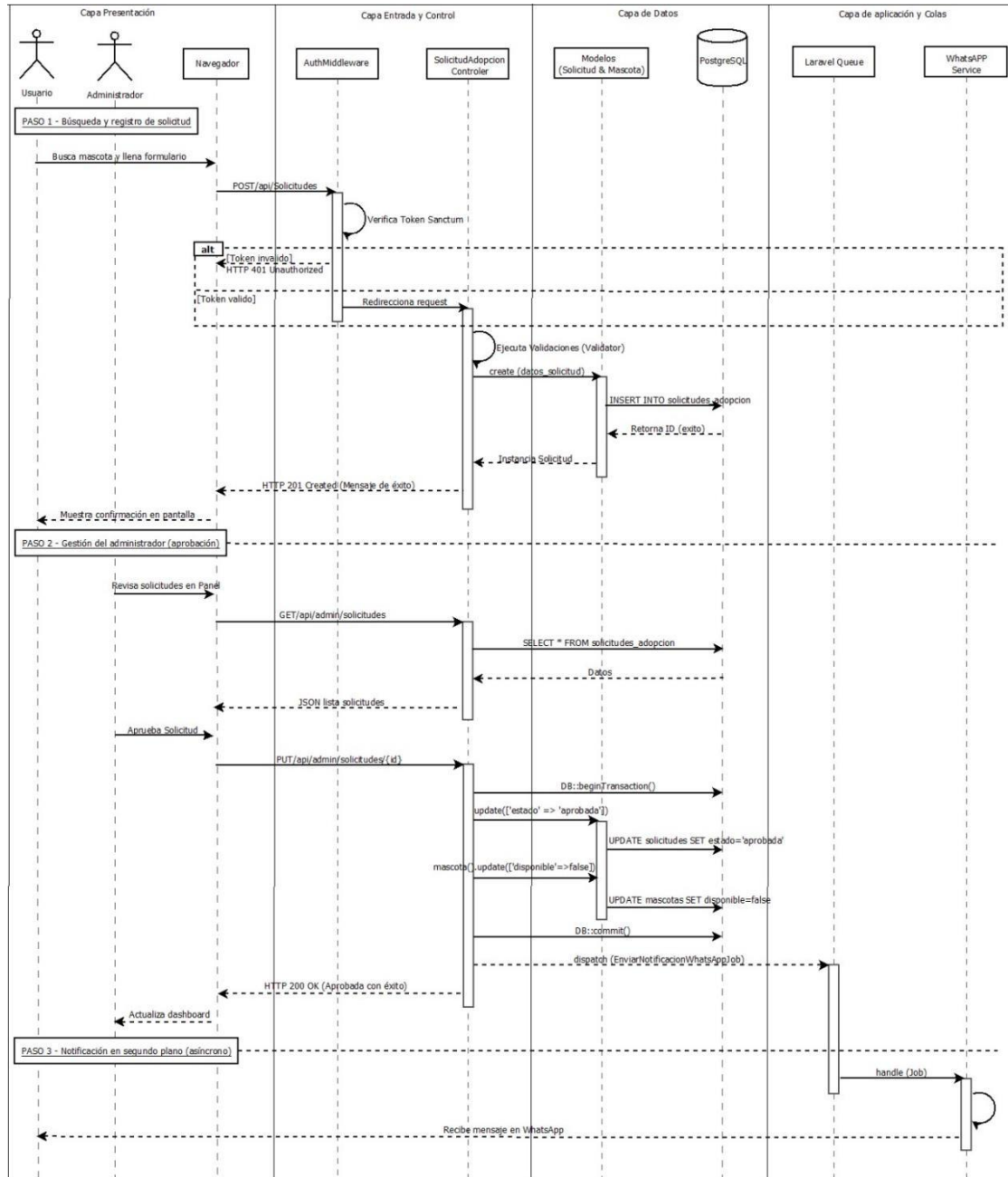
Las clase Favorito registra qué mascotas marca un usuario como preferidas, la clase Testimonio almacena el contenido que un usuario publica sobre una mascota y permite el enlace opcional con la solicitud de adopción de la cual se origina ese comentario. Categoría clasifica a las mascotas (una categoría agrupa cero o varias mascotas) e HistorialMedico almacena el diagnóstico, tratamiento y fecha de cada atención veterinaria asociada a una mascota, estas clases dependen de Mascota.

Diagrama de secuencia — proceso de adopción

El diagrama de secuencia describe el flujo dinámico de interacciones entre el usuario adoptante, administrador del CAEFU y componentes de software, durante el proceso completo de adopción.

Figura 5.

Diagrama de secuencia – Proceso de adopción



Nota. Representación del diagrama de secuencia que describe la interacción entre la capa de presentación, integrada por el Usuario, Administrador y Navegador; capa de control conformada por AuthMiddleware y SolicitudAdopcionController; capa de datos compuesta por los modelos Solicitud y Mascota y base de datos PostgreSQL; capa de aplicación representada por Laravel Queue y WhatsApp Service.

La primera fase Búsqueda y registro de solicitud inicia cuando el usuario consulta el catálogo de mascotas y completa el formulario de adopción desde el navegador que envía una petición POST /api/solicitudes al controlador. Antes de procesar la información el sistema ejecuta la validación del token de autenticación mediante AuthMiddleware y Sanctum, si el token no es válido, el sistema devuelve una respuesta HTTP 401 (No autorizado). Si la autenticación es exitosa el controlador valida los datos recibidos y registra la solicitud. El modelo correspondiente almacena la información en la tabla solicitudes_adopcion de PostgreSQL y retorna el identificador generado. El controlador responde con un código HTTP 201 (Creado) y el usuario recibe la confirmación del registro de su solicitud.

La segunda fase Gestión del administrador (aprobación), comienza cuando el administrador accede al panel de solicitudes, el navegador envía una petición GET /api/admin/solicitudes, donde el controlador consulta la base de datos y recupera la información almacenada en la tabla solicitudes_adopcion. El sistema devuelve esos resultados en JSON para visualizarlos.

Cuando el administrador aprueba una solicitud, el navegador envía un PUT /api/admin/solicitudes/{id}; ahí el controlador inicia una transacción con DB::beginTransaction() y actualiza el estado de la solicitud a “aprobada” y modifica la disponibilidad de la mascota asociada. PostgreSQL registra estos cambios sobre las tablas de solicitudes y mascotas, tras confirmar la transacción con DB::commit(), el controlador envía un trabajo asíncrono (EnviarNotificacionWhatsAppJob) a la cola de Laravel y responde con un código HTTP 200 indicando que la aprobación fue exitosa.

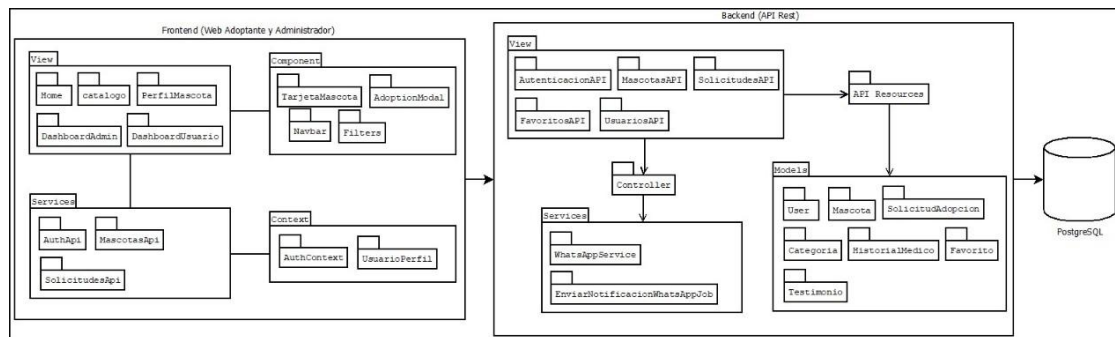
La tercera fase Notificación en segundo plano (asíncrona) corresponde al procesamiento independiente de la tarea encolada, donde Laravel Queue ejecuta el método handle() del trabajo programado y establece comunicación con el servicio externo de WhatsApp. El servicio envía la notificación al adoptante con el resultado de la solicitud.

4.2.2.3. Vista de despliegue

La vista de despliegue define la distribución de los componentes de software sobre los recursos de hardware. El diagrama se divide en dos contenedores, Frontend (Web Adoptante y Administrador) y Backend (API REST), y un tercer elemento que corresponde al motor de base de datos.

Figura 6.

Diagrama de paquetes (Vista de despliegue)



Nota. Representación de la distribución lógica de los paquetes de software del sistema. El Frontend React se comunica con el Backend Laravel mediante peticiones. El Backend gestiona la lógica de negocio y se conecta al motor relacional PostgreSQL para la persistencia de la información. Elaboración propia

El paquete Frontend se estructura en cuatro subpaquetes como View que reúne las vistas principales de la aplicación como Home, Catálogo, Perfil Mascota, DashboardAdmin y DashboardUsuario. El paquete Component contiene los componentes reutilizables de la interfaz como TarjetaMascota, AdoptionModal, Navbar y Filters. El paquete Services integra los módulos responsables de la comunicación con el backend, tales como AuthApi, MascotasApi y SolicitudesApi. El paquete Context agrupa AuthContext y UsuarioPerfil, encargados de la gestión del estado de autenticación y de la información del usuario dentro de la aplicación.

El paquete Backend corresponde a la capa de aplicación desarrollada en Laravel, el subpaquete View agrupa los controladores de la API: AutenticacionAPI, MascotasAPI, SolicitudesAPI, FavoritosAPI y UsuariosAPI. Estos componentes envían la información al paquete API Resources responsable de estructurar las respuestas JSON destinadas al frontend. Cada API interactúa con un Controller principal, que delega

funciones al paquete Services, donde se encuentran WhatsAppService y EnviarNotificacionWhatsAppJob, encargados del envío automático de notificaciones. El paquete Models contiene las entidades del dominio como User, Mascota, SolicitudAdopcion, Categoría, HistorialMedico, Favorito y Testimonio.

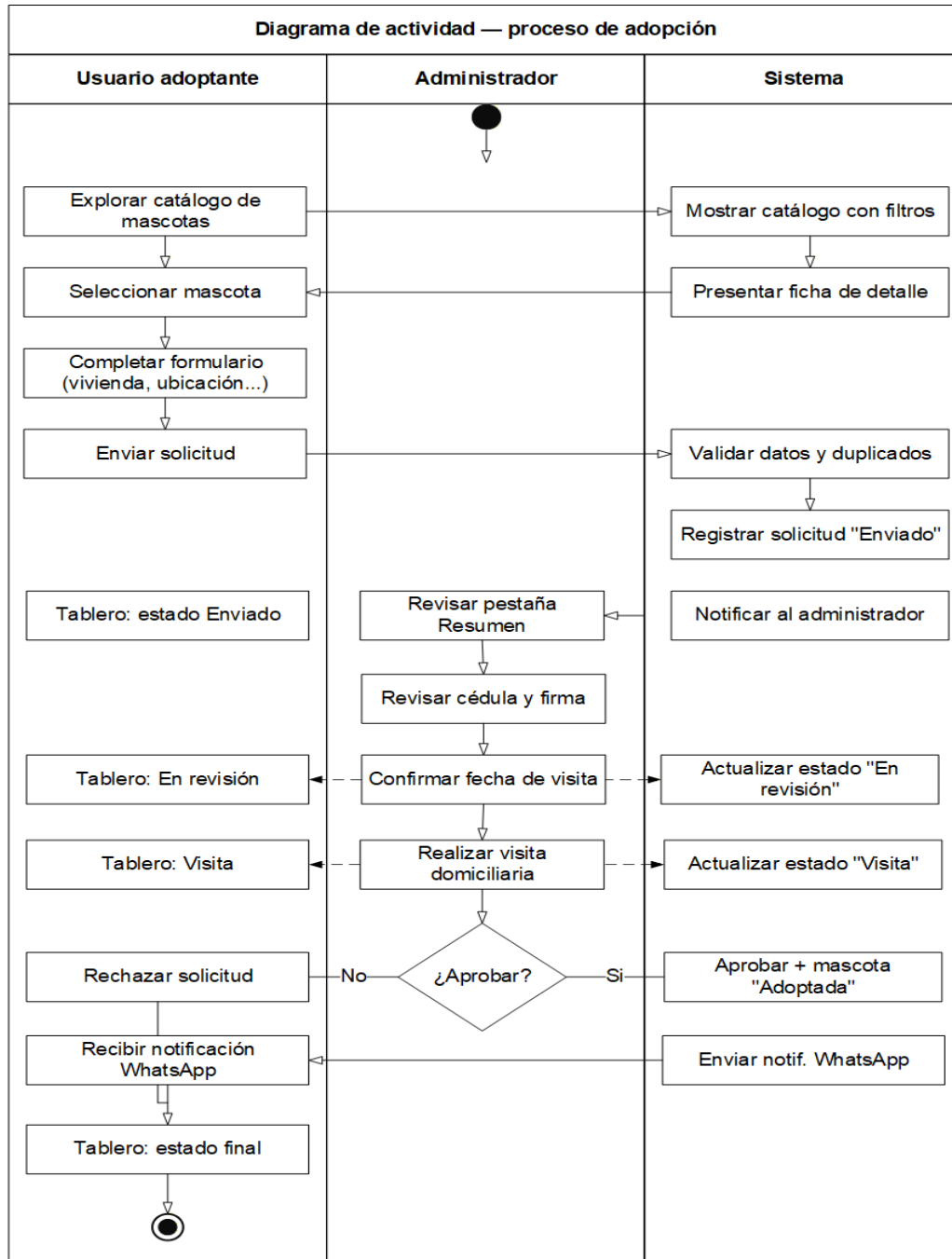
4.2.2.4. Vista de procesos

La vista de procesos representa, mediante diagramas de actividades, los flujos operativos principales del sistema; estos diagramas modelan la secuencia de acciones, decisiones y estados durante la ejecución de las funcionalidades más relevantes.

- a) **Registro de mascotas:** El administrador accede al panel de gestión de mascotas y completa el formulario con los datos del animal; el sistema valida los campos obligatorios y, el registro se guarda en la base de datos con estado "Disponible" y la mascota queda visible en el catálogo público. En caso de error el sistema muestra el mensaje para corrección.
- b) **Solicitud de adopción:** El usuario adoptante accede al catálogo, selecciona una mascota y completa el formulario con la información de su tipo de vivienda, dimensiones del espacio, horas de dedicación, tipo de alimentación, ubicación, cédula, firma, fecha de visita y aceptación de términos. El sistema valida los campos obligatorios y registra la solicitud en estado "Enviado". El usuario visualiza el tablero de seguimiento con las etapas: Enviado, En revisión, Visita y Aprobado.
- c) **Revisión administrativa:** cuando llega una solicitud nueva el administrador recibe la notificación y entra al panel de revisión. En la pestaña Resumen ve los datos generales; en Solicitud revisa la cédula y la firma adjuntas; en Visita completa un checklist de cinco puntos. Después de la visita registra si aprueba o rechaza la solicitud, con sus observaciones, y el sistema actualiza los estados correspondientes y notifica al usuario por WhatsApp.

Figura 7.

Diagrama de actividad — proceso de adopción



Nota. Representación del flujo de actividades del proceso de adopción mediante un diagrama con tres swimlanes: Usuario adoptante, Administrador y Sistema. El diagrama modela el ciclo de adopción. Elaboración propia

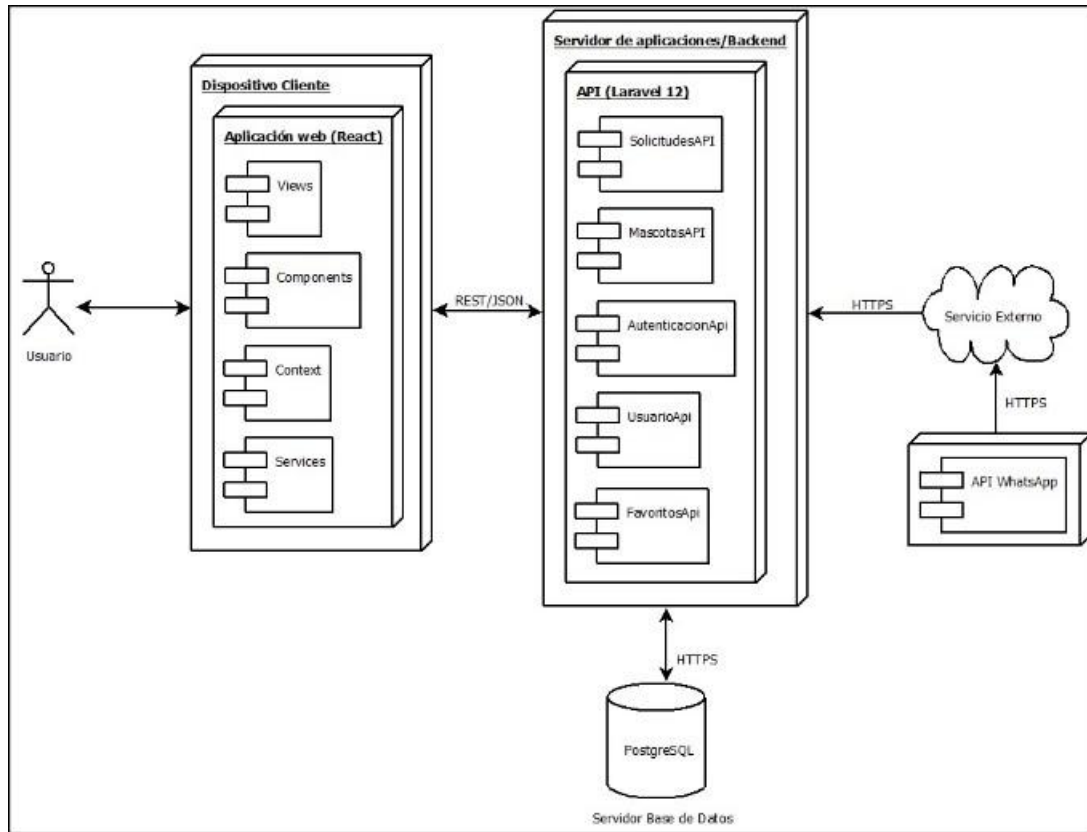
4.2.2.5. Vista física

La vista física del sistema se representa mediante el diagrama de despliegue, el cual ilustra la topología del hardware y los entornos de ejecución sobre los que se distribuyen los componentes de software. La infraestructura propuesta se compone de cuatro nodos.

- a) **Dispositivo cliente:** representa el dispositivo desde el cual el usuario accede a la aplicación web. En este nodo se ejecuta la Aplicación web (React), organizada en cuatro paquetes: Views, Components, Context y Services. La comunicación entre el dispositivo cliente y servidor de aplicaciones se realiza mediante el protocolo REST/JSON.
- b) **Servidor de aplicaciones / Backend:** aloja la API (Laravel 12) compuesta por cinco módulos: SolicitudesAPI, MascotasAPI, AutenticacionApi, UsuarioApi y FavoritosApi. Este nodo centraliza la lógica de negocio expuesta a través de endpoints y mantiene comunicación bidireccional con el servidor de base de datos mediante el protocolo HTTPS.
- c) **Servidor de base de datos:** Nodo independiente que aloja el motor relacional PostgreSQL que mantiene separado del servidor de aplicaciones por razones de seguridad y rendimiento. El acceso a este nodo desde el backend se realiza mediante peticiones HTTPS..
- d) **Servicio externo de mensajería (API de WhatsApp):** la comunicación entre el servidor de aplicaciones y API de WhatsApp no es directa, se establece a través de un nodo denominado Servicio Externo. El backend envía las instrucciones hacia este servicio externo mediante HTTPS y este servicio reenvía la solicitud al nodo API WhatsApp mediante HTTPS, para el envío de las notificaciones automáticas a los usuarios adoptantes cuando cambia el estado de su solicitud de adopción.

Figura 8.

Diagrama de despliegue (Vista física)



Nota. Representación de la topología física del sistema CAEFU. El diagrama ilustra los cuatro nodos de infraestructura con sus componentes internos y los protocolos de comunicación entre ellos: REST/JSON, JDBC y HTTPS. Elaboración propia.

4.3. Codificación

En esta fase el proyecto se dividió en backend desarrollado en Laravel y frontend en React, integrados mediante una API REST y respaldados por PostgreSQL como sistema de gestión de base de datos. El backend se organiza en carpetas que cumplen un rol dentro de la lógica del sistema de adopción de mascotas: los Controllers gestionan las peticiones HTTP y lógica de negocio separados por dominio entre autenticación, mascotas, usuarios y panel administrativo; los Requests centralizan las reglas de validación de los formularios antes de que los datos lleguen al controlador;

los Resources dan forma a las respuestas JSON que se envían al cliente, y los Services encapsulan lógica reutilizable, como el manejo de avatares de usuario.

4.3.1. Backend Laravel

El backend de la plataforma se desarrolló en Laravel y expone una API REST organizada en controladores, divididos en los endpoints públicos orientados al usuario final y los endpoints del panel administrativo, junto con un módulo de autenticación compartido por ambos perfiles.

Controllers (/app/Http/Controllers)

- **UsuarioPerfilController:** Administra el perfil del usuario autenticado a través de muestra, actualiza datos personales - avatar y permite el cambio de contraseña, para lo cual verifica la contraseña actual y exige que la nueva incluya mayúscula, minúscula y número.
- **SocialAuthController:** Maneja el login social con Google y Facebook a través de Socialite. En el callback, crea o vincula la cuenta del usuario y genera un código temporal de uso único por sesenta segundos. El frontend lo canjea con POST, evitando que el token quede expuesto en la URL.
- **PasswordResetController:** Cubre el flujo de recuperación de contraseña con un código de seis dígitos enviado por correo, válido durante quince minutos. El proceso pasa por envío del código, verificación y reseteo final.
- **MascotaController:** Expone los endpoints públicos para listar mascotas disponibles con filtros de categoría, tamaño, género y edad, además de consultar mascotas destacadas o por su ID. Tiene un endpoint protegido para que el usuario autenticado vea sus mascotas favoritas.
- **FavoritoController:** Implementa un CRUD simple de favoritos que alterna una mascota como favorita, lista los identificadores favoritos, objetos completos de las mascotas favoritas y marca como visto el tutorial correspondiente.
- **Controller:** Constituye la clase abstracta base de la cual heredan todos los demás controladores, conforme al patrón estándar de Laravel.

- **AuthController:** Gestiona registro e inicio de sesión con correo y contraseña, generando tokens Sanctum. Cuando algo falla devuelve mensajes genéricos para no revelar si un correo ya existe.
- **AdminUsuarioController:** Desde el panel administra los usuarios: estadísticas en caché, listado paginado con filtros, actualización de datos, rol y estado. Incluye protección para que un administrador no pueda desactivarse a sí mismo, eliminación por borrado lógico y consulta de las solicitudes de adopción de un usuario.
- **AdminPerfilController:** Muestra el perfil, permite actualizarlo con sanitización de datos y avatar, y habilita el cambio de contraseña previa verificación de la actual.
- **AdminNotificacionController:** Genera notificaciones a partir de solicitudes pendientes, controles médicos próximos o vencidos, y mascotas con más de treinta días en espera. Las ordena por prioridad y expone un endpoint que devuelve el conteo total.
- **AdminDashboardController:** Construye el panel principal del administrador con estadísticas comparativas entre el mes actual y el anterior, datos del gráfico de adopciones de los últimos seis meses, mascotas recientes y feed de actividad reciente; toda esta información se mantiene en caché durante cinco minutos.
- **SolicitudAdopcionController:** Maneja el ciclo de vida completo de una solicitud de adopción, para ello el usuario crea, reenvía o solicita una reevaluación; el administrador lista, filtra y cambia su estado en dos etapas, de pendiente a en visita y aprobada o rechazada, la mascota se marca como no disponible en la aprobación final y se disparan las notificaciones por WhatsApp.
- **AdminMascotaController:** Ofrece el CRUD de mascotas para el administrador: listado con filtros y estadísticas en caché, creación y edición con fotos e historial médico sanitizado, eliminación por borrado lógico, y utilidades para obtener el próximo ID o eliminar un registro médico puntual.

a) **Models (/app/Models)**

El backend contiene los modelos Eloquent que representan las tablas del sistema: categoría, configuración, favorito, historial médico, mascota, imagen de mascota,

solicitud de adopción, testimonio y usuario. Estos modelos encapsulan reglas de negocio como relaciones entre entidades, casteo de datos y definición de atributos asignables y permiten mediante Eloquent ORM, la interacción estructurada con la base de datos en operaciones de inserción, actualización, eliminación y consulta de información sobre las mascotas y su proceso de adopción.

- **Categoría:** Modelo para las categorías de mascotas, cuenta con nombre, descripción e imagen y relación hasMany hacia Mascota.
- **Favorito:** Actúa como tabla pivote entre usuario y mascota, representando el "me gusta". Se relaciona mediante belongsTo para User y Mascota.
- **HistorialMedico:** Registra los eventos médicos de una mascota, junto con fecha, detalle, próximo control y notas. Se relaciona con Mascota mediante belongsTo.
- **Mascota:** Incluye soft deletes y varias relaciones: categoría, solicitudes de adopción, favoritos (belongsToMany), testimonios, imágenes, historial médico y la adopción aprobada mediante el recurso ofMany.
- **MascotaImagen:** Representa la fotografía de una mascota, con un indicador de imagen principal y un orden de visualización para la galería. Se relaciona con Mascota mediante belongsTo.
- **SolicitudAdopcion:** Guarda los datos del proceso de adopción y un historial de rechazos almacenado en JSON. Se relaciona tanto con User como con Mascota.
- **Testimonio:** Almacena las reseñas que un usuario deja sobre una mascota adoptada, con calificación numérica e indicador de aprobación que permite la moderación del contenido. Se relaciona con User y con Mascota.
- **User.** Se vincula a la tabla usuarios e incorpora soft deletes, tokens Sanctum y columnas para la contraseña (contrasena) y verificación de roles (esAdmin, esUsuario), estas últimas con soporte para un rol híbrido que combine ambos. Define las relaciones hacia las solicitudes de adopción, favoritos, mascotas favoritas y testimonios.

b) Services (/app/Services)

- **WhatsAppService.** Se encarga del envío de notificaciones por WhatsApp a los usuarios sobre el resultado de su solicitud de adopción, a través de un proveedor

externo Green API, que conecta un número de WhatsApp convencional con una API HTTP. Dispone de dos métodos, uno que avisa al usuario que la adopción fue aprobada con instrucciones para firma del acta, y otro que comunica el rechazo, cuyo contenido cambia según la etapa de proceso, revisión documental o visita domiciliaria, incluye una invitación a corregir y reenviar la solicitud o una reevaluación de la vivienda. De forma interna construye el mensaje con el formato de WhatsApp y lo envía por HTTP a Green API; si la API responde con error, el servicio lanza una excepción que permite al job en cola reintentar el envío.

c) Providers (/app/Providers)

El proveedor de servicios corresponde a una de las clases estándar de Laravel, en la cual la aplicación registra y configura los comportamientos globales que se ejecutan al inicio.

- **Método register():** permanece vacío en este punto se registrarían vínculos personalizados en el contenedor de dependencias, pero el proyecto no recurre a esa práctica.
- **Método boot():** verifica si el entorno de la aplicación corresponde a producción y obliga a que todas las URL generadas por Laravel usen https en lugar de http. Esta configuración evita que el framework genere enlaces con HTTP cuando opera detrás de proxies o balanceadores de carga, situación que ocasiona errores de contenido mixto y compromete la seguridad de enlaces, como los de recuperación de contraseña. La restricción se aplica en producción para no afectar el entorno de desarrollo local

d) Migrations (/database/migrations)

El directorio de migraciones contiene la estructura completa de la base de datos del sistema, con definiciones de tablas de usuarios, mascotas, categorías, solicitudes de adopción, favoritos, testimonios, historial médico, domicilio de solicitud, documento de solicitud y visita domiciliaria. Cada entidad establece atributos con el tipo de dato correspondiente, sus claves primarias y foráneas, y las relaciones de cardinalidad necesarias entre tablas.

e) Seeders (/database/seeders)

Los seeders se utilizaron para cargar la información inicial para el funcionamiento del sistema, un usuario administrador por defecto, categorías de mascotas disponibles y datos de prueba para mascotas, todos ejecutados desde el DatabaseSeeder principal.

- **AdminSeeder:** Crea o actualiza un usuario administrador con correo fijo y rol admin.
- **CategorySeeder:** Inserta las categorías base de mascotas con nombre y descripción, evita duplicados mediante updateOrCreate por nombre.
- **DatabaseSeeder.** Ordena la ejecución de todos los seeders en orden, AdminSeeder, CategorySeeder y PetSeeder, cada vez que se ejecuta el comando de sembrado de la base de datos.

f) Factories (/database/factories)

El directorio factories facilita la creación de datos ficticios de usuarios a través de una plantilla con valores aleatorios empleada en las pruebas automatizadas y en el poblado rápido de la base de datos durante el desarrollo, permite la generación de escenarios completos para la validación de los controladores.

g) Routes (/routes/api.php)

Este directorio define los endpoints que el frontend en React consume mediante la API REST y funciona como un mapa de comunicación entre las funcionalidades visuales y la lógica del backend. Las rutas se organizan por módulos, autenticación, mascotas, favoritos, solicitudes de adopción y administración, cada uno enlazado con controlador correspondiente dentro de /app/Http/Controllers.

- **api.php:** Contiene los endpoints de la API REST agrupados en cuatro niveles de acceso, rutas públicas (registro, inicio de sesión y catálogo de mascotas), protegidas con throttle contra ataques de fuerza bruta; rutas compartidas que exigen una sesión autenticada; rutas exclusivas del usuario normal mediante middleware is_usuario para favoritos y solicitudes de adopción; y rutas exclusivas del administrador

mediante middleware `is_admin` para dashboard, gestión de mascotas, usuarios, solicitudes y notificaciones.

- **web.php:** Maneja las rutas que requieren sesión basada en navegador en lugar de tokens, como el inicio de sesión social con Google y Facebook, dado que el flujo OAuth necesita redirecciones del navegador que no funcionan con rutas API.
- **console.php:** Define los comandos de Artisan ejecutables desde la terminal, se conserva el comando de ejemplo `inspire`, incluido por defecto en Laravel sin comandos personalizados para el proyecto.

4.3.2. Frontend React

a) Components (/src/components/ui)

Contiene los componentes reutilizables de React que conforman la interfaz del frontend, organizados según si son secciones de página de inicio, tarjetas de mascotas, formularios de filtrado, layout general o guardias de seguridad de rutas. Todos consumen API REST del backend con `fetch` y administran su estado con `hooks`.

- **Featured:** carrusel con la mascota destacada del día, autoplay cada cinco segundos, datos desde `/api/mascotas/destacado`. El botón de adopción manda al login si no hay usuario autenticado.
- **Filters:** Barra de filtros para el catálogo de mascotas como categoría, tamaño, edad, género, favoritas, aplica cambios de forma automática mediante un `useEffect` cada vez que el usuario modifica alguna opción.
- **Footer:** Constituye el pie de página estático con información de contacto, redes sociales y enlaces de navegación del sitio del CAEFU.
- **Hero:** Sección de bienvenida de página de inicio contiene título, descripción institucional y botones de desplazamiento hacia otras secciones de la página.
- **MainLayout:** layout de las rutas públicas, si detecta un usuario con rol de administrador lo saca automáticamente, limpia su sesión local y cierra el token en el backend, para que el panel de admin no se cruce con el sitio público.

- **TarjetaMascota:** tarjeta de una mascota dentro del catálogo, tiene botón de favoritos, insignia de etapa de vida que se calcula a partir del texto de la edad, y el botón de adopción.
- **Testimonio:** sección de testimonios de adoptantes.
- **UserGuard:** protege rutas como /dashboard del lado del cliente. Expulsa a cualquiera sin token o sin rol válido, limpia la sesión y evita que se alcance a ver contenido no autorizado.
- **AdoptionModal:** modal de dos pasos para la solicitud de adopción. El primero pide datos de vivienda y dirección, con autocompletado por geolocalización y un mapa interactivo en Leaflet y cuidados. El segundo arma un acta digital prellenada con los datos de la mascota y del adoptante, agenda la visita domiciliaria y pide subir foto de cédula y firma antes de mandar todo al backend.
- **AuthModal:** modal combinado de login y registro, con pestañas deslizables. Valida la cédula con el módulo 10, revisa mayoría de edad, formato de correo y reglas de contraseña. Si la autenticación es correcta guarda el token y redirige según el rol. También permite entrar con Google.
- **Navbar:** barra de navegación con búsqueda en vivo que filtra sobre todo el listado de mascotas, un menú que cambia según si hay sesión iniciada, un desplegable de cuenta con cierre de sesión, y el disparo de eventos globales (open-auth-modal, open-forgot-password-modal) para poder abrir los modales de autenticación desde cualquier punto de la app.

b) Features (/src/features)

Este directorio contiene las páginas principales del panel de administración, cada una representa una vista completa, con lógica de obtención de datos, filtros, paginación del lado del servidor y modales de gestión. Todas consumen endpoints protegidos de /api/admin/* y comparten un estilo visual de tarjetas estadísticas en la parte superior.

- **Catálogo:** Es la página de exploración con filtros del lado del servidor y tutorial interactivo de dos pasos que guía al usuario la primera vez que accede a sus favoritos.

- **Dashboard:** Panel de usuario donde se muestran sus solicitudes recientes, mascotas favoritas, recomendación destacada y espacio para testimonios. Si quien ingresa es el administrador, se le redirige a su propio panel.
- **Home:** Página pública con sección Hero, listado de mascotas, destacados y testimonios. Los usuarios que ya iniciaron sesión no ven esta pantalla; se les redirige de inmediato a su Dashboard.
- **PerfilMascota:** Vista a detalle de una mascota: galería tipo lightbox, ficha con sus datos, historia y testimonios asociados. Muestra un botón para iniciar la adopción, salvo que ya exista una solicitud activa, en cuyo caso indica el estado pendiente.
- **AuthCallback:** Recibe el código temporal que entrega Google OAuth y lo intercambia por el token real mediante una petición POST segura. Con ese token guarda la sesión y redirige según el rol que tenga el usuario..

Panel de administrador

- **AdminConfiguracion:** Perfil del administrador dividido en una pestañas para datos personales y otra de seguridad para cambio de contraseña. Los cambios se sincronizan con el contexto de autenticación global y se reflejan en tiempo real.
- **AdminDashboard:** Pantalla de inicio del panel, presenta tarjetas estadísticas de mascotas, adopciones, usuarios y solicitudes, cada una comparada con el mes anterior. Incluye un gráfico de área en SVG construido a mano con las adopciones de los últimos seis meses, accesos rápidos hacia otras secciones y listas de mascotas y actividad reciente.
- **AdminMascotas:** Tabla de gestión de mascotas con filtros de búsqueda, estado, categoría, esterilización y desparasitación, paginada desde el servidor. Cuenta con modales para agregar, editar, ver el detalle o eliminar un registro y resalta visualmente la fila correspondiente cuando se llega desde una notificación o desde el dashboard.
- **AdminNotificaciones:** Lista de alertas administrativas como solicitudes pendientes, controles médicos próximos y mascotas en espera prolongada, con

filtros por tipo, insignias de prioridad y cada notificación resulta clicable y redirige hacia la sección correspondiente.

- **AdminSolicitudes:** Tabla de gestión de solicitudes de adopción con tarjetas de estadísticas por estado, filtros de búsqueda y fecha, y un modal de revisión que permite avanzar el flujo en sus dos etapas aprobando o rechazando cada una por separado.
- **AdminUsuarios:** Tabla de gestión de usuarios con tarjetas estadísticas, cambio de rol y estado desde la tabla, historial de adopciones por usuario, y modales para crear, editar, ver o eliminar (mediante borrado lógico) a un usuario.

Panel de usuario

Reúne las páginas donde el usuario administra su cuenta y sigue el estado de sus solicitudes de adopción.

Configuracion: Edición de datos personales y avatar, cambio de contraseña con indicador de fortaleza en tiempo real. Si la cuenta está vinculada a Google el correo queda bloqueado y no se puede editar.

MisSolicitudes: Lista las solicitudes del usuario con filtros por estado y fecha, y muestra una línea de tiempo visual con el avance de cada una (revisión, visita, resultado). Desde ahí el usuario puede corregir una solicitud rechazada o pedir una reevaluación de la vivienda.

c) Lib (/src/lib)

Reúne utilidades y configuraciones compartidas que dan soporte al funcionamiento del sistema: funciones para consumir la API REST construida en Laravel, construcción de URLs hacia el backend (apiUrl) y hacia el almacenamiento de archivos (storageUrl), y manejo del token de autenticación guardado en el navegador (getAuthToken), que se adjunta a cada petición protegida. Con esto se evita repetir la URL base o la lógica de encabezados en cada componente, y se centraliza el acceso a `import.meta.env.VITE_API_URL` y la gestión de la sesión del usuario.

4.3.3. Interfaces de pantalla

a) Tipografía

Para garantizar buena legibilidad del contenido el sistema utiliza la tipografía Inter, de la familia sans-serif, definida como fuente principal en toda la interfaz. Esta tipografía facilita la lectura en pantalla y aporta fluidez visual al usuario.

b) Logo

Figura 9.

Logo del CAEFU

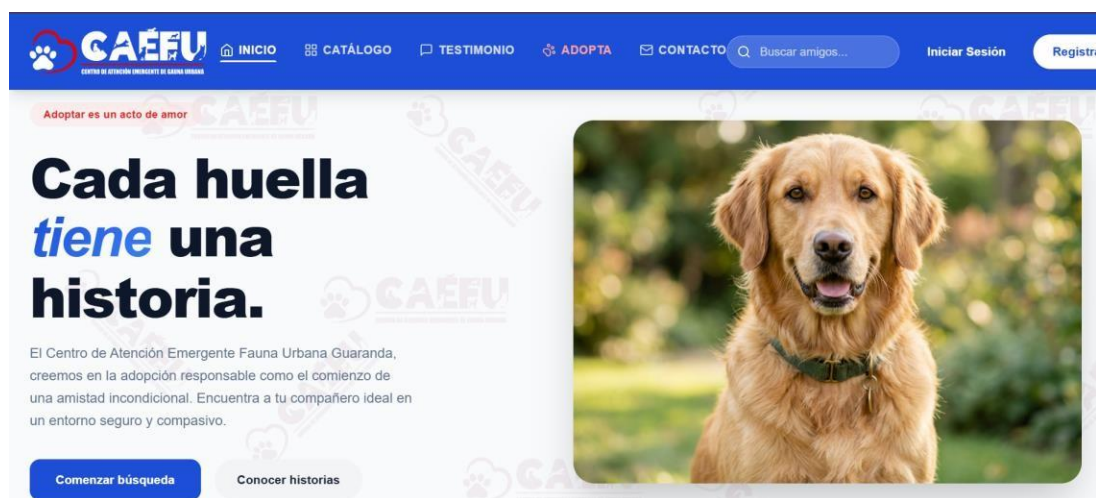


Nota. Pertenece al módulo Frontend dentro de los elementos de identidad visual.

c) Interfaz principal

Figura 10.

Interfaz principal (Home)

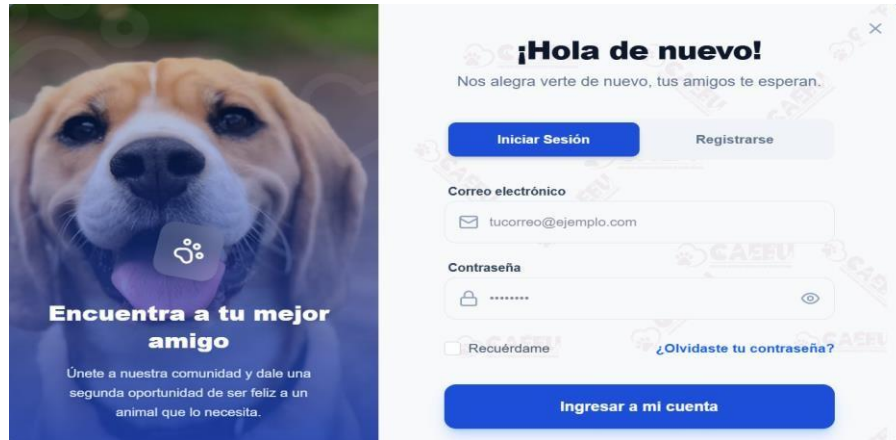


Nota. Página de inicio del frontend desarrollado en React. Pertenece al módulo Frontend, dentro del paquete Features en la página Home. Elaborado por: Cristófer M. Fuente: <https://caefugaranda.com/>

d) Loguin para administrador y usuario

Figura 11.

Inicio de sesión para administrador y usuario

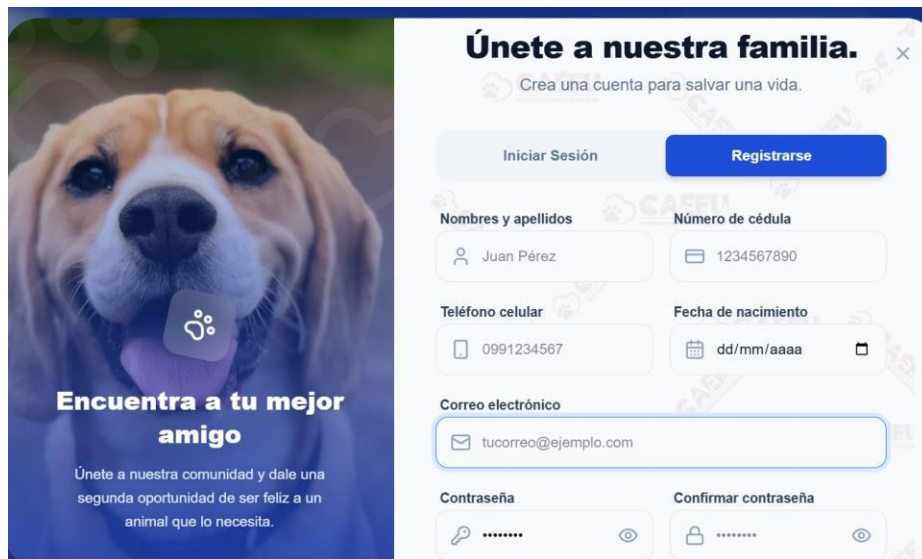


The login modal features a background image of a beagle dog. On the left, a text overlay reads: "Encuentra a tu mejor amigo" and "Únete a nuestra comunidad y dale una segunda oportunidad de ser feliz a un animal que lo necesita." The right side contains a form with the title "¡Hola de nuevo!" and the subtitle "Nos alegra verte de nuevo, tus amigos te esperan." It includes two buttons: "Iniciar Sesión" and "Registrarse". Below these are input fields for "Correo electrónico" (containing "tucorreo@ejemplo.com") and "Contraseña" (masked with asterisks). There is a "Recuérdame" checkbox and a link "¿Olvidaste tu contraseña?". At the bottom is a large blue button labeled "Ingresar a mi cuenta".

Nota. Modal de inicio de sesión con los campos de correo electrónico y contraseña. Pertenecce al módulo Frontend dentro del componente AuthModal. Elaborado por: Cristofer M. Fuente: <https://caefuguaranda.com/>

Figura 12.

Registro de nuevo usuario



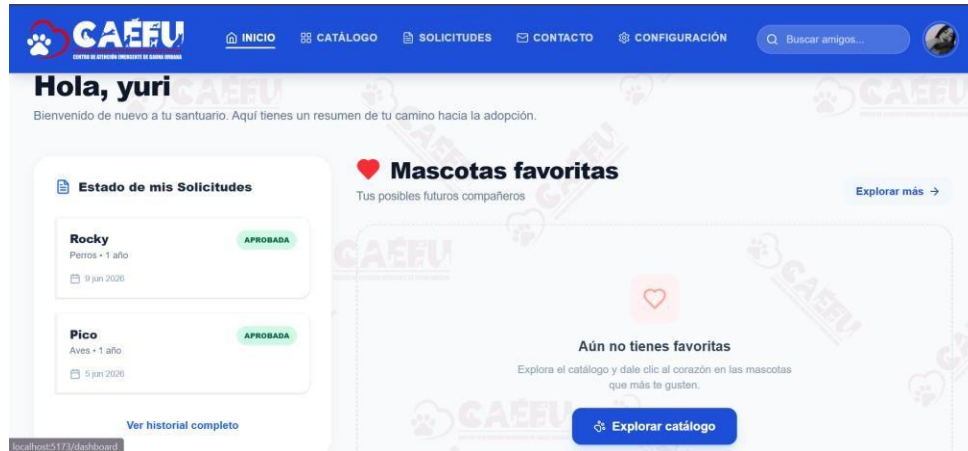
The registration modal features the same beagle dog background. On the left, the text overlay is identical to Figure 11. The right side has the title "Únete a nuestra familia." and the subtitle "Crea una cuenta para salvar una vida." It includes two buttons: "Iniciar Sesión" and "Registrarse". The form contains several input fields: "Nombres y apellidos" (filled with "Juan Pérez"), "Número de cédula" (filled with "1234567890"), "Teléfono celular" (filled with "0991234567"), "Fecha de nacimiento" (with a date picker showing "dd/mm/aaaa"), "Correo electrónico" (filled with "tucorreo@ejemplo.com"), "Contraseña" (masked), and "Confirmar contraseña" (masked). There are also eye icons to toggle password visibility.

Nota. Modal de registro de un nuevo usuario. Pertenecce al módulo Frontend dentro del componente AuthModal. Elaborado por: Cristofer M. Fuente: <https://caefuguaranda.com/>

e) Interfaz usuario

Figura 13.

Panel del usuario adoptante



Nota. Panel del usuario adoptante con estado de solicitudes de adopción y mascotas favoritas. Pertenecer al módulo Frontend dentro del paquete Features en la página Dashboard de usuario. Elaborado por: Cristófer M. Fuente: <https://caefugaranda.com/>

f) Interfaz administrador

Figura 14.

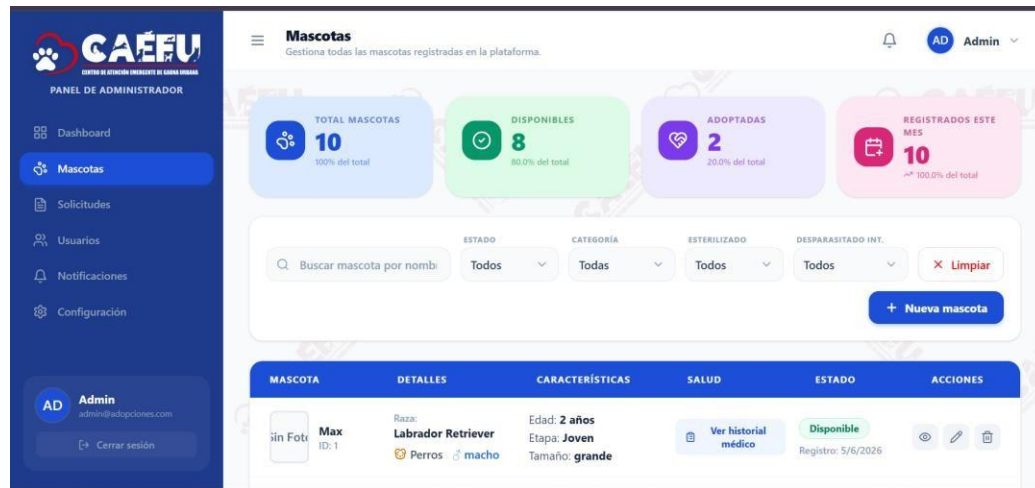
Panel de control



Nota. Panel de control (Dashboard) del administrador con estadísticas generales de la plataforma y el gráfico de adopciones de los últimos seis meses. Pertenecer al módulo Frontend dentro del paquete Features en la página AdminDashboard. Elaborado por: Cristófer M. Fuente: <https://caefugaranda.com/>

Figura 15.

Gestión de mascotas del panel administrativo



Nota. Módulo de gestión de mascotas del panel administrativo con estadísticas y tabla de registros. Pertenecce al módulo Frontend dentro del paquete Features en la página AdminMascotas. Elaborado por: Cristofer M. Fuente: <https://caefugaranda.com/>

Figura 16.

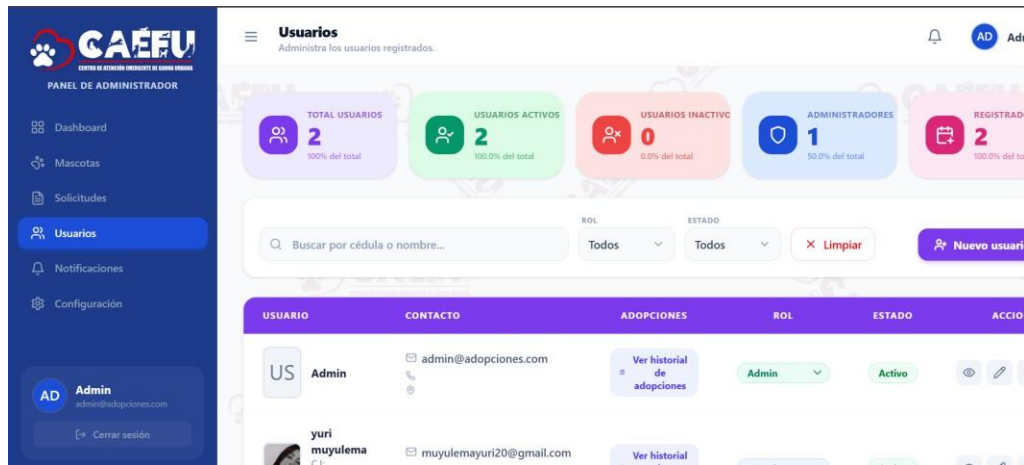
Gestión de solicitudes de adopción del panel administrativo



Nota. Módulo de gestión de solicitudes de adopción del panel administrativo con estadísticas por estado y tabla de registros. Pertenecce al módulo Frontend, del paquete Features en la página AdminSolicitudes. Elaborado por: Cristofer M. Fuente: <https://caefugaranda.com/>

Figura 17.

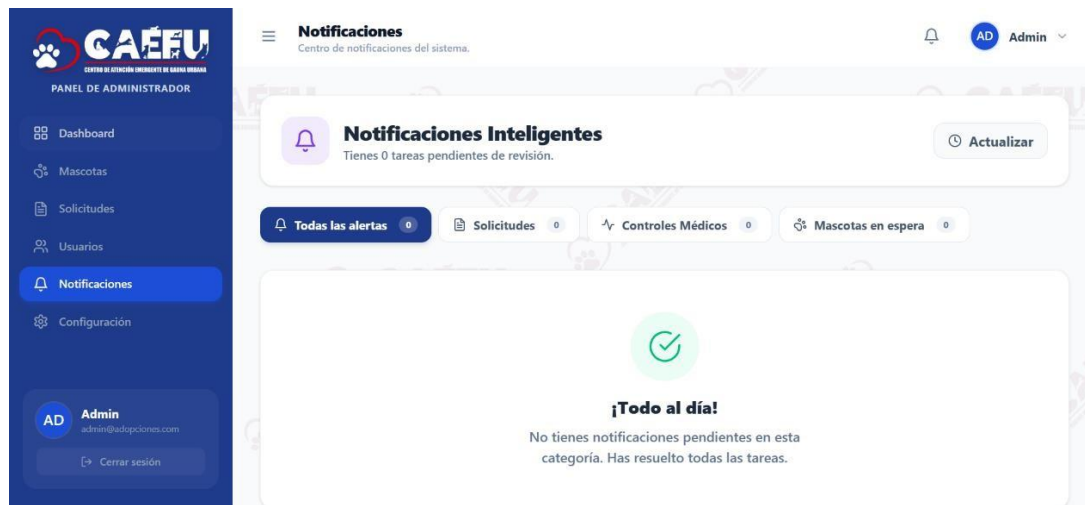
Gestión de usuarios del panel administrativo



Nota. Módulo de gestión de usuarios del panel administrativo con estadísticas y tabla de registros. Pertenece al módulo Frontend del paquete Features en la página AdminUsuarios. Elaborado por: Cristófer M. Fuente: <https://caefugaranda.com/>

Figura 18.

Notificaciones inteligentes del panel administrativo



Nota. Pertenece al módulo Frontend del paquete Features en la página AdminNotificaciones. Elaborado por: Cristófer M. Fuente: <https://caefugaranda.com/>

Figura 19.

Pestaña de datos personales del módulo de configuración del administrador

The screenshot shows the 'Configuración' (Configuration) page for the CAEFU administrator. The left sidebar contains a menu with options: Dashboard, Mascotas, Solicitudes, Usuarios, Notificaciones, and Configuración (highlighted). Below the menu is a user profile for 'Admin' (admin@adopciones.com) with a 'Cerrar sesión' button. The main content area is titled 'Configuración' and 'Ajustes generales de la plataforma.' It features a user profile card for 'Admin' (admin@adopciones.com, member since June 2026) and two tabs: 'Datos Personales' (active) and 'Seguridad'. The 'Datos Personales' tab contains a form with fields for: 'Nombre completo' (Admin), 'Correo electrónico' (admin@adopciones.com), 'Cédula de identidad', 'Teléfono', 'Fecha de nacimiento' (dd/mm/aaaa), and 'Dirección'.

Nota. Pertenecce al módulo Frontend del paquete Features en la página AdminConfiguracion. Elaborado por: Cristófer M. Fuente: <https://caefugaranda.com/>

Figura 20.

Pestaña de seguridad, para el cambio de contraseña, del módulo de configuración del administrador

The screenshot shows the 'Configuración' (Configuration) page for the CAEFU administrator, specifically the 'Seguridad' (Security) tab. The left sidebar is identical to Figure 19. The main content area is titled 'Configuración' and 'Ajustes generales de la plataforma.' It features the same user profile card for 'Admin' and the 'Datos Personales' and 'Seguridad' tabs. The 'Seguridad' tab is active and displays the 'Cambiar Contraseña' (Change Password) section. It includes a note: 'La contraseña debe tener al menos 8 caracteres. Usa una combinación de letras, números y símbolos.' The form has three input fields: 'Contraseña actual' (Ingresa tu contraseña actual), 'Nueva contraseña' (Mínimo 8 caracteres), and 'Confirmar nueva contraseña' (Repite la nueva contraseña). A 'Cambiar contraseña' button is at the bottom.

Nota. Pertenecce al módulo Frontend del paquete Features en la página AdminConfiguracion. Elaborado por: Cristófer M. Fuente: <https://caefugaranda.com/>

g) Código

A continuación se presenta el código fuente del servicio WhatsAppService encargado del envío de notificaciones de WhatsApp a través de Green API.

```
<?php
namespace App\Services;
use App\Utilities\PhoneNormalizer;
use Illuminate\Support\Facades\Http;
/**
 * Servicio de envío de mensajes WhatsApp a través de Green API.
 *
 * Green API permite enviar mensajes de WhatsApp usando tu propio número.
 * Endpoint: https://api.green-api.com/waInstance{id}/sendMessage/{token}
 *
 * Plan Developer (gratis): hasta 3 chats activos.
 *
 * Clase instanciable, resuelta automáticamente por el contenedor de
 * servicios de Laravel e inyectada en los Jobs que la requieran.
 */
class WhatsAppService
{
    private string $idInstance;
    private string $apiToken;
    private string $apiUrl;
    public function __construct()
    {
        $this->idInstance = config('services.whatsapp.id_instance', '');
        $this->apiToken = config('services.whatsapp.api_token', '');
        $this->apiUrl = config('services.whatsapp.api_url', 'https://api.green-api.com');
    }
}
```



```

/**
 * Envía el mensaje de aprobación de solicitud de adopción.
 *
 * @throws \RuntimeException Si la API devuelve un error.
 */
public function enviarAprobacion(
    string $telefono,
    string $nombreUsuario,
    int $solicitudId,
    string $nombreMascota
): void {
    $numero = PhoneNormalizer::normalize($telefono);
    $mensaje = $this->buildMensajeAprobacion(
        $nombreUsuario, $solicitudId, $nombreMascota
    );
    $this->send($numero, $mensaje);
}

/**
 * Envía el mensaje de rechazo de solicitud de adopción.
 *
 * @throws \RuntimeException Si la API devuelve un error.
 */
public function enviarRechazo(
    string $telefono,
    string $nombreUsuario,
    int $solicitudId,
    string $nombreMascota,
    string $motivo,
    int $etapa = 1
): void {

```

```

$numero = PhoneNormalizer::normalize($telefono);
// Sanitizar motivo: solo texto plano, sin HTML ni caracteres de control
$motivoSanitizado = strip_tags(trim($motivo));
$motivoSanitizado = $motivoSanitizado ?: 'No se especificó un motivo.';
$mensaje = $this->buildMensajeRechazo(
    $nombreUsuario, $solicitudId, $nombreMascota, $motivoSanitizado, $etapa
);
$this->send($numero, $mensaje);
}

// — Builders de mensajes —————
private function buildMensajeAprobacion(
    string $nombre, int $id, string $mascota
): string {
    return " 🦋 CAEFUG — Centro de Atención Emergente de Fauna Urbana\n\n"
        . "Estimado/a {$nombre},\n\n"
        . "Nos complace informarle que su solicitud de adopción N° {$id} "
        . "para la mascota {$mascota} ha sido APROBADA ✅.\n\n"
        . " 📋 Próximos pasos:\n"
        . "• Acérquese a nuestras instalaciones para firmar el acta de adopción definitiva.\n"
        . "• Recuerde llevar su cédula de identidad original.\n\n"
        . " 📍 Dirección: Guaranda, Bolívar — Ecuador\n"
        . "🕒 Horario: Lunes-Martes-Jueves-Viernes; 7:00 — 15:00\n\n"
        . "Gracias por darle un hogar a quien más lo necesita. 🐾\n\n"
        . "Gobierno Autónomo Descentralizado del Cantón Guaranda\n"
        . "Centro de Atención Emergente de Fauna Urbana — CAEFU";
}

private function buildMensajeRechazo(
    string $nombre, int $id, string $mascota, string $motivo, int $etapa = 1
): string {

```

```

$url = config('app.frontend_url', config('app.url'));
if ($etapa === 2) {
    return " 📄 CAEFUG — Centro de Atención Emergente de Fauna Urbana\n\n"
        . "Estimado/a {$nombre},\n\n"
        . "Le informamos que tras la visita domiciliaria, su solicitud de adopción N°
{$id} "
        . "para la mascota {$mascota} ha sido RECHAZADA.\n\n"
        . " 🚫 Detalles del Rechazo:\n{$motivo}\n\n"
        . " 💡 ¿Qué puede hacer?\n"
        . "Una vez que mejore las condiciones, puede solicitar una Re-evaluación de
Vivienda "
        . "desde la sección \"Solicitudes\" de nuestra web.\n\n"
        . " 🌐 {$url}\n\n"
        . "Si tiene alguna consulta, no dude en contactarnos.\n\n"
        . "Gobierno Autónomo Descentralizado del Cantón Guaranda\n"
        . "Dirección de Gestión Ambiental — CAEFUG";
}

return " 📄 CAEFUG — Centro de Atención Emergente de Fauna Urbana\n\n"
    . "Estimado/a {$nombre},\n\n"
    . "Le informamos que su solicitud de adopción N° {$id} "
    . "para la mascota {$mascota} ha sido RECHAZADA en la etapa de revisión
documental.\n\n"
    . " 🚫 Motivo:\n{$motivo}\n\n"
    . " 💡 ¿Qué puede hacer?\n"
    . "Puede corregir la información y reenviar su solicitud "
    . "en la sección \"Solicitudes\" de nuestra web.\n\n"
    . " 🌐 {$url}\n\n"
    . "Si tiene alguna consulta, no dude en contactarnos.\n\n"
    . "Gobierno Autónomo Descentralizado del Cantón Guaranda\n"
    . "Dirección de Gestión Ambiental — CAEFUG";

```

```

}

// — Envío HTTP a Green API —————
/**
 * Envía un mensaje de texto vía Green API (sendMessage).
 *
 * Endpoint: POST /waInstance{id}/sendMessage/{token}
 * Body: { "chatId": "593XXXXXXXXX@c.us", "message": "..." }
 *
 * @param string $telefono Número en formato internacional (solo dígitos, ej:
593987654321).
 * @param string $mensaje Cuerpo del mensaje de texto.
 * @throws \RuntimeException Si la respuesta de la API indica error.
 */
private function send(string $telefono, string $mensaje): void
{
    $url = "{$this->apiUrl}/waInstance{$this->idInstance}/sendMessage/{$this->apiToken}";
    $payload = [
        'chatId' => "{$telefono}@c.us",
        'message' => $mensaje,
    ];
    $response = Http::timeout(10)->post($url, $payload);
    if ($response->failed()) {
        $errorBody = $response->body();
        throw new \RuntimeException(
            "Green API error [{$response->status()}]: {$errorBody}"
        );
    }
}
}

```

4.4. Pruebas

4.4.1 Propósito

El enfoque principal de las pruebas es que el sistema debe cumplir con las necesidades planteadas en las historias de usuario, permitir la detección de distintos aspectos de calidad del software y posibles problemas de seguridad, con el fin de garantizar que el producto final sea de calidad, sin errores y resulte intuitivo a sus usuarios.

4.4.2 Diseño y desarrollo de casos de prueba

- a) **Pruebas de funcionalidad:** Se verificó que los procesos centrales del sistema operaran según lo establecido, entre ellos el registro y la autenticación de usuarios, gestión del catálogo de mascotas, envío y seguimiento de solicitudes de adopción a través de sus distintas etapas y envío de notificaciones automáticas por WhatsApp y correo electrónico.
- b) **Pruebas de rendimiento:** Se evaluó la respuesta del sistema bajo una carga operativa real en consideración del volumen de mascotas, usuarios y solicitudes de adopción gestionadas por el CAEFU, se verificó que las consultas a la base de datos, optimizadas mediante paginación y caché en módulos administrativos, respondan en tiempos adecuados.
- c) **Pruebas de seguridad:** Se validó la encriptación de contraseñas, protección de rutas mediante autenticación en tokens con Laravel Sanctum, separación de permisos entre roles de administrador y usuario, prevención de ataques de fuerza bruta por límites de intentos (throttling) en los endpoints de inicio de sesión y recuperación de contraseña.
- d) **Pruebas de usabilidad:** Se midió la facilidad de uso para el administrador del CAEFU y para los ciudadanos que desean adoptar una mascota. El diseño del sistema contribuyó a que resultara intuitivo a ambos perfiles de usuario.
- e) **Pruebas de integración:** Se evaluó la interoperabilidad entre los distintos componentes del sistema, comunicación entre el frontend desarrollado en React y API REST construida en Laravel, integración con servicios externos como Google OAuth y Green API.

Tabla 15.*Stakeholders*

Stakeholders	Descripción
Equipo de evaluación	Encargado de probar el sistema y verificar que los resultados obtenidos correspondan con lo definido.
Administrador del CAEFU	Responsable de la operación integral del sistema desde el panel administrativo. Sus funciones son el registro de mascotas, revisión y resolución de solicitudes de adopción, gestión de usuarios y generación de reportes estadísticos.
Usuario adoptante	Explora el catálogo de mascotas, crea una cuenta en la plataforma, envía solicitudes, hace seguimiento al estado de estas y comparte testimonios sobre su experiencia con el proceso.

Nota. Involucrados en el sistema.**Tabla 16.***Gestor de pruebas*

Gestor de pruebas	Detalle
Navegadores	Google Chrome, Mozilla Firefox, Microsoft Edge
Sistemas Operativos	Windows
Servidor Web	Nginx
Lenguajes de Programación	PHP, JavaScript
Framework	Backend (Laravel), Frontend (React)
Gestor de base de datos	PostgreSQL

Nota. Entorno de trabajo para realizar pruebas de desarrollo y funcionalidad.

4.4.4. Pruebas unitarias

Tabla 17.

Registro y autenticación de usuarios

Componente	Ítem a evaluar	Sí	No
Login	Verifica que el correo electrónico no conste como registrado previamente.	X	
Login	La gestión de roles entre administrador y usuario, se realiza de manera adecuada.	X	
Login	Verifica que el número de cédula sea único dentro del sistema.	X	
Login	La contraseña exige mayúscula, minúscula y número, con un mínimo de ocho caracteres.	X	

Nota. Pruebas de registro y autenticación de usuarios, desde la interfaz cliente.

Tabla 18.

Gestión de mascotas

Componente	Ítem a evaluar	Sí	No
Filtros de búsqueda	Verifica el funcionamiento de la búsqueda filtrada por categoría, tamaño, género y edad.	X	
Formulario	Permite crear y actualizar una mascota, con la carga de múltiples fotografías y del historial médico.	X	

Nota. Pruebas de funcionalidad y validación en el módulo de gestión de mascotas.

Tabla 19.*Gestión de solicitudes de adopción*

Componente	Ítem a evaluar	Sí	No
Barra de búsqueda	Verifica el funcionamiento de la búsqueda por identificador, cédula o nombre del solicitante.	X	
Flujo de revisión	Permite avanzar el estado de la solicitud entre pendiente, en visita, aprobada y rechazada, y marca la mascota como no disponible solo en la aprobación final.	X	

Nota. Pruebas de funcionalidad y seguridad en el módulo de solicitudes de adopción.

Tabla 20.*Notificaciones y reportes*

Componente	Ítem a evaluar	Sí	No
Notificaciones	Verifica el envío automático del mensaje de WhatsApp al cambiar el estado de una solicitud, sea aprobada o rechazada.	X	
Reportes administrativos	Constata que el panel muestra estadísticas correctas de mascotas, adopciones, usuarios y solicitudes, comparadas con el periodo anterior.	X	

Nota. Pruebas de integración y rendimiento en el módulo de notificaciones y reportes.

4.5 Entrega

a) Entregables del proyecto

Como resultado de esta fase, se entregarán los siguientes productos:

- Sistema web funcional para la gestión y el seguimiento de adopciones de mascotas.
- Código fuente del sistema organizado la arquitectura N-capas.
- Base de datos con sus tablas y relaciones ya estructuradas.
- Manual de usuario dirigido al administrador del CAEFU y al usuario adoptante.

b) Capacitación y retroalimentación

Junto con la entrega se capacitará al personal del CAEFU en el manejo del sistema y sus funcionalidades operativas para que puedan familiarizarse con la plataforma y brindar una retroalimentación inmediata, que podrá considerarse para futuras mejoras. Con esta entrega queda instalada una solución tecnológica funcional, confiable y acorde a las necesidades del centro..

CONCLUSIONES

- El diagnóstico de la situación actual en los procesos de gestión y adopción de mascotas del Centro de Atención Emergente de Fauna Urbana (CAEFU) de Guaranda se realizó mediante una guía de observación y una entrevista al personal responsable. Los resultados mostraron que la institución cuenta con procedimientos para el registro de mascotas, el control de historiales clínicos, la gestión de adopciones y el seguimiento post-adopción. La información se administra en registros físicos, que limita la centralización, consulta y difusión de datos; no existe mecanismos digitales para publicación de mascotas disponibles y registro de ciudadanos interesados en adoptar; estos aspectos justifican la necesidad de desarrollar un sistema web que optimice la gestión de información y fortalezca la interacción con la ciudadanía.
- El desarrollo de la arquitectura del sistema web bajo el patrón N-capas, permitió separar las responsabilidades entre la interfaz de usuario en React, lógica de negocio en Laravel y persistencia de datos en PostgreSQL. Esta separación facilita el mantenimiento del sistema, incorporación de nuevas funcionalidades y escalabilidad de cada capa de forma independiente, sin que un cambio en la interfaz comprometa la lógica del backend o la integridad de los datos.
- El modelado de la base de datos representado en el diagrama de clases y en las migraciones de Laravel, permitió estructurar la información del proceso de adopción, que inicia en el registro de solicitud, seguimiento posterior, mediante entidades como SolicitudAdopcion, VisitaDomiciliaria, DocumentoSolicitud e HistorialMedico.
- La ejecución de pruebas unitarias, funcionalidad, rendimiento, seguridad e integración validó que los módulos críticos del sistema como autenticación, gestión de mascotas, solicitudes de adopción y notificaciones a través de WhatsApp, operan acorde a lo instituido en los requerimientos. De este modo, se avala que el sistema es una herramienta confiable para respaldar el proceso de adopción del CAEFU, dado que no se detectaron fallas en los componentes evaluados.

RECOMENDACIONES

- Se recomienda implementar el sistema web desarrollado para digitalizar los procesos de registro y adopción de mascotas, con el fin de centralizar la información institucional y agilizar la gestión de las solicitudes de adopción. La plataforma incorpora un catálogo de mascotas disponibles, un módulo de registro de usuarios y herramientas para dar seguimiento a las adopciones y genera reportes para mejorar la eficiencia operativa de la institución y promover la participación ciudadana en la adopción responsable y protección animal.
- Es importante mantener la separación de capas señalada en la arquitectura del sistema en futuras ampliaciones del proyecto, de esta manera, cualquier incorporación de nuevas funcionalidades respetará la independencia entre el frontend, backend y capa de persistencia, en consecuencia, se preserva la facilidad de mantenimiento alcanzada.
- Es recomendable completar los componentes del modelo de datos que según el levantamiento de información permanecen pendientes o incompletos, como el modelo Configuración y factory de usuarios, esta última con los nombres de columna en español que utiliza el modelo User, con el fin de evitar fallas en la generación de datos de prueba.
- Es importante ampliar la cobertura de pruebas hacia escenarios de carga y concurrencia más exigentes, en especial sobre los módulos de solicitudes de adopción y notificaciones por WhatsApp, e incorporar pruebas automatizadas que permitan repetir la validación del sistema de forma periódica ante cada nueva actualización del código.

BIBLIOGRAFÍA

- Aldalur, I. (2024). Web Augmentation: A systematic mapping study. *Science of Computer Programming*, 232, 103045. <https://doi.org/10.1016/j.scico.2023.103045>
- Araujo-Inastrilla, C., Roche-Madrigal, M., & García-Savón, Y. (2023). Diseño de base de datos para el departamento de Sistemas de Información en Salud, La Habana 2021. *Revista Información Científica*, 102(4099), 1–11. <http://www.revinfoinformatica.sld.cu/index.php/ric/article/view/4099>
- Armijos, Lady, Velez, C., & Lojan, E. (2024). Estudio de la adopción de metodologías ágiles en proyectos de desarrollo de software en la región 7 del Ecuador. *Espacios*, 45(04), 73–84. <https://doi.org/10.48082/espacios-a24v45n04p06>
- Bautista-Villegas, E. (2022). Metodologías ágiles XP y Scrum, empleadas para el desarrollo de páginas web, bajo MVC, con lenguaje PHP y framework Laravel. *Revista Amazonía Digital*, 1(1), e168. <https://doi.org/10.55873/rad.v1i1.168>
- Bravo-Huivin, E., Cieza-Mostacero, S., Flores-Rodriguez, L., & Uceda-Davila, L. (2022). Revisión Sistemática de la Literatura sobre Metodologías Ágiles en el Desarrollo de Software aplicadas a la Gestión de Proyectos. *Revista Ibérica de Sistemas e Tecnologías de Informação*, (N.º E54), 314–326.
- Castillo, M., & Guaña, E. (2024). Kanban: Una metodología ágil para la gestión eficiente del flujo de trabajo en el desarrollo de software, una revisión sistemática. *Revista Ingenio Global*, 3(1), 17–28. <https://doi.org/10.62943/rig.v3n1.2024.68>
- Código Orgánico de La Economía Social de Los Conocimientos, Creatividad e Innovación, Pub. L. 899, Suplemento – Registro Oficial N° 899 113 (2016). <https://www.gobiernoelectronico.gob.ec/wp-content/uploads/2018/10/Codigo-Organico-de-la-Economia-Social-de-los-Conocimientos-Creatividad-e-Innovacion.pdf>

- Concepción, D., Muñoz, L., Villarreal, V., & Pardo, C. (2022). A Systematic Review of the Software Architectures for the Development of Mobile Applications in Education. *2022 8th International Engineering, Sciences and Technology Conference (IESTEC)*, 231–237. <https://doi.org/10.1109/IESTEC54539.2022.00042>
- Cruz, G., Delgado, L., Ponce, B., & Marcillo, M. (2022). Riesgos de seguridad de los datos en la web. *Journal TechInnovation*, 1(2), 43–49. <https://doi.org/10.47230/Journal.TechInnovation.v1.n2.2022.43-49>
- Cruz-Hernández, J., & Valencia-Esquivel, D. (2023). *Sistema de gestión de adopción animal para la Fundación OPAM* [Trabajo de Grado.]. Universidad Católica de Colombia.
- Díaz, A., Gómez, F., Figueroa, C., & Durán, A. (2023). Aprendizaje del lenguaje SQL en el programa educación, mención informática de la Universidad del Zulia. *SUMMA*, 5(2), 1–11. <https://doi.org/10.47666/summa.5.2.10>
- Ebenezer, J., Akinseinde, S., Kalesanwo, O., Adeagbo, M., Oladapo, K., Awoseyi, A., & Kasali, F. (2022). Application of Artificial Intelligence in User Interfaces Design for Cyber Security Threat Modeling. In *Software Usability*. IntechOpen. <https://doi.org/10.5772/intechopen.96534>
- Ehsan, A., Abuhaliqa, M., Catal, C., & Mishra, D. (2022). RESTful API Testing Methodologies: Rationale, Challenges, and Solution Directions. *Applied Sciences*, 12(9), 4369. <https://doi.org/10.3390/app12094369>
- Enríquez, F., Fierro, S., Flores, B., Imbaquingo, D., & Michelena, J. (2023). Impacto del patrón modelo vista controlador (MVC) en la seguridad, interoperabilidad y usabilidad de un sistema informático durante su ciclo de vida. *EASI: Ingeniería y Ciencias Aplicadas En La Industria*, 2(1), 11–16. <https://doi.org/10.53591/easi.v2i1.2043>
- Espíndola, D., Fernández, D., Ojeda, L., Ramírez, A., & Pioli, L. (2024). Implementación de una app en una red local y comparación de su arquitectura en

- dos y tres capas. *XXX Congreso Argentino de Ciencias de La Computación*, 639–648. <http://sedici.unlp.edu.ar/handle/10915/177542>
- Espinoza-Marca, B., & Cieza-Mostacero, S. (2024). Aplicación Web Móvil en el Control de Incidencias Delictivas. *Revista Ibérica de Sistemas e Tecnologías de Informação*, (E73), 58–74.
- Estepa, J., Cajiao, M. N., & Monsalve, S. (2023). Gestión poblacional canina y felina en el marco One Welfare: Una mirada retrospectiva Bogotá 2004 a 2021. *Revista MVZ Córdoba*, 28(1), e2925. <https://doi.org/10.21897/rmvz.2925>
- Feriz, D., Cardozo, E., Valencia, N., Dueñas, G., & Cardozo, P. (2024). Agile Educación: Herramientas y Beneficios de la Aplicación de Metodologías Ágiles en la Enseñanza Universitaria. *ConCiencia*, 14, 127–144.
- Flores-Cerna, F., Sanhueza-Salazar, V., Valdés-González, H., & Reyes-Bozo, L. (2021). Metodologías ágiles: un análisis de los desafíos organizacionales para su implementación. *Revista Científica*, 43(1), 38–49. <https://doi.org/10.14483/23448350.18332>
- Fucci, D., Alégroth, E., & Axelsson, T. (2022). When traceability goes awry: An industrial experience report. *Journal of Systems and Software*, 192, 111389. <https://doi.org/10.1016/j.jss.2022.111389>
- Gaete, J., Villarroel, R., Figueroa, I., Cornide-Reyes, H., & Muñoz, R. (2021). Enfoque de aplicación ágil con Serum, Lean y Kanban. *Ingeniare. Revista Chilena de Ingeniería*, 29(1), 141–157. <https://doi.org/10.4067/S0718-33052021000100141>
- Gamarra, M., Morales, F., Nuñez, D., & Ramirez, J. (2026). Using templates in ConectaPet: A web-based system for dog and cat adoptions and donations. *TechNoLogica*, 1(1), 1–13. <https://doi.org/10.17268/tech1.1.2026.03>
- Garcés-Giraldo, L., Benjumea-Arias, M., Cardona-Acevedo, S., Bermeo-Giraldo, C., Valencia-Arias, A., Patiño-Vanegas, C., Moreno-López, G., & García, R. (2022). Uso de inteligencia artificial en gestión de la información: una revisión

- bibliométrica. *Revista Ibérica de Sistemas e Tecnologias de Informação*, (E54), 506–517.
- García, L. (2025). *Diseño y desarrollo de un prototipo de aplicación web para el reporte del encuentro de mascotas perdidas en Bogotá* [Tesis]. Universidad Católica de Colombia.
- Golmohammadi, A., Zhang, M., & Arcuri, A. (2024). Testing RESTful APIs: A Survey. *ACM Transactions on Software Engineering and Methodology*, 33(1), 1–41. <https://doi.org/10.1145/3617175>
- Gómez-Álvarez, M., & Zapata-Jaramillo, C. (2022). Una propuesta de clasificación de juegos con propósito educativo para ingeniería de software. *Ingeniare. Revista Chilena de Ingeniería*, 30(2), 239–254. <https://doi.org/10.4067/S0718-33052022000200239>
- González, Y., MC lenan, J., & Abrego, A. (2022). Los proyectos y las metodologías. Modelos de procesos: Ciclos de vida de desarrollo de software. *Revista Semilla Científica*, (3), 185–194. <https://revistas.umecit.edu.pa/index.php/sc/article/view/1090>
- Guambo, M., Ojeda, G., Pachay, C., & Peñafiel, J. (2025). Impacto del abandono de mascotas en la salud pública y biodiversidad urbana. *Revista UGC*, 3(52), 178–184. <https://universidadugc.edu.mx/ojs/index.php/rugc/article/view/168>
- Guanoluisa, E., Jhomara, P., & Cadena, J. (2024). Herramientas eficientes de desarrollo de software más utilizados en Ecuador: caso de estudio aplicado al Club de Robótica “BOT’S UTC.” *Technology Rain Journal*, 3(1), 1–22. <https://doi.org/10.55204/trj.v3i1.e24>
- Guzmán, C., Jaramillo, J., Pulla, T., & Sarmiento, S. (2024). Evolución de las Metodologías de Desarrollo de Software en la Gestión Académica. *Ciencia Latina Revista Científica Multidisciplinar*, 8(4), 2468–2475. https://doi.org/10.37811/cl_rcm.v8i4.12490

- Haro, H., Cárdenas, K., Córdor, B., & Yopez, G. (2025). Enfoque de aprendizaje colaborativo a través de SCRUM: Implementación de enseñanza en Eduplanet. *Revista Social Fronteriza*, 5(1), e-599. [https://doi.org/10.59814/resofro.2025.5\(1\)599](https://doi.org/10.59814/resofro.2025.5(1)599)
- Hidalgo, J., & Usiña, A. (2025). *Sistema Web de Gestión para adopción y recuperación de mascotas extraviadas para la Fundación Fauna Urbana Ibarra* [Tesis, Pontificia Universidad Católica del Ecuador]. <https://repositorio.puce.edu.ec/items/e2e266a3-425a-4e02-aa83-d42b59483b68>
- Hossman, D., Aduni, R., & Jelani, M. (2024). Exploring Traceability Techniques on Software Engineering: A Review and Future Directions. *Journal of Applied Science technology and Computing*, 2(1), 32–42. <https://publisher.uthm.edu.my/ojs/index.php/jastec/article/view/23632/7611>
- Ilyas, Q., Ahmad, M., Zaman, N., Alshamari, M., & Ahmed, I. (2022). Localized Text-Free User Interfaces. *IEEE Access*, 10, 2357–2371. <https://doi.org/10.1109/ACCESS.2021.3139525>
- Jones, C., Ascenzi, L., Piñones, J., & Caffaratti, M. (2026). Diseño de un dispositivo pedagógico para la enseñanza de SQL. *Revista Científica DUTI*, (3), 65–76. <https://duti.unr.edu.ar/index.php/revista/article/view/48>
- Ley Orgánica de Protección de Datos Personales, Pub. L. 459, Registro Oficial Suplemento 459 de 26-may.-2021 38 (2018). www.lexis.com.ec
- Martin-Lopez, A., Segura, S., & Ruiz-Cortés, A. (2022). Online testing of RESTful APIs: promises and challenges. *Proceedings of the 30th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, 408–420. <https://doi.org/10.1145/3540250.3549144>
- Mieles, A., & Tipanluisa, J. (2022). *Sistema web y aplicación móvil para la gestión de promociones, adopciones y red colaborativa para publicación de mascotas extraviadas de la Clínica Hospital*. Universidad Agraria del Ecuador .

- Molina, S. (2026). *Mecanismos de Seguridad para el Diseño de Desarrollo Web. Buenas prácticas de Seguridad*. [Universitat Oberta de Catalunya]. <https://openaccess.uoc.edu/server/api/core/bitstreams/5504f3a6-6970-4d7b-b6e7-843e124610a0/content>
- Monterrubio-Hernández, E. (2024). Hardware. *Con-Ciencia Serrana Boletín Científico de La Escuela Preparatoria Ixtlahuaco*, 6(11), 6–8. <https://doi.org/10.29057/ixtlahuaco.v6i11.11967>
- Morales-Carrillo, J., Cedeño-Valarezo, L., Cajape-Bravo, J., & Ormaza-Calderón, J. (2022). Metodologías de desarrollo de software y su ámbito de aplicación: Una revisión sistemática. *Revista Ibérica de Sistemas e Tecnologias de Informação*, (E47), 29–45. https://www.researchgate.net/profile/Luis-Cristobal-Cedeno-Valarezo/publication/374165932_Metodologias_de_desarrollo_de_software_y_su_ambito_de_aplicacion_Una_revision_sistemica/links/693997b9a1fd01798906a43c/Metodologias-de-desarrollo-de-software-y-su-ambito-de-aplicacion-Una-revision-sistemica.pdf
- Mota, D., Calderón, N., Lezama, K., Sepiurka, L., & Garcia, R. (2021). Abandonment of dogs in Latin America: Strategies and ideas. *Veterinary World*, 14(9), 2371–2379. <https://doi.org/10.14202/vetworld.2021.2371-2379>
- Ozkan, N., Bal, S., Erdogan, T., & Gök, M. (2022). Scrum, Kanban or a Mix of Both? A Systematic Literature Review. *Conferencia Sobre Ciencias de La Computación y Sistemas de Inteligencia (FedCSIS)*, 883–893. <https://doi.org/10.15439/2022F143>
- Pacheco, J., Hernández, B., Toro, M., & Batista, F. (2023). Base de datos automatizada para el control de la literatura docente. *Revista Cubana de Informática Médica*, 15(1), e553. http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1684-18592023000100007&lng=es&nrm=iso&tlng=es
- Pacheco, M., Sarmiento, A., Salazar, V., & Mendoza, L. (2022). Software libre vs Software privativo: su implicación en la educación del siglo XXI. *Revista*

- Científica Multidisciplinaria Ogma*, 1(2), 62–73.
<https://doi.org/10.69516/16wtyc06>
- Paredes, J., & Páez, D. (2024). Análisis comparativo de metodologías de gestión de proyectos en construcción: Cascada, Ágil y Lean. *Revista Científica Multidisciplinaria G-Nerando*, 5(2), 310–334.
<https://doi.org/10.60100/rcmg.v5i2.273>
- Pata, P., Benites, L., Cuenca, C., Nazareno, I., & Sacón, H. (2024). Mejorando el bienestar animal y promoviendo la sostenibilidad a través de la tecnología: desarrollo de una aplicación móvil para la adopción de mascotas. *Revista Científica Multidisciplinaria G-Nerando*, 5(1), 703–727.
<https://doi.org/10.60100/rcmg.v5i1.220>
- Pelaiz, S., & Delgado, M. (2024). Ciclo de vida de desarrollo seguro de software en la empresa datys. *Revista Cubana De Administración Pública Y Empresarial*, 8(2), e306. <https://apye.esceg.cu/index.php/apye/article/view/306>
- Pocladova, V., Menchón, M., Olsowy, V., Antonelli, L., & Thomas, P. (2024). Una revisión de Trazabilidad de Requerimientos en el contexto de Gestión de Proyectos. *XXX Congreso Argentino de Ciencias de La Computación*, 639–648.
- Quezada-Sarmiento, P., Alban- Cartuche, O., López -Pilataxi, L., Gonzaga-Tillaguango, H., Espinosa-Lara, E., & Ludeña-Reyes, L. (2023). Factores de riesgo, amenazas, vulnerabilidades y defensa en aplicaciones web turísticas. *Revista Ibérica de Sistemas e Tecnologías de Informação*, (N.º E56), 104–112.
https://www.researchgate.net/profile/Pablo-Quezada-Sarmiento/publication/365654917_Factores_de_riesgo_amenazas_vulnerabilidades_y_defensa_en_aplicaciones_web_turisticas/links/6629a6fb08aa54017ab47ce8/Factores-de-riesgo-amenazas-vulnerabilidades-y-defensa-en-aplicaciones-web-turisticas.pdf
- Quisaguano, L., Lasluisa, A., Oto-Usuño, L., & Esquivel, G. (2024). Optimización del desarrollo Front-end aplicando frameworks de diseño web: Bootstrap, Foundation

- y Bulma. *Revista UTCiencia*, 11(1), 17–31.
<https://investigacion.utc.edu.ec/index.php/utciencia/article/view/650>
- Quispe, J. (2022). Diseño de interfaces de sistemas interactivos utilizando técnicas de machine learning: una revisión del diseño y la usabilidad. *Interfases*, (016), 202–214. <https://doi.org/10.26439/interfases2022.n016.6028>
- Rana, M., & Saleh, O. (2022). High assurance software architecture and design. In *System Assurances* (pp. 271–285). Elsevier. <https://doi.org/10.1016/B978-0-323-90240-3.00015-1>
- Reyes, J., Fuertes, W., Arévalo, P., & Macas, M. (2022). An Environment-Specific Prioritization Model for Information-Security Vulnerabilities Based on Risk Factor Analysis. *Electronics*, 11(9), 1334. <https://doi.org/10.3390/electronics11091334>
- Rodríguez, E., Carrillo, M., & Carrillo, H. (2026). Arquitectura híbrida EfficientNet y Vision Transformer para detección multiplataforma de patologías agrícolas. *Revista de Investigación En Tecnologías de La Información*, 14(33), 111–125. <https://doi.org/10.36825/RITI.14.33.009>
- Salcedo, M., Tapia, S., Benítez, A., González, J., & Olivares, S. (2023). El desarrollo de un sistema tecnológico de comunicados en la Unidad Académica de Economía. *Revista Electrónica Sobre Tecnología, Educación y Sociedad*, 10(20). <https://www.ctes.org.mx/index.php/ctes/article/view/812>
- Sassa, A., Almeida, I., Pereira, T., & Oliveira, M. (2023). Scrum: A Systematic Literature Review. *International Journal of Advanced Computer Science and Applications*, 14(4), 173–181. <https://doi.org/10.14569/IJACSA.2023.0140420>
- Suescún, E., Tabares, M., González, L., & Vásquez, M. (2024). Hacia un Modelo de Gobierno de APIs, Mapeo Sistemático de la Literatura. *Revista Perspectivas*, 6(2), 71–83. <https://doi.org/10.47187/perspectivas.6.2.219>

- Taj, I., & Farooq, U. (2023). Towards Machine Learning-Based FPGA Backend Flow: Challenges and Opportunities. *Electronics*, 12(4), 935. <https://doi.org/10.3390/electronics12040935>
- Varelas, M., & Castro, K. (2026). *Desarrollo de un sistema web para la gestión de información de la fundación Adopta Club Púrpura de la ciudad de Dosquebradas (Rescata por amor)* [Tesis, Universidad Católica de Pereira]. <https://repositorio.ucp.edu.co/entities/publication/c270893b-96e7-4d74-a591-36531f865625>
- Yunanto, A., Hardiansyah, F., Anantha, A., Afridian, M., & Arifiani, S. (2022). Development of sandbox components with microservices architecture and design patterns in games. *Procedia Computer Science*, 197, 354–361. <https://doi.org/10.1016/j.procs.2021.12.150>

ANEXOS

Anexo 1. Cronograma (Gantt)

||
||
||

Autenticación y control de acceso																	
Pruebas unitarias — iteración 1																	
FASE 4 · Codificación — Iteración 2: gestión de mascotas																	
Módulo de registro de mascotas																	
Catálogo web (fotos, especie, estado de salud)																	
Pruebas unitarias — iteración 2																	
FASE 5 · Codificación — Iteración 3: solicitudes de adopción																	
Módulo de solicitudes de adopción																	
Aprobación / rechazo de solicitudes																	
Seguimiento post-adopción																	
Pruebas unitarias — iteración 3																	
FASE 6 · Codificación — Iteración 4: reportes y administración																	
Módulo de reportes estadísticos																	
Administración del refugio (página inicio, testimonios)																	
Pruebas unitarias — iteración 4																	
FASE 7 · Pruebas integrales y entrega																	
Pruebas de integración y regresión																	
Pruebas de aceptación con personal CAEFU																	
Correcciones y ajustes finales																	
Entrega y documentación final																	
Duración total: 16 semanas (4 meses)	MES 1				MES 2				MES 3				MES 4				

Nota. Diagrama de seguimiento de actividades. **Elaborado por:** Muyulema C, 2026

Anexo 2. Presupuesto Ejecutado

Tabla 21.

Presupuesto estimado - (CAEFU – Guaranda)

Nº	Descripción del rubro	Cantidad	Unidad	Valor unitario (USD)	Subtotal (USD)
1	Servicio de base de datos PostgreSQL (cloud)	4	Mes	15,00	60,00
2	Registro de dominio web	1	Año	30,00	30,00
3	Servicio API WhatsApp Business	4	Mes	20,00	80,00
4	Computadora portátil	1	Equipo	900,00	900,00
5	Visitas al CAEFU (transporte)	8	Visita	5,00	40,00
6	Impresión de instrumentos (guías y fichas)	3	—	3,00	9,00
7	Materiales de oficina (papelería)	1	—	20,00	20,00
8	Empastado / encuadernación del informe final	3	—	15,00	45,00
9	Impresión y reproducción del informe escrito	3	—	12,00	36,00
10	CD/USB con entregables digitales del proyecto	2	unidad	3,00	6,00
Imprevistos (5% sobre subtotal de rubros)					61,30
TOTAL GENERAL DEL PROYECTO (USD)					1.287,30

Anexo 3. Carta de aceptación de la organización donde se realizó el trabajo de integración curricular.



Guaranda, 06 de abril de 2026

Dr. Inti Andrés Yumbay Taris

ALCALDE DEL GOBIERNO AUTÓNOMO DESCENTRALIZADO MUNICIPAL DE
GUARANDA

Presente. –

De mi consideración:

*Dr. Yumbay Taris
Por analista
07/04/2026*

Yo, **CRISTOFER FABIÁN MUYULEMA CUVI**, portador de la cédula de ciudadanía N.º **0202161063**, egresado de la **UNIVERSIDAD ESTATAL DE BOLÍVAR**, me dirijo a usted de la manera más comedida con la finalidad de solicitar el respectivo **permiso para realizar el proceso de titulación**, el mismo que se llevará a cabo en el **Refugio de Animales de Guaranda**.

El presente trabajo de titulación tiene como tema:

“SISTEMA WEB PARA LA GESTIÓN Y ADOPCIÓN DE MASCOTAS EN EL REFUGIO DE ANIMALES DE GUARANDA”, cuyo objetivo principal es contribuir al mejoramiento de los procesos de gestión, control y adopción de mascotas, fortaleciendo así el bienestar animal dentro del cantón.

Por lo expuesto, solicito muy comedidamente se me autorice el acceso y desarrollo de las actividades necesarias dentro de las instalaciones del mencionado refugio, para la correcta ejecución de este proyecto académico.

Agradezco de antemano la atención brindada a la presente, y quedo a la espera de una respuesta favorable.

Atentamente,

CRISTOFER FABIÁN MUYULEMA CUVI
CEL. 0985577630



26026.2359

Anexo 4. Instrumentos de recopilación de datos - Guía de entrevista

UNIVERSIDAD ESTATAL DE BOLÍVAR FACULTAD DE CIENCIAS ADMINISTRATIVAS, GESTIÓN EMPRESARIAL E INFORMÁTICA	GUÍA DE ENTREVISTA	Código: GE-CAEFU-01 Versión: 1.0
	CAEFU – Guaranda	Fecha de aplicación: 12 / 01 / 2026

GUÍA DE ENTREVISTA PARA EL DIAGNÓSTICO DE LOS PROCESOS DE GESTIÓN Y ADOPCIÓN DE MASCOTAS EN EL CAEFU

I. DATOS GENERALES

Institución:	Centro de Atención Emergente de Fauna Urbana (CAEFU) – Guaranda	Fecha:	12/01/2026
Nombre del entrevistado:	CARLOS DANILO VALVERDE	Cargo:	TENICO
Entrevistador:	Cristofer Moyulema	Hora:	9:00 AM
Lugar:	CAEFU		

II. OBJETIVO

Recopilar información sobre los procesos actuales de gestión y adopción de mascotas en el CAEFU de Guaranda, con el fin de identificar las necesidades funcionales que orientarán el diseño del sistema web.

III. INSTRUCCIONES

La presente entrevista tiene carácter académico y su información será utilizada exclusivamente para el desarrollo del proyecto de titulación, por lo cual se solicita responder con la mayor precisión posible. Las respuestas serán registradas por el entrevistador en los espacios destinados para el efecto.

IV. CUESTIONARIO

1	¿Cómo se realiza actualmente el registro de las mascotas que ingresan al CAEFU? ¿Qué información se registra de cada animal?
	SE REALIZA UNA FICHA INDIVIDUAL POR CADA UNA DE LAS MASCOTAS QUE INGRESAN CON FOTO Y EL ESTADO EN LA QUE INGRESAN
2	¿De qué manera se lleva el control de las mascotas disponibles para adopción y de aquellas que ya han sido adoptadas?
	DE LAS MASCOTAS SE LLEVA UNA HISTORIA CLINICA DURANTE EL TIEMPO QUE ESTA EN EL CENTRO. DE LAS MASCOTAS QUE SE VA EN ADOPCION SE REALIZA SEGUIMIENTO A LOS HOGARES QUE FUERON ADOPTADAS.

--	--

3	¿Cómo se gestiona actualmente el proceso de adopción? ¿Qué pasos sigue un ciudadano para adoptar una mascota y cómo se registra o aprueba la solicitud?
	REALIZAR LA SOLICITUD APROBAR LA SOLICITUD COPIA DE LA CEDULA PAPEL DE VOTACION CERVICIO BASICO

4	¿Qué información se solicita al adoptante durante el proceso de adopción (datos personales, domicilio, condiciones del hogar, etc.)?
	LA COPIA DE LA CEDULA Y EL SERVICIO BASICO

5	¿Existe algún tipo de seguimiento post-adopción? De ser así, ¿cómo se lleva a cabo actualmente?
	SE REALIZA SEGUIENTOS PARA VERIFICAR EL ESTADO DE LAS MASCOTAS QUE SE DAN EN ADOPCION,

6	¿Qué dificultades o limitaciones identifica en los procesos actuales de registro, adopción y administración de la información del centro?
	LA FALTA DE COMPROMISO DE LA CIUDADANIA PARA CON LOS ANIMALES DE COMPAÑÍA, NO TIENE ESE BUEN GESTO CON LOS ANIMALES

7	¿Qué información del refugio considera importante publicar en la plataforma web para informar a los ciudadanos (historia del centro, particularidades de las mascotas, testimonios, etc.)?
---	--

}

	PARA INFORMAR A LA CUIDADANIA LA ORDENANZA Y LAS SANCIONES EXISTENTES Y VIGENTES POR MALTRATO ANIMAL
--	--

8	¿Qué tipo de reportes o estadísticas le serían útiles para la toma de decisiones en la gestión del CAEFU (número de mascotas registradas, adopciones por mes, solicitudes pendientes, usuarios registrados, etc.)?
	LOS REGISTRO DE LOS ADOPTANTES Y REALIZAR UN CENSO DE POBLACION DE ANIMALES DE COMPAÑIA
9	¿Qué requisitos o condiciones deben cumplir los usuarios que deseen adoptar una mascota a través del sistema web?
	TENER ESPACIO LO MAS IMPORTANTE Y PÓSIBILIDAD ECONOMICA

10	¿Existe personal con conocimientos en informática en el centro? ¿Con qué dispositivos y conexión a internet cuenta el CAEFU para el acceso al sistema?
	SI EL PERSONALÑ ADMINISTRATIVO

V. OBSERVACIONES ADICIONALES

--

Firma del entrevistado:	Firma del entrevistador:
	

Anexo 5. Instrumentos de recopilación de datos - Guía de observación

UNIVERSIDAD ESTATAL DE BOLÍVAR FACULTAD DE CIENCIAS ADMINISTRATIVAS, GESTIÓN EMPRESARIAL E INFORMÁTICA	GUÍA DE OBSERVACIÓN	Código: GO-CAEFU-01 Versión: 1.0
	CAEFU – Guaranda	Fecha de aplicación: 12 / 01 / 2026

GUÍA DE OBSERVACIÓN DE LOS PROCESOS DE GESTIÓN Y ADOPCIÓN DE MASCOTAS EN EL CAEFU

I. DATOS GENERALES

Institución:	Centro de Atención Emergente de Fauna Urbana (CAEFU) – Guaranda	Fecha:	12/01/2026
Nombre del observador:	CARLOS DANILO VALVERDE	Hora inicio:	9:00 AM
Lugar de observación:	Ofina del Encargado	Hora fin:	9:30 AM
Responsable presente:	Dr. Carlos Danilo Valverde		

II. OBJETIVO

Identificar y documentar los procedimientos actuales de registro de mascotas, control de adopciones, manejo de solicitudes y administración de la información en el CAEFU, con el propósito de levantar los requerimientos funcionales para el diseño del sistema web.

III. INSTRUCCIONES

El observador deberá marcar con una (✓) la casilla correspondiente según lo observado durante la visita al CAEFU. En la columna 'Observaciones' se registrarán detalles adicionales, particularidades o inconsistencias identificadas en cada aspecto.

IV. ASPECTOS A OBSERVAR

BLOQUE 1: Registro de mascotas				
I.1	Aspecto observado	Sí	No	Observaciones
	Se registra información básica de cada mascota (nombre, especie, raza, edad, género, tamaño).	☒	☐	
	Se registra el historial médico de la mascota (vacunas, desparasitaciones, cirugías realizadas).	☐	☐	
	Se utilizan fichas físicas o formularios impresos para el registro de mascotas.	☐	☐	
	Existe un archivo físico o digital donde se almacenan los registros de las mascotas.	☐	☒	
	El registro incluye fotografías de las mascotas.	☒	☐	
	Se diferencia entre mascotas disponibles para adopción y las ya adoptadas en el registro.	☒	☐	

BLOQUE 2: Proceso de adopción				
2.1	Aspecto observado	Sí	No	Observaciones
	Existe un formulario físico que el adoptante debe completar para solicitar la adopción.	☒	☐	
	Se verifican los datos del adoptante antes de aprobar la solicitud (nombre, cédula, domicilio, teléfono).	☒	☐	
	Las solicitudes de adopción son revisadas y aprobadas o rechazadas por el administrador.	☒	☐	
	Se notifica al adoptante sobre el resultado de su solicitud.	☒	☐	
	Se realiza algún tipo de seguimiento posterior a la entrega de la mascota (post-adopción).	☒	☐	
	Se registra la fecha y el responsable de cada adopción aprobada.	☒	☐	

BLOQUE 3: Administración de la información				
3.1	Aspecto observado	Sí	No	Observaciones
	Existe un listado actualizado de mascotas disponibles para adopción.	☒	☐	
	La información de las mascotas es publicada por algún medio (redes sociales, cartelera, etc.).	☐	☒	
	Se lleva un registro de los usuarios o ciudadanos interesados en adoptar.	☐	☒	
	Se generan reportes o resúmenes periódicos sobre el estado de las mascotas y las adopciones.	☒	☐	
	Existe un responsable designado para la actualización y mantenimiento de la información del refugio.	☒	☐	
	Se cuenta con equipos informáticos y conexión a internet en las instalaciones del CAEFU.	☒	☐	

BLOQUE 4: Recursos tecnológicos y operativos				
4.1	Aspecto observado	Sí	No	Observaciones
	El centro dispone de computadoras o dispositivos para acceder a un sistema web.	☒	☐	
	Existe conexión estable a internet en las instalaciones.	☒	☐	

	El personal tiene experiencia en el uso de sistemas informáticos o plataformas digitales.	☒	☐	
	Se identifican cuellos de botella o demoras frecuentes en los procesos actuales.	☐	☒	
	Los registros físicos presentan problemas de pérdida, deterioro o difícil acceso a la información.	☐	☒	

V. OBSERVACIONES GENERALES

VI. FIRMAS

Responsable del CAEFU presente:	Observador (investigador):
Nombre: <u>David Cabredo</u> Cargo: <u>Técnico de Fomento Urbano</u>	Nombre: <u>Gustavo Moya</u> C.I.: <u>0982161063</u>




Anexo 6. Manual de usuario



Manual de Usuario

Sistema Web de Adopción de Mascotas CAEFU

Guía para ciudadanos: catálogo, registro, solicitud de adopción y seguimiento

Autor:

Cristofer Fabian Muyulema Cuvi

Guaranda, Ecuador

Versión: 1.0 - /2026

Índice

Índice	2
Índice de tablas	3
Índice de figuras	3
Objeto del documento	5
Elaborado	5
Objetivo	5
Requerimientos	5
1. Introducción al sistema	5
2. Pantalla de inicio y navegación	6
3. Registro e inicio de sesión	8
3.1. Crear una nueva cuenta	8
3.2. Iniciar sesión	9
4. Catálogo de mascotas	10
5. Proceso de Adopción	11
6. Seguimiento de Solicitudes	17
6.1. Estados de una solicitud	18
6.2. Buscar y filtrar solicitudes	18
7. Sección de Testimonios	19
8. Configuración de Cuenta	20
8.1. Perfil	20
8.2. Seguridad – Cambiar contraseña	21
9. Configuración de Cuenta	22

10.	Recomendaciones y Buenas Prácticas	23
-----	------------------------------------	----

Índice de tablas

1	Menú de navegación principal	7
2	Formulario de registro de nuevo usuario.	8
3	Inicio de sesión.	10
4	Catálogo de mascotas	11
5	Formulario de entorno y cuidados (Paso 1)	12
6	Estados de una solicitud	18
7	Configuración del perfil	20
8	Cambio de contraseña	22
9	Datos de contacto y redes sociales	22

Índice de figuras

1	Flujo general de uso del sistema CAEFU	6
2	Pantalla de inicio del sistema CAEFU	6
3	Menú de navegación principal	7
4	Botones de acción	7
5	Formulario de registro de nuevo usuario.	8
6	Pantalla de inicio de sesión	9
7	Perfil detallado de mascota	10
8	Formulario de entorno y cuidados (Paso 1)	12
9	Mapa interactivo para ubicar el domicilio	13
10	Campos de dirección completados automáticamente	14
11	Acta de adopción generada automáticamente (parte 1)	14

12	Compromisos del adoptante en el acta (parte 2)	15
13	Programación de visita domiciliaria y documentos requeridos	16
14	Confirmación de solicitud enviada exitosamente	16
15	Pantalla de seguimiento de solicitudes.	17
16	Estados de una solicitud	18
17	Buscar y filtrar solicitudes	18
18	Sección de testimonios y mascota recomendada	19
19	Pantalla de configuración de perfil	20
20	Pantalla de cambio de contraseña	21
21	Pie de página con información de contacto	22

Objeto del documento

Este manual describe el funcionamiento del sistema web CAEFU desde el punto de vista del ciudadano usuario, en el contexto del proceso de adopción responsable de mascotas rescatadas en el cantón Guaranda, Ecuador. Está pensado como guía de referencia para quien quiera explorar el catálogo de mascotas disponibles, registrarse en la plataforma y completar una solicitud de adopción.

Elaborado

Participante: Cristofer Fabian Muyulema Cuvi

Área/Departamento Carrera de Software-UEB

Teléfono 0985577630

Rol: Desarrollador

Objetivo

Guiar a la ciudadanía en el uso del sistema web CAEFU, de modo que pueda explorar el catálogo de mascotas, registrarse, iniciar sesión, completar el proceso de solicitud de adopción y dar seguimiento a su estado sin necesidad de asistencia adicional.

Requerimientos

- Conexión a internet y un navegador actualizado (Chrome, Firefox, Edge o Safari)
- Número de cédula de ciudadanía
- Correo electrónico activo
- Fotografías de la cédula de identidad, tanto del frente y reverso

1. Introducción al sistema

El sistema web de CAEFU busca facilitar el proceso de adopción responsable de mascotas rescatadas, conectando a los ciudadanos con animales que necesitan un hogar.

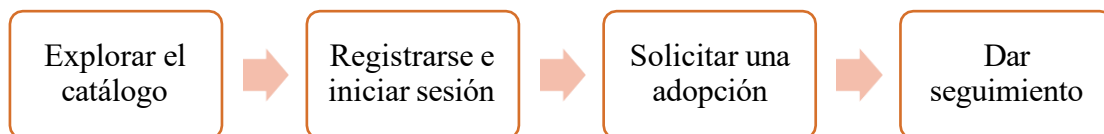
¿Qué puedes hacer en CAEFU?

- Explorar el catálogo de mascotas disponibles para adopción.
- Consultar el perfil de cada animal (raza, edad, tamaño, historial médico).

- Registrarte y crear una cuenta personal.
- Enviar solicitudes de adopción con información sobre tu hogar.
- Firmar el acta de adopción y adjuntar los documentos requeridos.
- Dar seguimiento al estado de tus solicitudes.
- Publicar y leer testimonios de otros adoptantes..

Figura 1.

Flujo general de uso del sistema CAEFU



2. Pantalla de inicio y navegación

Al ingresar al sitio web de CAEFU verás la página de inicio con el mensaje principal: "Cada huella tiene una historia", desde aquí accedes a todas las secciones del sistema.

Figura 2.

Pantalla de inicio del sistema CAEFU



Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com/>

Tabla 1.

Menú de navegación principal

Elemento	Descripción
Inicio	Regresa a la pantalla principal.
Catálogo	Mascotas disponibles para adopción.
Testimonio	Historias de éxito de adoptantes.
Adopta	Acceso directo al proceso de adopción.
Contacto	Formulario para comunicarse con CAEFU.
Buscar amigos...	Búsqueda rápida de mascotas por nombre.
Iniciar Sesión	Acceso para usuarios registrados.
Registrar	Crear una nueva cuenta de usuario.

Figura 3.

Menú de navegación principal



Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com/>

Botones de Acción

- Comenzar búsqueda: Te lleva al catálogo de mascotas.
- Conocer historias: Redirige a la sección de testimonios.

Figura 4.

Botones de acción



Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com/>

3. Registro e inicio de sesión

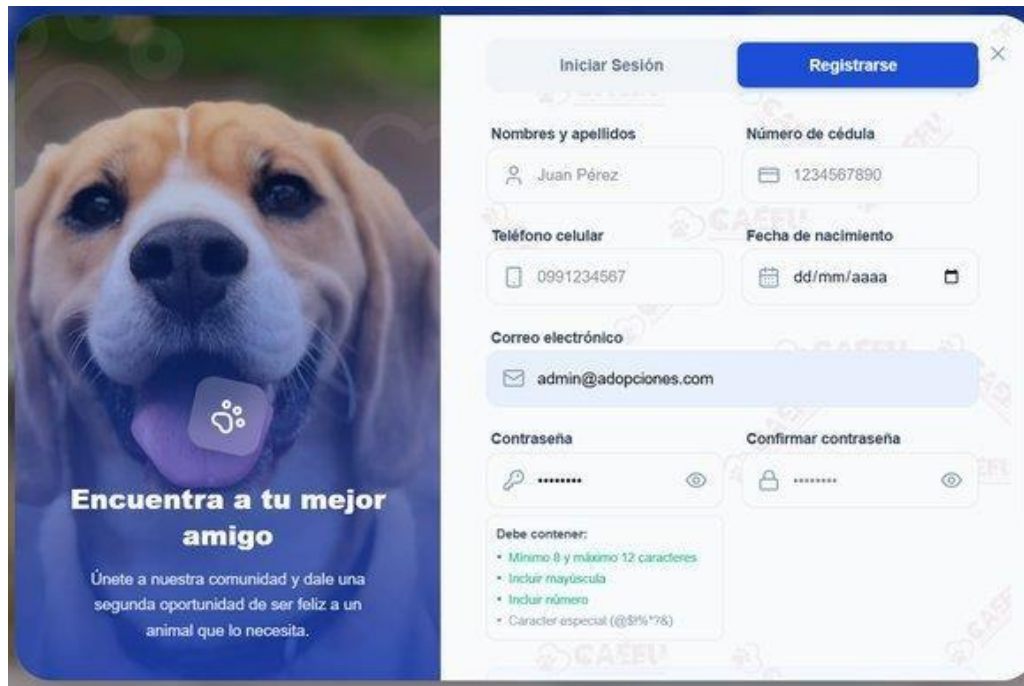
Para acceder a las funcionalidades completas del sistema es necesario crear una cuenta o iniciar sesión.

3.1. Crear una nueva cuenta

Haz clic en el botón "Registrar" en la esquina superior derecha. Se abrirá el siguiente formulario:

Figura 5.

Formulario de registro de nuevo usuario.

The image shows a registration form for a website. On the left, there is a vertical banner featuring a close-up of a beagle's face. The banner has the text "Encuentra a tu mejor amigo" in white, followed by "Únete a nuestra comunidad y dale una segunda oportunidad de ser feliz a un animal que lo necesita." in smaller white text. On the right, the registration form is displayed. It has a header with two buttons: "Iniciar Sesión" and "Registrarse". Below the buttons, there are several input fields: "Nombres y apellidos" (with a person icon and the text "Juan Pérez"), "Número de cédula" (with a document icon and the text "1234567890"), "Teléfono celular" (with a phone icon and the text "0991234567"), "Fecha de nacimiento" (with a calendar icon and the text "dd/mm/aaaa"), "Correo electrónico" (with an envelope icon and the text "admin@adopciones.com"), "Contraseña" (with a key icon and masked text "*****"), and "Confirmar contraseña" (with a padlock icon and masked text "*****"). At the bottom of the form, there is a section titled "Debe contener:" followed by a list of requirements: "Mínimo 8 y máximo 12 caracteres", "Incluir mayúscula", "Incluir número", and "Carácter especial (@\$!%*78)". The form has a light blue background and a close button (X) in the top right corner.

Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com/>

Tabla 2.

Formulario de registro de nuevo usuario.

Acción	Descripción
Nombres y apellidos	Tu nombre completo tal como aparece en tu cédula.
Número de cédula	Cédula de ciudadanía (10 dígitos).
Teléfono celular	Número activo con formato 09XXXXXXXX.

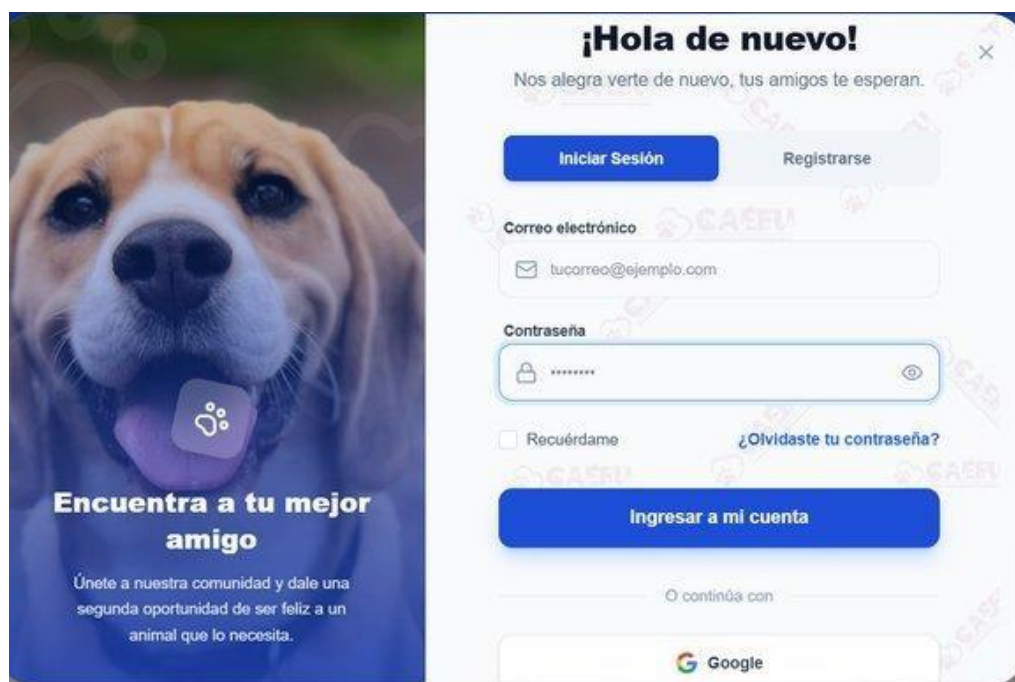
Fecha de nacimiento	Selecciona en formato dd/mm/aaaa.
Correo electrónico	Correo válido que usarás para iniciar sesión.
Contraseña	Crea una contraseña segura (ver requisitos).
Confirmar contraseña	Repite la contraseña para verificar.
IMPORTANTE:	Mínimo 8 y máximo 12 caracteres.
Requisitos de contraseña	Debe incluir al menos una letra mayúscula.
	Debe incluir al menos un número.
	Carácter especial (\$;%*?)

3.2. Iniciar sesión

Si ya tienes cuenta haz clic en "Iniciar Sesión", el sistema mostrará el mensaje ¡Hola de nuevo!

Figura 6.

Pantalla de inicio de sesión



Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com/>

Tabla 3.

Inicio de sesión.

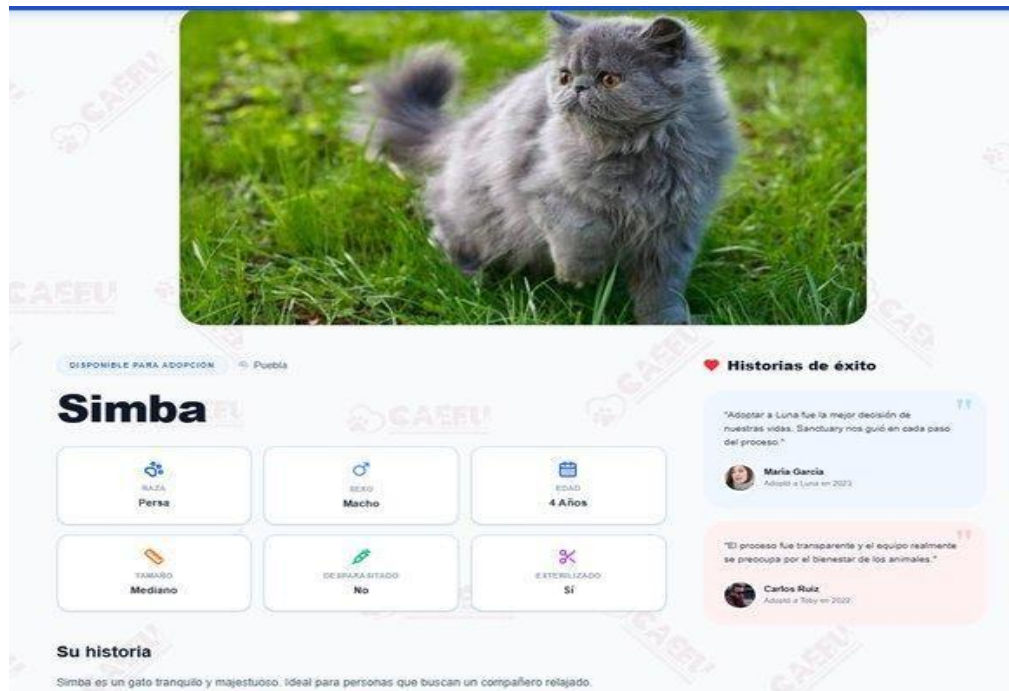
Acción	Descripción
Correo electrónico	El correo con el que te registraste.
Contraseña	Tu contraseña. Usa el ícono del ojo para mostrarla.
Recuérdame	Activa si deseas mantener la sesión iniciada.
Ingresar	Haz clic en "Ingresar a mi cuenta".
Opciones adicionales	¿Olvidaste tu contraseña?: recupera el acceso a tu cuenta. Continúa con Google: inicia sesión con tu cuenta de Google sin contraseña adicional.

4. Catálogo de mascotas

En el catálogo encontrarás todas las mascotas disponibles para adopción. Haz clic sobre cualquiera para ver su perfil.

Figura 7.

Perfil detallado de mascota



Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com/>

Tabla 4.

Catálogo de mascotas

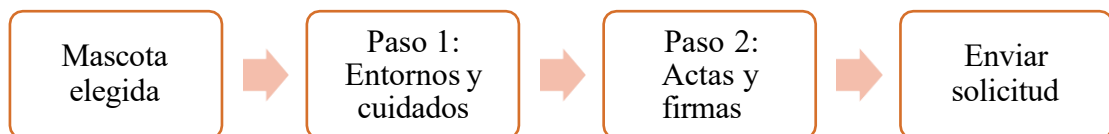
Elemento	Descripción
Fotografía	Imagen principal de la mascota.
Estado	Indica si está "Disponible para adopción".
Nombre	Nombre asignado en el centro.
Raza	Raza o tipo de animal.
Sexo	Macho o hembra.
Edad	Edad aproximada del animal.
Tamaño	Pequeño, mediano o grande.
Desparasitado	Si ha recibido tratamiento antiparasitario.
Esterilizado	Si la mascota ha sido esterilizada.
Su historia	Carácter y necesidades de la mascota.
Historias de éxito	Testimonios de otros adoptantes.

5. Proceso de Adopción

Para iniciar una solicitud debes estar registrado e iniciar sesión, el proceso tiene dos pasos: Entorno y Cuidados, y Acta y Firma.

Figura 1.

Proceso de solicitud de adopción



Pasos:

1. Entorno y Cuidados de la Mascota: Este paso informa sobre donde vivirá y en qué condiciones de cuidado estará la mascota.

Figura 8.
Formulario de entorno y cuidados (Paso 1)

Solicitud de Adopción: Simba

1 Entorno y Cuidados 2 Acta y Firma

Paso 1: Entorno y Cuidados de la Mascota
Por favor, indicanos dónde habitará la mascota y el tiempo estimado para sus cuidados cotidianos.

Foto del Espacio (Opcional)
Subir foto
Formatos: JPG, PNG, WEBP

Tipo de Vivienda *
Casa Propia

Dimensiones del Espacio (Largo x Ancho en metros) *
30 m x 12 m

Horas diarias de dedicación *
3 horas / día

Tipo de Alimentación *
Croquetas

DIRECCIÓN DEL DOMICILIO
Arrastra el mapa para ubicar el puntero en tu domicilio
Centrar en mi ubicación

Cancelar Continuar

Elaborado por: Cristófer M.
Fuente: <https://caefuguaranda.com/>

Tabla 5.
Formulario de entorno y cuidados (Paso 1)

Acción	Descripción
Foto del Espacio	Opcional. Sube una foto del espacio. Formatos: JPG, PNG, WEBP.
Tipo de Vivienda	Selecciona: Casa Propia, Arrendada, etc.
Dimensiones del Espacio	Largo y ancho en metros del espacio disponible.
Horas de dedicación	Horas diarias que podrás dedicar a la mascota.
Tipo de Alimentación	Croquetas, Balanceado, u otras opciones disponibles.
Mapa de ubicación	Arrastra el puntero hasta tu domicilio exacto.
Calle Principal	Nombre de la calle principal de tu domicilio.
Calle Secundaria	Opcional. Calle de intersección.

Figura 9.

Mapa interactivo para ubicar el domicilio



Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com/>

Uso del mapa de ubicación

- Arrastra el mapa para mover el puntero rojo hasta tu domicilio exacto.
- Usa los botones + y – para hacer zoom.
- Haz clic en "Centrar en mi ubicación" para detección GPS automática.
- Los campos de parroquia, cantón y provincia se completan automáticamente.

Figura 10.

Campos de dirección completados automáticamente



Formulario de dirección con los siguientes campos prellenados:

- Calle Principal *: Coronel García
- Calle Secundaria (Opcional): Calle secundaria
- Parroquia *: Ventimilla
- Cantón *: Guaranda
- Provincia *: Bolívar

Botones: Cancelar, Continuar

Elaborado por: Cristófer M.

Fuente: <https://caefugaranda.com/>

2. Acta y Firma: El sistema genera el Acta de Adopción oficial con todos tus datos, revisala antes de proceder.

Figura 11.

Acta de adopción generada automáticamente (parte 1)



Solicitud de Adopción: Simba

Progreso: 1. Entorno y Cuidados (completado) → 2. Acta y Firma (actual)

SOLICITUD DE ADOPCIÓN

Elaborado por: Dr. Danilo Valverde
ADMINISTRADOR CAEFU

Autorizado por: Ing. Geovany Ramirez
DIRECTOR GESTIÓN AMBIENTAL -
GADCG

A los 27 del mes de junio año 2026, comparecen a la suscripción de la presente acta, por una parte, el Centro de Atención Emergente de Fauna Urbana de Guaranda, debidamente representado por el señor **Dr. Danilo Valverde**, Servidor Público del CAEFU; y, por otra parte, el/la señor/a: **yuri muyulema**, en calidad de adoptante de la mascota, motivo de la presente acta, portador de la cédula de ciudadanía N° **0202159190**.

El GADCG a través del Centro de Atención Emergente de Fauna Urbana de Guaranda, declara haber recogido en estado de abandono un canino/felino con las siguientes características:

ID	Raza	Sexo	Edad	Tamaño
5	PERSA	MACHO	4 AÑOS	MEDIANO

Mascota que ha recibido los siguientes tratamientos profilácticos y/o quirúrgicos:

Elaborado por: Cristófer M.

Fuente: <https://caefugaranda.com/>

Figura 12.

Compromisos del adoptante en el acta (parte 2)

mascota que ha recibido los siguientes tratamientos preventivos y/o terapéuticos:

FECHA	VACUNA	DESPARASITACIÓN	FECHA VACUNACIÓN SUGERIDA	FECHA ESTERILIZACIÓN
Pendiente				

Por su parte el/la señor/a: **yuri muyulema** con cédula de ciudadanía No. **0202159190**, domiciliado en las calles **Coronel García** de la parroquia **Ventimilla** del cantón **Guaranda** de la provincia **Bolívar**, con Teléfono No. **0994138861**, declara tener condiciones suficientes para adoptar a la mascota en cuestión, por lo que el Gobierno Autónomo Descentralizado del cantón Guaranda a través del Centro de Atención Emergente de Fauna Urbana, le entrega en **ADOPCIÓN**, y se compromete a:

- Alimentarlo adecuadamente (alimentación definida: **Balanceado / Croquetas**).
- Ubicarlo en su domicilio (vivienda: **Casa Propia**) en un espacio adecuado de **30m x 12m**.
- Dedicarle tiempo diario para ejercicio y recreación (dedicación: **3 horas al día**).
- Darle atención veterinaria, mantener acercamiento humano, y evitar que termine nuevamente en las calles.

El Adoptante se compromete además a permitir la visita de un representante del Gobierno Autónomo Descentralizado del cantón Guaranda, a fin de verificar el buen estado de la mascota y el cumplimiento del presente convenio; así como de la ordenanza municipal que regula el cuidado de mascotas.

Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com/>

- Información del adoptante
- Datos de la mascota
- Historial médico de la mascota
- Compromisos aceptados correspondientes a alimentación, espacio, tiempo de dedicación y atención veterinaria
- Cláusula de visita domiciliaria por parte del GAD Guaranda

Programación de Visita Domiciliaria Obligatoria

- Se selecciona una fecha de visita entre mañana y los próximos 60 días.
- Se selecciona una hora disponible para recibir la visita.

Documentos Requeridos

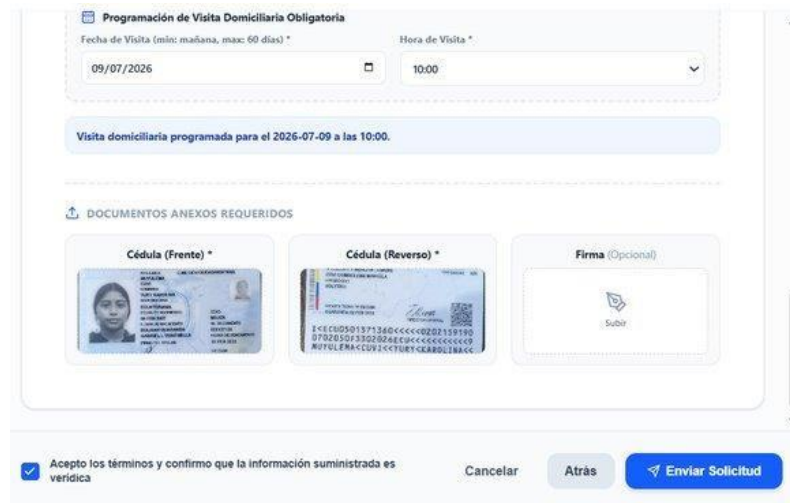
- Cédula (de frente y reverso): Subir una foto nítida de la cédula. **OBLIGATORIO**.
- Firma: Subir una imagen de tu firma manuscrita. **OPCIONAL**.

Enviar la Solicitud

- Deberá leer el acta de adopción generada por el sistema y marcar la casilla "Acepto los términos y confirmo que la información suministrada es verídica".
- Hacer clic en el botón "Enviar Solicitud".

Figura 13.

Programación de visita domiciliaria y documentos requeridos



Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com/>

Figura 14.

Confirmación de solicitud enviada exitosamente



Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com/>

Confirmación de envío: Una vez enviada la solicitud el sistema mostrará: ¡Solicitud Enviada! – Proceso completado exitosamente. Recibirás notificaciones a través de:

- La sección Solicitudes en tu perfil.
- La sección de Notificaciones del sistema.
- Un mensaje de WhatsApp cuando tu solicitud sea aceptada.

6. Seguimiento de Solicitudes

Desde la sección "Solicitudes" del menú principal puedes monitorear el estado de todas tus solicitudes de adopción.

Figura 15.

Pantalla de seguimiento de solicitudes.



Elaborado por: Cristófer M.

Fuente: <https://caefugaranda.com/>

6.1. Estados de una solicitud

Figura 16.

Estados de una solicitud



Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com/>

Tabla 6.

Estados de una solicitud

Estado	Descripción
Pendiente	La solicitud fue enviada y está en revisión por el equipo CAEFU.
En Visita	Se ha programado la visita domiciliaria, pendiente de realizarse.
Aprobada	Solicitud aprobada. Puedes proceder a recoger a tu mascota.
Rechazada	Solicitud no aprobada. Revisa las observaciones del evaluador.

6.2. Buscar y filtrar solicitudes

- Busca por nombre de mascota o ID en el campo de búsqueda.
- Filtrar por rango de fechas con los selectores de inicio y fin.
- Haz clic sobre una solicitud para expandir y ver detalles.

Figura 17.

Buscar y filtrar solicitudes



Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com/>

7. Sección de Testimonios

La sección de Testimonios te permite leer experiencias de otros adoptantes y también compartir la tuya si ya adoptaste una mascota a través de CAEFU.

Figura 18.

Sección de testimonios y mascota recomendada



Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com/>

Qué encontrarás en esta sección?

- Historias reales de personas que adoptaron mascotas a través de CAEFU.
- Un carrusel de mascotas recomendadas con opción directa de adopción.
- El formulario "Comparte tu historia" para publicar tu propio testimonio.

Cómo Publicar tu Testimonio

- Inicia sesión en tu cuenta.
- Dirígete a la sección Testimonio desde el menú.
- Escribe tu experiencia en el campo "Escribe tu testimonio aquí...".
- Haz clic en el botón Publicar Testimonio.

Mascotas Recomendadas

- En el costado derecho verás mascotas destacadas con el indicador "Recomendado para ti". Puedes hacer clic en Ver Perfil para conocerlas mejor o en Adoptar para iniciar el proceso directamente

8. Configuración de Cuenta

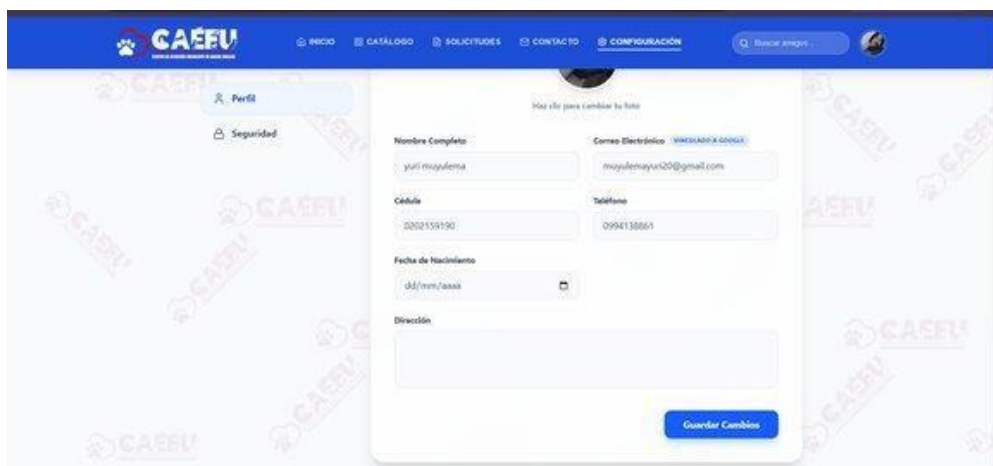
Desde el menú Configuración puedes gestionar tu información personal y las opciones de seguridad de tu cuenta. Esta sección tiene dos pestañas: Perfil y Seguridad.

8.1. Perfil

Aquí puedes actualizar tus datos personales registrados en el sistema.

Figura 19.

Pantalla de configuración de perfil



Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com/>

Tabla 7.

Configuración del perfil

Elemento	Descripción
Foto de perfil	Haz clic en la imagen para cambiar tu foto de perfil.
Nombre Completo	Tu nombre completo registrado en el sistema.
Correo Electrónico	El correo de tu cuenta (puede estar vinculado a Google).

Cédula	Número de cédula de ciudadanía.
Teléfono	Número de celular de contacto.
Fecha de Nacimiento	Tu fecha de nacimiento en formato dd/mm/aaaa.
Dirección	Tu dirección de domicilio.

Una vez realizados los cambios, haz clic en el botón Guardar Cambios para actualizarlos.

****Cuenta vinculada a Google:** Si iniciaste sesión con Google, tu correo electrónico aparecerá con la etiqueta VINCULADO A GOOGLE y no podrá modificarse directamente desde el sistema.

8.2. Seguridad – Cambiar contraseña

Desde la pestaña Seguridad puedes actualizar tu contraseña de acceso.

Figura 20.

Pantalla de cambio de contraseña

The screenshot displays the 'Configuración de Cuenta' (Account Configuration) interface. On the left, a sidebar contains 'Perfil' and 'Seguridad' (highlighted). The main content area is titled 'Cambiar Contraseña' (Change Password) with a subtitle 'Asignarle de usar una contraseña larga y compleja para mantener su cuenta segura'. It features three input fields: 'Contraseña Actual', 'Nueva Contraseña', and 'Confirmar Nueva Contraseña', each with a small circular icon for toggling password visibility. A blue button labeled 'Actualizar Contraseña' is positioned at the bottom of the form.

Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com/>

Tabla 8.

Cambio de contraseña

Acción	Descripción
Contraseña Actual	Ingresa tu contraseña actual para verificar tu identidad.
Nueva Contraseña	Escribe la nueva contraseña (respeta los requisitos de seguridad).
Confirmar Contraseña	Repite la nueva contraseña para confirmar.
Actualizar	Haz clic en "Actualizar Contraseña" para guardar los cambios.

9. Configuración de Cuenta

En la parte inferior de todas las páginas encontrarás el pie de página con información de contacto y enlaces rápidos del sistema CAEFU.

Figura 21.

Pie de página con información de contacto



Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com/>

Tabla 9.

Datos de contacto y redes sociales

Elemento	Descripción
Correo electrónico	CAEFU@gmail.com
Teléfono	0987654321
Ubicación	Guaranda, Ecuador

También se puede usar el formulario de Contacto disponible en el menú principal para enviar consultas o comentarios directamente al equipo CAEFU.

10. Recomendaciones y Buenas Prácticas

Antes de Solicitar una Adopción

Conversa con todos en casa, ya que la decisión de adoptar debe ser de todos, piensa si dispones de un espacio adecuado para el tamaño de la mascota y si puedes darle tiempo todos los días y ten a mano tu cédula antes de empezar el trámite.

Durante el Llenado del Formulario

Los datos ingresados deben coincidir con la de cédula, las fotos subidas deben ser nítidas y legibles, deberás ubicar tu domicilio con precisión para la visita domiciliar y lee el acta de adopción con calma.

Después de Enviar la Solicitud

Mantén el WhatsApp activo ya que ahí llegan las notificaciones, para ver el estado del trámite entra a la sección "Solicitudes" y si tienes duda escríbenos por la sección "Contacto".

Anexo 7. Manual de administrador



Manual de Administrador del Sistema Web de Adopción de Mascotas CAEFU

Panel de Control del Administrador: Gestión de mascotas, solicitudes de adopción, usuarios y notificaciones

Autor:

Cristofer Fabian Muyulema Cuvi

Guaranda, Ecuador

Versión: 1.0 - /2026

Índice

Índice	2
Índice de figuras	3
Objeto del documento	5
Elaborado	5
Objetivo	5
Requerimientos	5
1. Introducción al sistema	5
1.1. Acceso al panel	6
1.2. Estructura del menú lateral	6
2. Dashboard – Panel de control	7
2.1. Gráfico de adopciones	7
2.2. Secciones adicionales del Dashboard	8
3. Gestión de mascotas	9
3.1. Filtros y búsqueda	10
3.2. Registrar una nueva mascota	11
4. Gestión de solicitudes de adopción	13
4.1. Tabla de solicitudes	14
4.2. Proceso de revisión en tres etapas	14
5. Gestión de usuarios	16
5.1. Resumen de usuarios	17
5.2. Tabla de usuarios	17
5.3. Crear un nuevo usuario	18

5.4.	Cambiar el rol de un usuario	18
6.	Notificaciones inteligentes	19
6.1.	Tipo de notificaciones	20
6.2.	Pestañas de filtro	20
7.	Configuración de la cuenta Admin	21
7.1.	Datos personales	21
7.2.	Seguridad – Cambiar contraseña	22
8.	Recomendaciones operativas	23

Índice de tablas

1	Estructura del menú lateral del administrador	6
2	Tarjetas de resumen del Dashboard	7
3	Tarjetas de resumen de gestión de mascotas	10
4	Formulario Paso 1: información básica de la mascota	12
5	Formulario Paso 2: historial médico de la mascota	13
6	Formulario Paso 2: historial médico de la mascota	13
7	Solicitudes	14
8	Proceso de revisión de solicitudes de adopción en tres etapas	15
9	Resumen de usuarios	17
10	Usuarios	17
11	Tipo de notificaciones	20
12	Configuración de datos personales	22
13	Cambio de contraseña	23

Índice de figuras

1	Flujo de acceso al Panel de Administrador	6
2	Dashboard con estadísticas generales de la plataforma	7
3	Gráfico de adopciones	8
4	Mascotas recientes	8
5	Actividad reciente	9
6	Acciones rápidas	9
7	Pantalla de gestión de mascotas con filtros y estadísticas.	10
8	Filtros y búsqueda	11
9	Información básica para agregar mascota	11
10	Agregar historial médico por mascota	12
11	Pantalla de gestión de solicitudes de adopción.	14
12	Indicador visual de los 3 pasos del proceso de revisión	15
13	Pantalla de gestión de usuarios registrados.	16
14	Resumen de usuarios	17
15	Usuarios	18
16	Crear nuevo usuario	18
17	Cambiar el rol de un usuario	19
18	Centro de notificaciones inteligentes del sistema.	19
19	Tipo de notificaciones.	20
20	Pestañas filtro de notificaciones.	21
21	Configuración de datos personales del administrador	21
22	Pantalla de cambio de contraseña del administrador	22

Objeto del documento

Este documento describe el funcionamiento del Panel de Administrador del sistema web CAEFU, desarrollado para gestionar el proceso de adopción en el cantón Guaranda, Ecuador. Sirve como guía para el personal administrador y se establece los procedimientos, pantallas, campos y flujos de trabajo necesarios para operar la plataforma.

Elaborado

Participante: Cristofer Fabian Muyulema Cuvi

Área/Departamento Carrera de Software-UEB

Teléfono 0985577630

Rol: Desarrollador

Objetivo

Guiar el uso del Panel de Control del sistema web de adopción de mascotas, de modo que la persona administradora pueda gestionar los módulos de mascotas, solicitudes de adopción, usuarios, notificaciones y configuración de cuenta.

Requerimientos

- Para acceder al panel contar con credenciales de administrador (usuario y contraseña)
- Tener habilitado el rol ADMIN dentro del sistema
- Dispositivo de escritorio, laptop o tablet con conexión estable a internet
- Navegador web actualizado (Google Chrome, Microsoft Edge o Firefox) en su versión más reciente

1. Introducción al sistema

El sistema es una plataforma web pensada para modernizar y centralizar el proceso de adopción de mascotas del Centro de Atención Emergente de Fauna Urbana de

Guaranda. El Panel de Administrador es el módulo de gestión interna que controla el ciclo de vida de una mascota dentro del programa.

1.1. Acceso al panel

El administrador ingresa al sistema con las credenciales de administrador, el sistema detecta el rol ADMIN y redirige al usuario hacia el Panel de Control, donde el menú lateral izquierdo muestra todas las secciones disponibles para su gestión.

Figura 2.

Flujo de acceso al Panel de Administrador



Importante

- El panel de administrador NO es accesible para usuarios regulares.
- Mantén tus credenciales en un lugar seguro.
- Cierra sesión siempre al terminar tu jornada de trabajo.

1.2. Estructura del menú lateral

El menú lateral organiza el panel en seis secciones principales, cada una asociada a un módulo funcional del sistema, tal como se presenta en la Tabla 1.

Tabla 1.

Estructura del menú lateral del administrador

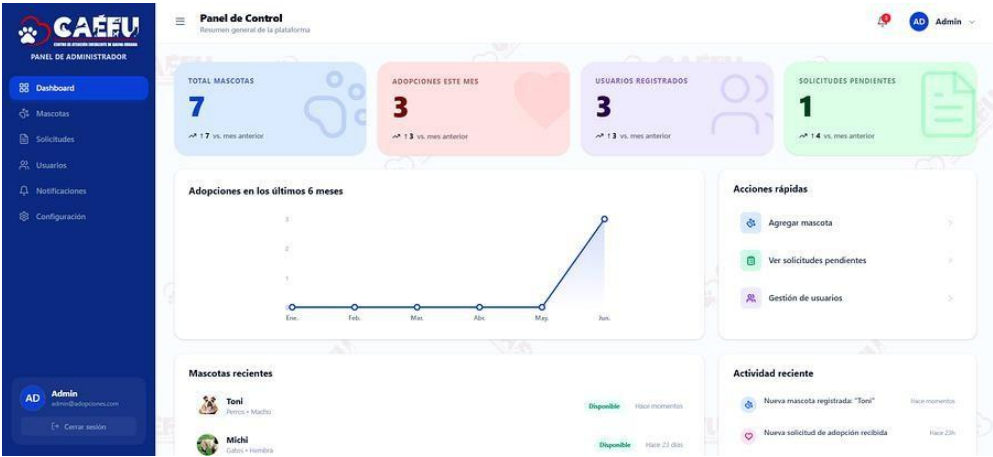
Sección	Descripción
Dashboard	Resumen general con estadísticas e indicadores clave.
Mascotas	Gestión completa del registro de mascotas del centro.
Solicitudes	Revisión y gestión de solicitudes de adopción.
Usuarios	Administración de cuentas de usuarios registrados.
Notificaciones	Centro de alertas inteligentes del sistema.
Configuración	Datos personales y seguridad de la cuenta admin.

2. Dashboard – Panel de control

El Dashboard es la pantalla principal al ingresar al panel, ofrece un resumen visual del estado actual de la plataforma, permite al administrador identificar el volumen de mascotas, adopciones y solicitudes pendientes.

Figura 3.

Dashboard con estadísticas generales de la plataforma



Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com/>

Tabla 2.

Tarjetas de resumen del Dashboard

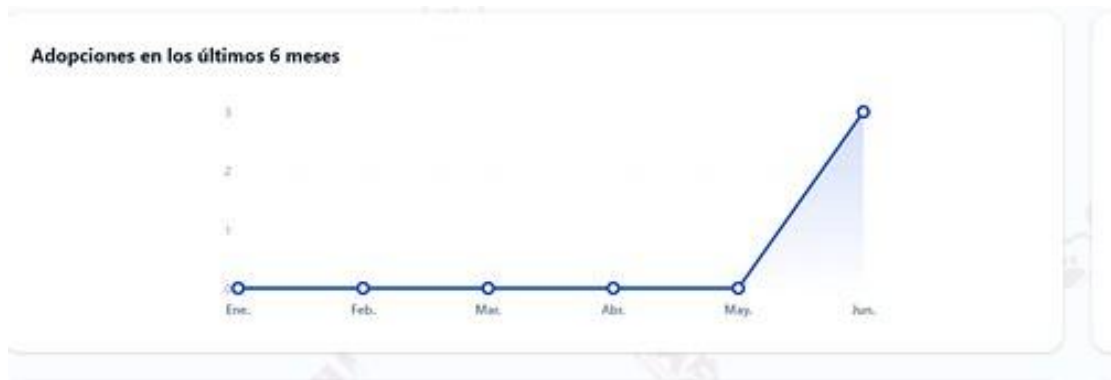
Indicador	Significado
Total Mascotas	Número total de mascotas registradas en el sistema.
Adopciones este mes	Cantidad de adopciones completadas en el mes actual.
Usuarios Registrados	Total de usuarios con cuenta activa en la plataforma.
Solicitudes Pendientes	Solicitudes que requieren revisión inmediata del administrador.

2.1. Gráfico de adopciones

El gráfico de línea muestra la evolución de adopciones en los últimos seis meses, permitiendo identificar tendencias y picos de actividad.

Figura 4.

Gráfico de adopciones



Elaborado por: Cristofer M.

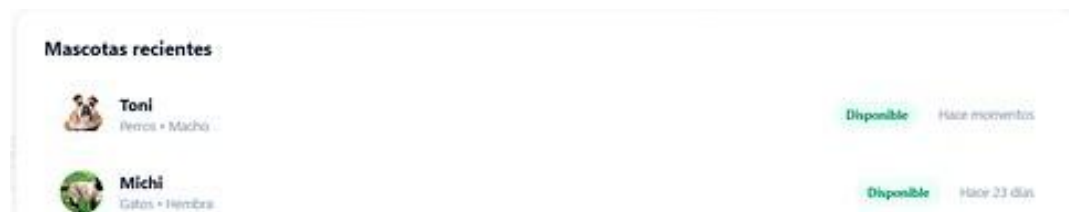
Fuente: <https://caefugaranda.com/>

2.2. Secciones adicionales del Dashboard

- Mascotas recientes: últimas mascotas registradas con su estado (Disponible / Adoptada).

Figura 5.

Mascotas recientes



Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com/>

- Actividad reciente: registro de las últimas acciones del sistema (nuevas mascotas, solicitudes recibidas).

Figura 6.

Actividad reciente



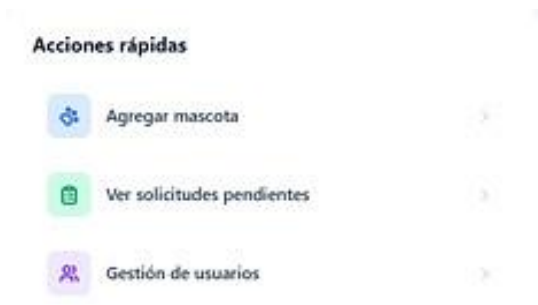
Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com/>

- Acciones rápidas: atajos para Agregar mascota, Ver solicitudes pendientes y Gestión de usuarios.

Figura 7.

Acciones rápidas



Elaborado por: Cristofer M.

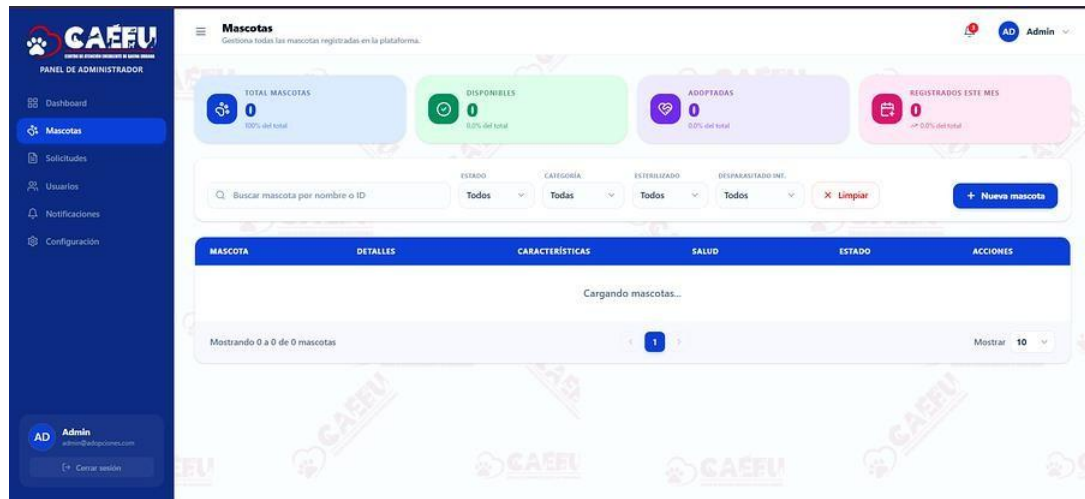
Fuente: <https://caefugaranda.com/>

3. Gestión de mascotas

Desde la sección Mascotas el administrador puede registrar, consultar, editar y eliminar los animales del centro.

Figura 8.

Pantalla de gestión de mascotas con filtros y estadísticas.



Elaborado por: Cristófer M.

Fuente: <https://caefugaranda.com/>

Tabla 3.

Tarjetas de resumen de gestión de mascotas

Indicador	Significado
Total Mascotas	Cantidad total de animales registrados.
Disponibles	Mascotas listas para ser adoptadas.
Adoptadas	Mascotas que ya tienen un hogar asignado.
Registradas este mes	Nuevos ingresos al centro en el mes actual.

3.1. Filtros y búsqueda

- Buscar por nombre o ID: campo de texto para localizar un animal.
- Estado: Todos / Disponible / Adoptado / En proceso.
- Categoría: Perros / Gatos / otras categorías.
- Esterilizado: filtrar por estado de esterilización.
- Desparasitado Int.: filtra por estado de desparasitación interna.
- Limpiar: restablece todos los filtros a sus valores predeterminados.

Figura 9.

Filtros y búsqueda



The interface shows a search bar with the placeholder text "Buscar mascota por nombre o ID". To the right of the search bar are four filter dropdown menus: "ESTADO" (set to "Todos"), "CATEGORÍA" (set to "Todas"), "ESTERILIZADO" (set to "Todos"), and "DESparasitado (SI/NO)" (set to "Todos"). There is a red "X Limpiar" button and a blue "+ Nueva mascota" button.

Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com/>

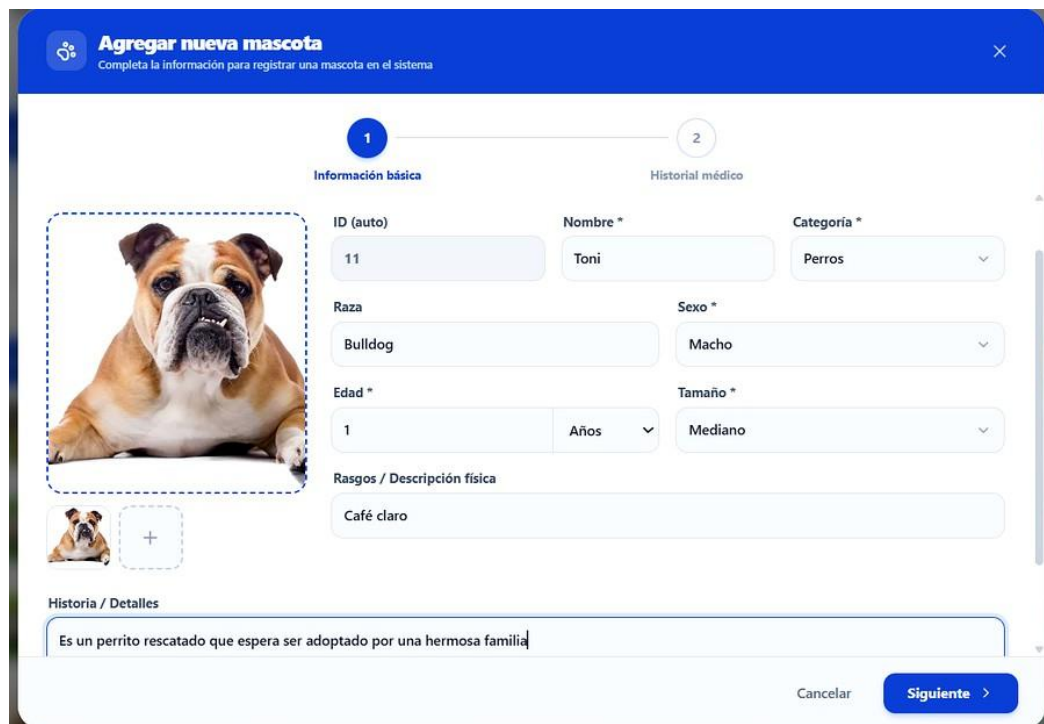
3.2. Registrar una nueva mascota

Al hacer clic en el botón + Nueva mascota, ubicado en la esquina superior derecha, se abre un formulario compuesto por dos pasos secuenciales:

Paso 1: Información básica

Figura 10.

Información básica para agregar mascota



The form is titled "Agregar nueva mascota" with a subtitle "Completa la información para registrar una mascota en el sistema". It has a progress indicator with two steps: "1" (selected) and "2". The first step is "Información básica" and the second is "Historial médico".

The form contains the following fields:

- ID (auto):** A text field with the value "11".
- Nombre *:** A text field with the value "Toni".
- Categoría *:** A dropdown menu with the value "Perros".
- Raza:** A text field with the value "Bulldog".
- Sexo *:** A dropdown menu with the value "Macho".
- Edad *:** A text field with the value "1".
- Tamaño *:** A dropdown menu with the value "Mediano".
- Rasgos / Descripción física:** A text field with the value "Café claro".

There is a photo upload section on the left with a dashed box containing a bulldog image and a "+" button. Below it is a "Historia / Detalles" section with a text area containing the text "Es un perrito rescatado que espera ser adoptado por una hermosa familia".

At the bottom right, there are two buttons: "Cancelar" and "Siguiente >".

Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com/>

Tabla 4.

Formulario Paso 1: información básica de la mascota

# Campo	Descripción
1 Foto principal	Sube la foto principal (obligatoria). Agrega fotos adicionales con el botón +.
2 ID (auto)	El sistema asigna automáticamente un ID único. No es editable.
3 Nombre	Nombre asignado a la mascota en el centro. Campo obligatorio.
4 Categoría	Selecciona: Perros, Gatos u otras categorías.
5 Raza	Escribe la raza del animal (ej.: Bulldog, Persa, Mestizo).
6 Sexo	Selecciona: Macho o Hembra.
7 Edad	Ingresa la edad y selecciona la unidad: Meses o Años.
8 Tamaño	Selecciona: Pequeño, Mediano o Grande.
9 Rasgos / Desc. física	Descripción del aspecto físico (color, marcas, etc.).
10Historia / Detalles	Texto con la historia y personalidad de la mascota.

Haz clic en Siguiente para avanzar al Paso 2.

Paso 2 – Historial Médico

Se registran los tratamientos médicos que ha recibido la mascota. Se debe hacer clic en + Agregar fila de registro para añadir cada tratamiento.

Figura 11.

Agregar historial médico por mascota

Agregar nueva mascota
Completa la información para registrar una mascota en el sistema.

1 Información básica 2 **Historial médico**

Historial médico
Agrega los registros clínicos en la tabla. Haz clic en "Agregar fila de registro" para añadir un registro.

FECHA	TIPO	DETALLE / VACUNA	PRÓX. CONTROL	NOTAS	ACCIONES
2026-06-11	Desparasitación	Interna	—	—	

+ Agregar fila de registro

< Anterior Cancelar **Guardar mascota**

Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com/>

Tabla 5.

Formulario Paso 2: historial médico de la mascota

#Campo	Descripción
1 Fecha	Fecha del tratamiento (formato AAAA-MM-DD).
2 Tipo	Vacuna, Desparasitación, Esterilización, etc.
3 Detalle / Vacuna	Nombre específico del medicamento o vacuna.
4 Próx. Control	Fecha sugerida para el siguiente control.
5 Notas	Observaciones adicionales del veterinario.
6 Acciones	Editar o eliminar cada registro.

Al finalizar, se hace clic en Guardar mascota para registrar el animal en el sistema.

Buenas prácticas al registrar mascotas

- Sube siempre una fotografía clara y de buena calidad de frente.
- Completa el historial médico lo más detallado posible; el adoptante lo verá en el acta.
- Si el animal no tiene historial médico, deja la tabla vacía y agrégala después.
- El estado inicial de toda mascota nueva es “Disponible”.

4. Gestión de solicitudes de adopción

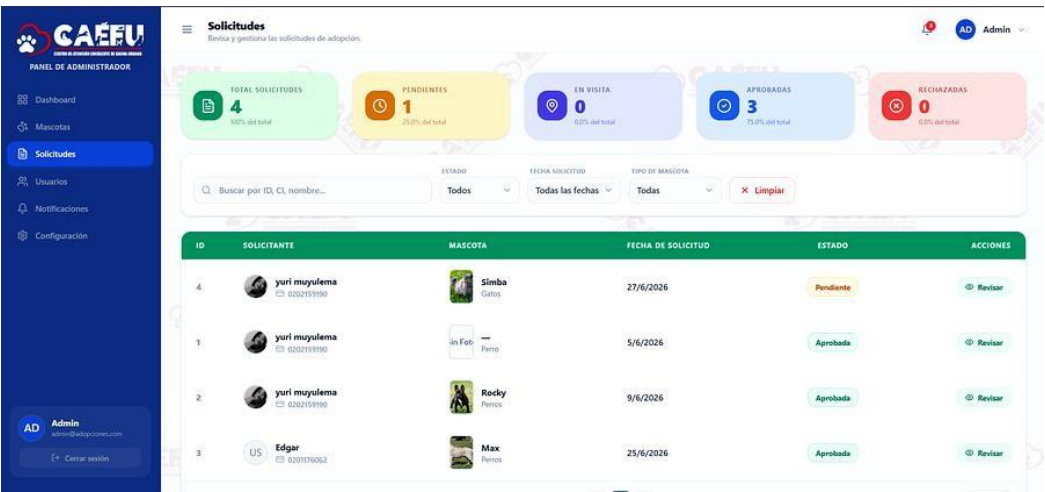
La sección Solicitudes centraliza todas las peticiones de adopción enviadas por los usuarios, el administrador debe revisar cada solicitud en tres etapas antes de emitir una decisión final.

Tabla 6.

Formulario Paso 2: historial médico de la mascota

Estado	Descripción
Total Solicitudes	Número total de solicitudes recibidas.
Pendientes	Solicitudes nuevas que aún no han sido revisadas.
En Visita	Solicitudes con visita domiciliaria programada.
Aprobadas	Solicitudes que completaron el proceso exitosamente.
Rechazadas	Solicitudes que no cumplieron los requisitos.

Figura 12.
Pantalla de gestión de solicitudes de adopción.



Elaborado por: Cristofer M.
Fuente: <https://caefugaranda.com/>

4.1. Tabla de solicitudes

Tabla 7.
Solicitudes

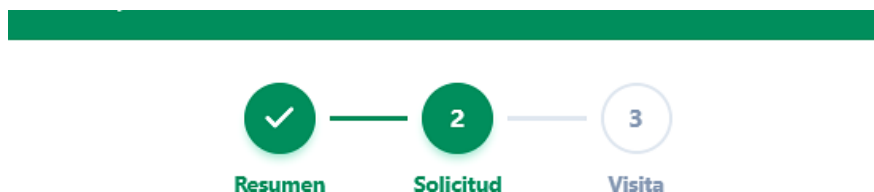
Columna	Descripción
ID	Número identificador único de la solicitud.
Solicitante	Nombre completo y cédula del ciudadano adoptante.
Mascota	Foto, nombre y categoría del animal solicitado.
Fecha Solicitud	Fecha en que el usuario envió la solicitud.
Estado	Pendiente / En Visita / Aprobada / Rechazada.
Acciones	Botón Revisar para abrir el panel de evaluación.

4.2. Proceso de revisión en tres etapas

Al hacer clic en el botón Revisar se abre el panel de evaluación con tres pasos obligatorios que deben completarse en orden.

Figura 13.

Indicador visual de los 3 pasos del proceso de revisión



Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com>

Tabla 8.

Proceso de revisión de solicitudes de adopción en tres etapas

Paso 1	Paso 2	Paso 3
Resumen	Solicitud	Visita
Revisión de datos del solicitante, mascota y acta de adopción.	Evaluación del hogar, documentos adjuntos y visita domiciliaria.	Registro del resultado de la visita y decisión final: Aprobar o Rechazar.

Paso 1 – Resumen

El administrador revisa el resumen de la solicitud antes de avanzar, donde puede ver los datos del solicitante, información de la mascota solicitada, el acta de adopción y una vista previa de los documentos adjuntos.

En este paso no hay nada que editar, hacer clic en Siguiente para continuar.

Paso 2 – Solicitud

El administrador evalúa las condiciones del hogar que el solicitante declaró y verificar la ubicación del domicilio, también quedan disponibles las fotografías de la cédula adjuntadas y la fecha y hora programadas para la visita domiciliaria.

Acciones disponibles en el Paso 2

El administrador puede marcar la solicitud como En Visita hasta que se realice la inspección y aprobarla cuando los documentos y condiciones sean satisfactorios o

rechazarla si la información no cumple los requisitos mínimos o agregar observaciones que queden registradas en el expediente.

Paso 3 – Visita

Este paso se habilita una vez realizada la visita domiciliaria, el administrador confirma que la visita se llevó a cabo y registra las observaciones sobre las condiciones del hogar. Con esa información se aprueba o rechaza la solicitud.

Resultado del proceso

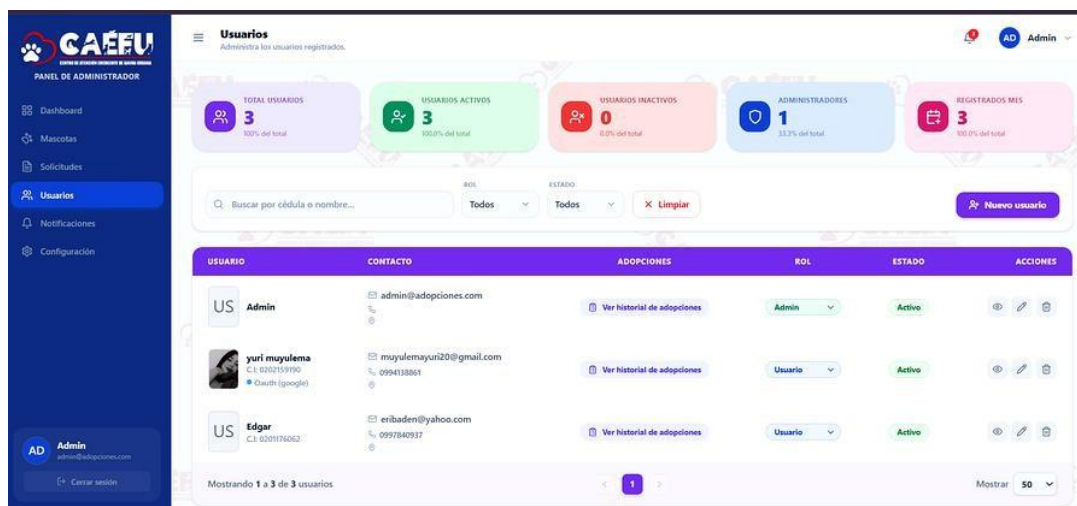
- **APROBADA:** el sistema notifica al usuario por WhatsApp como por correo electrónico, y la mascota cambia a estado “Adoptada”.
- **RECHAZADA:** el sistema notifica al usuario que su solicitud no fue aprobada y, la mascota se mantiene disponible para nuevas solicitudes.

5. Gestión de usuarios

La sección Usuarios permite ver, crear, editar y eliminar las cuentas de todos los usuarios registrados en la plataforma.

Figura 14.

Pantalla de gestión de usuarios registrados.



Elaborado por: Cristófer M.

Fuente: <https://caefugaranda.com>

5.1. Resumen de usuarios

Tabla 9.

Resumen de usuarios

Indicador	Descripción
Total Usuarios	Número total de cuentas registradas.
Usuarios Activos	Cuentas habilitadas que pueden iniciar sesión.
Usuarios Inactivos	Cuentas deshabilitadas temporalmente.
Administradores	Cuentas con rol de administrador del sistema.
Registrados el Mes	Nuevas cuentas creadas en el mes actual.

Figura 15.

Resumen de usuarios



Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com>

5.2. Tabla de usuarios

Tabla 10.

Usuarios

Columna	Descripción
Usuario	Foto de perfil, nombre completo y número de cédula.
Contacto	Correo electrónico, teléfono y dirección registrada.
Adopciones	Enlace Ver historial de adopciones del usuario.
Rol	Admin o Usuario. Modificable desde el selector desplegable.
Estado	Activo o Inactivo, indicado con etiqueta de color.

Figura 16.

Usuarios

USUARIO	CONTACTO	ADOPCIONES	ROL	ESTADO	ACCIONES
US Admin	admin@adopciones.com	Ver historial de adopciones	Admin	Activo	Ver Editar Eliminar
yuri muyulema C.I. 6202159190 Oauth (google)	muyulemayuri20@gmail.com 0994138861	Ver historial de adopciones	Usuario	Activo	Ver Editar Eliminar
US Edgar C.I. 6201176062	eribaden@yahoo.com 0997640937	Ver historial de adopciones	Usuario	Activo	Ver Editar Eliminar

Mostrando 1 a 3 de 3 usuarios

Mostrar 50

Elaborado por: Cristofer M.

Fuente: <https://caefuguaranda.com>

5.3. Crear un nuevo usuario

Se hace clic en + Nuevo usuario para registrar manualmente una cuenta desde el panel de administrador.

Figura 17.

Crear nuevo usuario

Buscar por cédula o nombre...

ROL: Todos ESTADO: Todos

[X Limpiar](#)

[+ Nuevo usuario](#)

Elaborado por: Cristofer M.

Fuente: <https://caefuguaranda.com>

5.4. Cambiar el rol de un usuario

- Localiza al usuario en la tabla.
- Haz clic en el selector desplegable de la columna Rol.
- Selecciona Admin o Usuario.
- El cambio se aplica de inmediato.

Precaución al gestionar usuarios

- Asignar el rol Admin otorga acceso completo al panel de administrador.

- Eliminar un usuario es irreversible, considerar solo desactivarlo.
- Los usuarios con Google aparecen como “Oauth (google)”

Figura 18.

Cambiar el rol de un usuario

USUARIO	CONTACTO	ADOPCIONES	ROL	ESTADO	ACCIONES
 Admin	admin@adopciones.com	Ver historial de adopciones	Admin	Activo	Ver Editar Eliminar
 yuri muyulema C.I. 0202159990 Oauth (google)	muyulemayuri20@gmail.com 0994138861	Ver historial de adopciones	Usuario	Activo	Ver Editar Eliminar
 Edgar C.I. 0201175052	eribaden@yahoo.com 0997640937	Ver historial de adopciones	Usuario	Activo	Ver Editar Eliminar

Mostrando 1 a 3 de 3 usuarios

Elaborado por: Cristofer M.

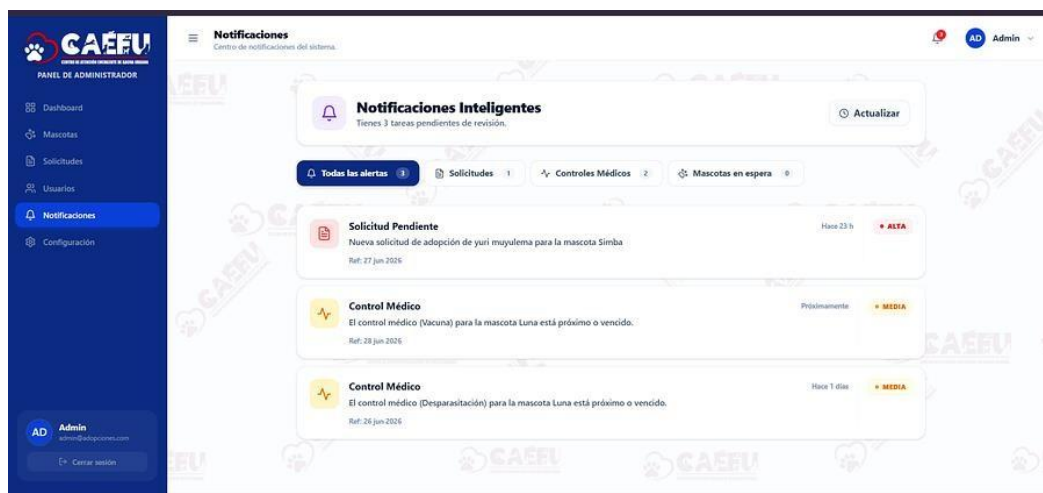
Fuente: <https://caefugaranda.com>

6. Notificaciones inteligentes

El sistema de notificaciones alerta al administrador sobre nuevas solicitudes, controles médicos próximos y animales sin solicitudes activas.

Figura 19.

Centro de notificaciones inteligentes del sistema.



Notificaciones Inteligentes
Tienes 3 tareas pendientes de revisión.

Todas las alertas 3 | Solicitudes 1 | Controles Médicos 2 | Mascotas en espera 0

Solicitud Pendiente
Nueva solicitud de adopción de yuri muyulema para la mascota Simba
Ref: 27 jun 2026

Control Médico
El control médico (Vacuna) para la mascota Luna está próximo o vencido.
Ref: 28 jun 2026

Control Médico
El control médico (Desparasitación) para la mascota Luna está próximo o vencido.
Ref: 26 jun 2026

Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com>

6.1. Tipo de notificaciones

Tabla 11.

Tipo de notificaciones

Tipo	Descripción
ALTA Solicitud Pendiente	Nueva solicitud de adopción. Requiere una revisión urgente.
MEDIA Control Médico	Vacuna o tratamiento próximo a vencer o ya vencido.
Mascotas en espera	Animales sin solicitudes activas de adopción.

Figura 20.

Tipo de notificaciones.



Elaborado por: Cristofer M.

Fuente: <https://caefugaranda.com>

6.2. Pestañas de filtro

- Todas las alertas: vista general de notificaciones pendientes.
- Solicitudes: alertas de solicitudes de adopción.
- Controles médicos: alertas de vacunas o tratamientos próximos a vencer.
- Mascotas en espera: animales sin solicitudes de adopción.

Hacer clic en Actualizar ver el listado y obtener información reciente.

Recomendación

- Revisar las notificaciones al inicio de cada jornada.
- Las alertas ALTA (rojo) se procura que se atiendan el mismo día.
- El ícono de la campana muestra el conteo de alertas no leídas.

Figura 21.

Pestañas filtro de notificaciones.



Elaborado por: Cristófer M.

Fuente: <https://caefugaranda.com>

7. Configuración de la cuenta Admin

Desde Configuración el administrador puede actualizar su información personal y cambiar su contraseña de acceso al panel.

7.1. Datos personales

Figura 22.

Configuración de datos personales del administrador

Elaborado por: Cristófer M.

Fuente: <https://caefugaranda.com>

Tabla 12.

Configuración de datos personales

Campo	Descripción
Foto de perfil	Haz clic en el ícono de cámara para cambiar la foto.
Nombre completo	Nombre del administrador registrado en el sistema.
Correo electrónico	Correo de la cuenta administradora.
Cédula de identidad	Número de cédula del administrador.
Teléfono	Número de contacto del administrador.
Fecha de nacimiento	Fecha de nacimiento en formato dd/mm/aaaa.
Dirección	Dirección de domicilio del administrador.
Campo	Descripción

Se hace clic en Guardar cambios para confirmar las modificaciones.

7.2. Seguridad – Cambiar contraseña

Figura 23.

Pantalla de cambio de contraseña del administrador

The screenshot displays the CAEFU administrator dashboard. On the left is a dark blue sidebar with the CAEFU logo and a menu including Dashboard, Mascotas, Solicitudes, Usuarios, Notificaciones, and Configuración (highlighted). The main content area is titled 'Configuración' and shows the user profile 'Admin' with email 'admin@adopciones.com'. Below the profile, there are tabs for 'Datos Personales' and 'Seguridad'. The 'Seguridad' tab is active, showing a 'Cambiar Contraseña' form. The form includes instructions: 'La contraseña debe tener al menos 8 caracteres. Usa una combinación de letras, números y símbolos.' It has three input fields: 'Contraseña actual' (with a hint 'Ingresa tu contraseña actual'), 'Nueva contraseña' (with a hint 'Mínimo 8 caracteres'), and 'Confirmar nueva contraseña' (with a hint 'Repite la nueva contraseña'). A 'Cambiar contraseña' button is at the bottom of the form. The background of the interface is watermarked with the CAEFU logo.

Elaborado por: Cristófer M.

Fuente: <https://caefugaranda.com>

Tabla 13.

Cambio de contraseña

Acción	Descripción
Contraseña actual	Se debe ingresar la contraseña actual para verificar la identidad.
Nueva contraseña	La nueva contraseña deberá tener mínimo 8 caracteres.
Confirmar contraseña	Reescritura de la nueva contraseña
Cambiar contraseña	Hacer clic en “guardar los cambios”.

Importante:

La contraseña deberá tener mínimo 8 caracteres y combinado con letras mayúsculas, minúsculas, números y símbolos. Se recomienda cambiar de contraseña periódicamente y no compartir sus credenciales con otra persona.

8. Recomendaciones operativas

Gestión Diaria

Al empezar la jornada revisar primero las notificaciones, las solicitudes se atienden en orden de llegada, el historial médico de cada mascota tiene que mantenerse al día y revisar que las fotos a publicar no estén borrosas o mal encuadradas.

Seguridad del Sistema

Se recomienda cerrar sesión al terminar, en especial en equipos compartidos. El rol Admin no se le deberá otorgar a ninguna persona que no esté autorizada por la dirección del CAEFU y ante cualquier comportamiento anómalo reportar al equipo técnico.

Anexo 8. Certificado Antiplagio



FACULTAD DE CIENCIAS
ADMINISTRATIVAS,
GESTIÓN EMPRESARIAL
E INFORMÁTICA

**ING. Rodrigo Del Pozo EN CALIDAD DE DIRECTOR(A) DEL TRABAJO
DE INTEGRACIÓN CURRICULAR,**

CERTIFICA

Que el trabajo de integración curricular denominado “**Sistema web para la gestión y adopción de mascotas en el centro de atención emergente de fauna urbana en Guaranda (CAEFU)**”, presentado por **Cristofer Fabian Muyulema Cui** estudiante de la **Carrera de Software** pasó el análisis de coincidencia no accidental en la herramienta **Compilatio**, reflejando un **porcentaje de similitud del 4%**, como se puede evidenciar en el documento adjunto.

Guaranda, 06 de Julio del 2026

Atentamente,



Ing. Rodrigo Del Pozo
Director(a)



Certificado de análisis
Compilatio Magister+ | UNIVERSIDAD ESTATAL DE BOLIVAR (UEB)

Cristofer Muyulema 6-7-2026

ID : f6ca88a46c497cb3833dcf5977e958740114a1fe



20%

Textos sospechosos

Nombre del fichero : Cristofer Muyulema 6-7-2026.txt
Tamaño del archivo original : 12,87 MB
Número de palabras : 26.124

Depositante : RODRIGO HUMBERTO DEL POZO DURANGO
Fecha de depósito : 6 de julio de 2026
Tipo de carga : interface
fecha de fin de análisis : 6 de julio de 2026

Resumen (sección 1/2)

Localización de los textos sospechosos en el documento :



Incluido en el porcentaje de textos sospechosos :



Similitudes

4%

Pasajes con similitudes a fuentes encontradas en diferentes colecciones.



Detección de IA

13%

Textos estilísticamente próximos a un texto generado por una IA.

Este índice es un indicador y no una prueba. Comprueba con el autor si domina los conocimientos mencionados en el documento.



Idiomas no reconocidos

4%

Pasajes en los que parte del vocabulario utilizado no forma parte del diccionario de la lengua.

Puede tratarse de un intento del autor de modificar el texto para evitar ser detectado.



No incluido en el porcentaje de textos sospechosos :



Textos entre comillas

1%

Pasajes entre comillas, a menudo indicativos de una cita.

Fuentes de similitudes (sección 2/2)



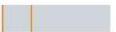
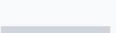
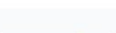
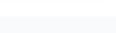
Similitudes

4%

Pasajes con similitudes a fuentes encontradas en diferentes colecciones.



Fuente principal detectada

N°	Descripciones	Similitudes	Ubicaciones
1	 dspace.ueb.edu.ec dspace.ueb.edu.ec/server/api/core/bitstreams/ba6b323e-58fa-485e-a5b2-...	<1%	
2	 LEY ORGÁNICA DE PROTECCIÓN DE DATOS PERSONALES www.dspace.espol.edu.ec/handle/123456789/57303	<1%	
3	 Estudio de la adopción de metodologías ágiles en proyectos de... doi.org/10.48082/espacios-a24v45n04p06	<1%	
4	 doi.org doi.org/http://dx.doi.org/10.53591/easi.v2i1.2043	<1%	
5	 usgn.espe.edu.ec usgn.espe.edu.ec/wp-content/uploads/2020/07/LEY-INGENIOS.pdf	<1%	
6	 Desarrollo de un sistema web y aplicación móvil para la gestión d... [Link]	<1%	
7	 Diseño de interfaces de sistemas interactivos utilizando técnicas ... doi.org/10.26439/interfases2022.n016.6028	<1%	
8	 Kanban: Una metodología ágil para la gestión eficiente del flujo d... editorialinnova.com/index.php/rig/article/view/68	<1%	
9	 Sistema Web de Gestión para adopción y recuperación de mascot... repositorio.puce.edu.ec/handle/123456789/45564	<1%	
11	 GAD-C-GUARANDA CONTINUA CON LA CAMPAÑA DE ADOPCIÓN... www.guaranda.gob.ec/newsiteCMT/gad-c-guaranda-continua-con-la-cam-...	<1%	

Fuente mencionada (sin similitudes detectadas)

N°	Descripciones
1	 https://api.green-api.com/walInstance{id}/sendMessage/{token}
2	 https://api.green-api.com
3	 https://doi.org/10.1016/j.scico.2023.103045
4	 http://www.revinfcientifica.sld.cu/index.php/ric/article/view/4099
5	 https://doi.org/10.1109/IESTEC54539.2022.00042

N°		Descripciones
6		https://doi.org/10.3390/app12094369
7		https://doi.org/10.21897/rmvz.2925
8		https://doi.org/10.1016/j.jss.2022.111389
9		https://doi.org/10.4067/S0718-33052021000100141
10		https://doi.org/10.17268/tech1.1.2026.03
11		https://doi.org/10.1145/3617175
12		https://doi.org/10.4067/S0718-33052022000200239
13		https://revistas.umecit.edu.pa/index.php/sc/article/view/1090
14		https://doi.org/10.37811/cl_rcm.v8i4.12490
15		https://doi.org/10.59814/resofro.2025.5(1)
16		https://publisher.uthm.edu.my/ojs/index.php/jastec/article/view/23632/7611
17		https://doi.org/10.1109/ACCESS.2021.3139525
18		https://doi.org/10.1145/3540250.3549144
19		https://openaccess.uoc.edu/server/api/core/bitstreams/5504f3a6-6970-4d7b-b6e7-843e124610a0/content
20		https://www.researchgate.net/profile/Luis-Cristobal-Cedeno-Valarezo/publication/374165932_Metodologi...
21		https://doi.org/10.14202/vetworld.2021.2371-2379
22		http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1684-18592023000100007&lng=es&nrm=iso&lng=es
23		https://doi.org/10.60100/rcmg.v5i2.273
24		https://www.researchgate.net/profile/Pablo-Quezada-Sarmiento/publication/365654917_Factores_de_ries...
25		https://investigacion.utc.edu.ec/index.php/utciencia/article/view/650
26		https://doi.org/10.1016/B978-0-323-90240-3.00015-1
27		https://doi.org/10.3390/electronics11091334
28		https://doi.org/10.36825/RITI.14.33.009
29		https://www.ctes.org.mx/index.php/ctes/article/view/812
30		https://doi.org/10.14569/IJACSA.2023.0140420
31		https://doi.org/10.47187/perspectivas.6.2.219
32		https://doi.org/10.3390/electronics12040935
33		https://repositorio.ucp.edu.co/entities/publication/c270893b-96e7-4d74-a591-36531f865625
34		https://doi.org/10.1016/j.procs.2021.12.150

