



PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

PUCE TEC

**TRABAJO DE INTEGRACIÓN CURRICULAR PREVIO A LA OBTENCIÓN DEL
TÍTULO DE TECNÓLOGO SUPERIOR EN DESARROLLO DE SOFTWARE**

**SISTEMA WEB DE GESTIÓN PARA ADOPCIÓN Y RECUPERACIÓN DE
MASCOTAS EXTRAVIADAS PARA LA FUNDACIÓN FAUNA URBANA IBARRA**

HIDALGO JACOME JAIRO PATRICIO

USIÑA FOLLARAN ANDY ISMAEL

TUTOR: IBARRA ESTEVEZ JOSÉ LUIS

IBARRA – ECUADOR

Marzo, 2025

Ibarra, 26 de febrero del 2025.

CERTIFICACIÓN TUTOR

En mi calidad de Tutor del Trabajo de titulación titulado:

Sistema Web De Gestión Para Adopción Y Recuperación De Mascotas Extraviadas Para La Fundación Fauna Urbana Ibarra, presentado por el estudiante Andy Ismael Usiña Follaran con cédula de ciudadanía N° 1004895049 y el estudiante Jairo Patricio Hidalgo Jácome con cédula de ciudadanía N° 1723978803, para obtener el Título de Tecnólogo Superior en Desarrollo de Software.

Certifico que el trabajo cumple con todos los parámetros establecidos, mediante el cual el estudiante demuestra el desarrollo de competencias en el campo de conocimiento de su profesión con un nivel de argumentación coherente, para ser sometido a la evaluación por parte de los lectores.

Adicionalmente, se adjunta el certificado de porcentaje de originalidad de TURNITIN.

26/3/25, 9:43 a.m. Turnitin - Informe de Originalidad - SISTEMA WEB DE GESTIÓN PARA ADOPCIÓN Y RECUPERACIÓN DE MASCOTAS EX...

Turnitin Informe de Originalidad					
Procesado el: 25-mar-2025 17:32 -05 Identificador: 2625219351 Número de palabras: 16127 Entregado: 1	<table border="1"><thead><tr><th>Índice de similitud</th><th>Similitud según fuente</th></tr></thead><tbody><tr><td>2%</td><td>Internet Sources: 2% Publicaciones: 0% Trabajos del estudiante: N/A</td></tr></tbody></table>	Índice de similitud	Similitud según fuente	2%	Internet Sources: 2% Publicaciones: 0% Trabajos del estudiante: N/A
Índice de similitud	Similitud según fuente				
2%	Internet Sources: 2% Publicaciones: 0% Trabajos del estudiante: N/A				
SISTEMA WEB DE GESTIÓN PARA ADOPCIÓN Y RECUPERACIÓN DE MASCOTAS EXTRAVIADAS PARA LA FUNDACIÓN FAUNA URBANA IBARRA Por JAIRO PATRICIO HIDALGO JACOME					
2% match (Internet desde 18-mar-2025) https://dl.dropboxusercontent.com/sc/f/tlkr85jhuvveycehlkosp/Programaci-n-Web-del-Frontend-al-Backend-Ricardo-Javier-Cel-P-rraga-Miguel-Fabrizio-Bon-Andrade-Aldo-Patricio-Mora-Olivero.pdf?dl=0&rkey=4u95mhsppg7k1ap8x6kzwx991					
PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR PUCE TEC TRABAJO DE INTEGRACIÓN CURRICULAR PREVIO A LA OBTENCIÓN DEL TÍTULO DE TECNÓLOGO SUPERIOR EN DESARROLLO DE SOFTWARE SISTEMA WEB DE GESTIÓN PARA ADOPCIÓN Y RECUPERACIÓN DE MASCOTAS EXTRAVIADAS PARA LA FUNDACIÓN FAUNA URBANA IBARRA HIDALGO JACOME JAIRO PATRICIO USIÑA FOLLARAN ANDY ISMAEL TUTOR: IBARRA ESTEVEZ JOSÉ LUIS IBARRA – ECUADOR Marzo, 2025 ii Ibarra, 26 de febrero del 2025. CERTIFICACIÓN TUTOR En mi calidad de Tutor del Trabajo de titulación titulado: Sistema Web De Gestión Para Adopción Y Recuperación De Mascotas Extraviadas Para La Fundación Fauna Urbana Ibarra, presentado por el estudiante Andy Ismael Usiña Follaran con cédula de ciudadanía N° 1004895049 y el estudiante Jairo Patricio Hidalgo Jácome con cédula de ciudadanía N° 1723978803, para obtener el Título de Tecnólogo Superior en Desarrollo de Software. Certifico que el trabajo cumple con todos los parámetros establecidos, mediante el cual el estudiante demuestra el desarrollo de competencias en el campo de conocimiento de su profesión con un nivel de argumentación coherente, para ser sometido a la evaluación por parte de los lectores. Adicionalmente, se adjunta el certificado de porcentaje de originalidad de TURNITIN. (f):					

(f): _____

Mgs. José Luis Ibarra Estevez
TUTOR DE TRABAJO
C.C.: 1002640728

PÁGINA DE APROBACIÓN DEL TRIBUNAL

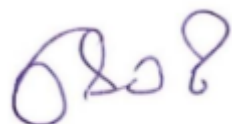
El tribunal examinador, aprueba el presente trabajo en nombre de la Pontificia Universidad Católica del Ecuador Ibarra:



(f):

Mg. José Luis Ibarra Estevez

C.C.: 1002640728



(f):.....

Mg. Galo Hernán Puetate Huera

C.C.: 0401375787



(f):.....

Mg. Álvaro Mauricio Cevallos Ramírez

C.C.: 1002494019

ACTA DE CESIÓN DE DERECHOS

Yo, *Andy Ismael Usiña Follaran* y *Jairo Patricio Hidalgo Jácome*, declaro conocer y aceptar la disposición del Art. 165 del Código Orgánico de Economía Social de los Conocimientos, Creatividad e Innovación, que manifiesta textualmente: “Se reconoce facultad de los autores y demás titulares de derechos de disponer de sus derechos o autorizar las utilidades de sus obras o prestaciones a título gratuito y oneroso, según las condiciones que determinen. Esta facultad podrá ejercerse mediante licencias libres, abiertas y otros modelos alternativos de licenciamiento o la renuncia”.

Ibarra, 27 de febrero del 2025.



(f): _____

Usiña Follaran Andy Ismael

C.C.: 1004895049



(f): _____

Hidalgo Jácome Jairo Patricio

C.C.: 1723978803

AUTORIA

Yo, *Andy Ismael Usiña Follaran* portador de la cedula de ciudadanía N° 1004895049, y *Jairo Patricio Hidalgo Jácome*, portador de la cedula de ciudadanía N° 1723978803, declaro que el presente trabajo de investigación es de total responsabilidad del autor, y eximo expresamente a la Pontificia Universidad Católica del Ecuador Ibarra de posibles reclamos o acciones legales.



(f):.....

Usiña Follaran Andy Ismael

C.C.: 1004895049



(f):.....

Hidalgo Jácome Jairo Patricio

C.C.: 1723978803

DEDICATORIA Y AGRADECIMIENTOS

A mis padres, por creer en mí, por su apoyo inquebrantable en cada paso de este camino y por ser un ejemplo de perseverancia y constancia. Gracias por inculcarme valores que han guiado mi vida y por su motivación constante, que me ha impulsado a crecer y a superarme.

Su amor y entrega han sido mi mayor inspiración, dándome la fuerza para seguir adelante incluso en los momentos más difíciles. Cada logro que alcanzo es también suyo, porque sin su apoyo, este camino habría sido imposible de recorrer.

A ustedes, les dedico este trabajo con profunda gratitud y la certeza de que todo lo que soy y he logrado es gracias a su presencia en mi vida.

Andy Usiña

ÍNDICE DE CONTENIDOS

CERTIFICACIÓN TUTOR	ii
PÁGINA DE APROBACIÓN DEL TRIBUNAL	iii
ACTA DE CESIÓN DE DERECHOS	iv
AUTORIA	v
DEDICATORIA Y AGRADECIMIENTOS.....	vi
ÍNDICE DE CONTENIDOS	vii
ÍNDICE DE TABLAS.....	x
ÍNDICE DE FIGURAS	xi
RESUMEN	xii
ABSTRACT.....	xiii
1. CAPÍTULO I	1
1.1 Aplicaciones Web realizadas.....	1
1.2 Trabajos realizados en aplicaciones móviles	2
1.3 Marco Teórico.....	4
1.3.1 Páginas y sitios Web.....	4
1.3.2 Fronted	4
1.3.3 Backend.....	4
1.3.4 HTML	4
1.3.5 CSS.....	5
1.3.6 JavaScript.....	5
1.3.7 PHP	5
1.3.8 Framework	6
1.3.9 Laravel	6
1.3.10 Tailwindcss	6
1.3.11 MySQL	7
1.3.12 Git.....	7
1.3.13 Control de versiones	7
1.3.14 Entornos integrados de desarrollo (IDE).....	8
1.4 Origen de las metodologías ágiles	9
1.4.1 Metodología XP	10
1.4.2 Fase de exploración.....	11
1.4.3 Fase de Diseño	11

1.4.4	Fase de Codificación.....	12
1.4.5	Fase de Pruebas	12
1.4.6	Fase de Lanzamiento.....	12
2	CAPITULO II	13
2.1	Generalidades de la investigación	13
2.2	Tipo de la investigación.....	13
2.3	Lugar donde se realizó la investigación.	14
2.3.1	Población.....	14
2.4	Técnicas e Instrumentos	15
2.4.1	Técnicas para la identificación de requerimientos	15
2.4.2	Entrevistas:.....	16
2.4.3	Observación directa:	16
2.5	Actores del sistema.....	16
2.5.1	Usuario administrador	17
2.5.2	Usuario normal.....	17
2.6	Especificación de requisitos del Software	17
2.6.1	Historias de usuario.....	18
2.7	Identificación de requerimientos no funcionales	25
2.8	Determinación de roles y responsabilidades.....	27
2.9	Iteraciones	29
2.9.1	Iteración 1: Planificación Inicial.....	30
2.9.2	Iteración 2: Publicación de Mascotas.....	31
2.9.3	Iteración 3: Visualización y gestión de perfiles	32
2.9.4	Iteración 4: Solicitud y aprobación de adopción.....	33
2.9.5	Iteración 5: Visualización y gestión de adopciones.....	34
2.9.6	Iteración 6: Gestión de adopciones y historial de solicitudes.....	35
2.10	Diseño de la Interfaz Gráfica (GUI).....	37
2.11	Diseño de la Base de Datos	39
2.11.1	Modelo relacional	39
2.11.2	Diagrama relacional	40
2.11.3	Modelo conceptual	41
2.11.4	Modelo Lógico.....	42
2.12	Diseño de la arquitectura de Software.....	43
2.12.1	Modelo arquitectónico.....	43
2.12.2	Estilos de capas	43
2.12.3	Diagrama de clases:.....	44

2.12.4	Descripción de las clases principales.....	45
2.13	Diseños de casos de prueba	47
3	CAPITULO III	48
3.1	Resultados de la implementación del sistema	48
3.1.1	Módulo de interfaz principal.....	48
3.1.2	Módulo de login	49
3.1.3	Módulo de registro.....	50
3.1.4	Módulo de mascotas.....	51
3.1.5	Módulo de localización de mascotas	55
3.1.6	Módulo crear adopción.....	57
3.1.7	Módulo de administrador	58
3.2	Resultados funcionales	72
3.3	Resultados de las pruebas realizadas	72
3.3.1	Análisis de los resultados	73
3.3.2	Facilidad de uso.....	73
3.3.3	Eficiencia:	73
3.3.4	Visibilidad y alcance:	74
3.3.5	Escalabilidad y mantenimiento:	74
	CONCLUSIONES.....	75
	BIBLIOGRAFÍA	78
4	ANEXOS.....	80

ÍNDICE DE TABLAS

Tabla 1 Registro de usuarios en el sistema	18
Tabla 2 Inicio y cierre de sesión.....	18
Tabla 3 Visualización de la página principal	19
Tabla 4 Visualización de instrucciones de adopción	19
Tabla 5 Publicación de mascotas en adopción	20
Tabla 6 Visualización de la lista de mascotas en adopción	20
Tabla 7 Visualización del perfil de la mascota	21
Tabla 8 Edición y visualización del perfil de usuario	21
Tabla 9 Solicitud de adopción	22
Tabla 10 Aprobación de solicitudes de adopción	22
Tabla 11 Visualización de sección de adopciones	23
Tabla 12 Gestión de perfil de mascota.....	23
Tabla 13 Gestión de notificaciones	24
Tabla 14 Gestión de perfiles de usuario (Administrador)	24
Tabla 15 Gestión de publicaciones de mascotas (Administrador)	25
Tabla 16 Rendimiento	25
Tabla 17 Usabilidad	26
Tabla 18 Seguridad	26
Tabla 19 Compatibilidad	26
Tabla 20 Escalabilidad	27
Tabla 21 Mantenibilidad	27
Tabla 22 Roles y responsabilidades de los integrantes.....	27
Tabla 23 Análisis y refinamiento de las historias de usuario	28
Tabla 24 Distribución de Sprints	29
Tabla 25 Casos de Prueba	47
Tabla 26 Resultados de las pruebas realizadas	72

ÍNDICE DE FIGURAS

Figura 1 <i>Página de Inicio</i>	37
Figura 2 <i>Área principal</i>	38
Figura 3 <i>Footer</i>	38
Figura 4 <i>Diagrama relacional de la Base de Datos</i>	40
Figura 5 <i>Modelo conceptual fauna urbana</i>	41
Figura 6 <i>Modelo lógico fauna urbana</i>	42
Figura 7 <i>Diagrama de clases</i>	44
Figura 8 <i>Módulo de interfaz principal</i>	48
Figura 9 <i>Interfaz de login</i>	49
Figura 10 <i>Interfaz de registro</i>	50
Figura 11 <i>Interfaz de selección de mascota</i>	51
Figura 12 <i>Interfaz de catálogo de perros</i>	52
Figura 13 <i>Interfaz de ver mascota</i>	53
Figura 14 <i>Interfaz de catálogo de gatos</i>	54
Figura 15 <i>Interfaz de mascotas perdidas</i>	55
Figura 16 <i>Formulario publicar mascota perdida</i>	56
Figura 17 <i>Formulario publicar mascota en adopción</i>	57
Figura 18 <i>Interfaz de Administrador</i>	58
Figura 19 <i>Administrador - agregar categorías</i>	60
Figura 20 <i>Administrador - ingreso de nuevas mascotas</i>	61
Figura 21 <i>Administrador - formulario de ingreso nuevas mascotas</i>	62
Figura 22 <i>Administrador - esterilizaciones</i>	64
Figura 23 <i>Administrador - formulario esterilizaciones</i>	65
Figura 24 <i>Administrador - esterilización liberación</i>	66
Figura 25 <i>Administrador - formulario esterilización liberación</i>	67
Figura 26 <i>Administrador - medicación</i>	68
Figura 27 <i>Administrador - formulario medicación</i>	69
Figura 28 <i>Administrador - mascotas para adopción</i>	70
Figura 29 <i>Administrador - formulario mascotas para adopción</i>	71

RESUMEN

Sistema web para adopción y localización de mascotas, implementado en la Fundación Fauna Urbana de Ibarra. Su objetivo principal es optimizar los procesos de gestión de adopciones y recuperación de animales extraviados, mejorando la interacción entre adoptantes y administradores. A través de una metodología ágil (eXtreme Programming), se construyó una plataforma que permite a los usuarios registrarse, buscar mascotas y reportar animales perdidos. Por su parte, los administradores pueden gestionar solicitudes de adopción, actualizar el estado de las mascotas y registrar animales extraviados.

Como resultado, el sistema implementa medidas de seguridad robustas, incluyendo la encriptación de contraseñas y un proceso de autenticación seguro, protegiendo así los datos personales. Además, su interfaz amigable facilita la adopción responsable al mejorar la visibilidad de las mascotas y optimizar la búsqueda. La integración de una sección dedicada a la recuperación de mascotas perdidas proporciona una experiencia completa y eficiente. La plataforma es escalable, lo que permite su expansión conforme la Fundación crezca.

En conclusión, la implementación de este sistema web agiliza los procesos de adopción y promueve el bienestar animal, contribuyendo a una comunidad más consciente y responsable.

Palabras clave: adopción de mascotas, localización de mascotas perdidas, sistema web para adopción, gestión de adopciones y mascotas, plataforma digital de adopción.

ABSTRACT

This web system for pet adoption and lost pet location was implemented at the Fauna Urbana Foundation in Ibarra, aiming to optimize the processes of managing adoptions and recovering lost animals, thereby improving interactions between adopters and administrators. By employing the agile eXtreme Programming (XP) methodology, a platform was developed that enables users to register, search for pets, and report missing animals, while administrators can manage adoption requests, update pet statuses, and record lost animals.

As a result, robust security measures were integrated, including password encryption and a secure authentication process, safeguarding users' personal information. Additionally, the user-friendly interface facilitates responsible adoption by enhancing pet visibility and streamlining the search process. The platform also includes a dedicated section for lost pets, providing a comprehensive and efficient user experience. Moreover, the system is scalable, allowing it to expand as the Foundation grows.

In conclusion, the implementation of this web-based solution accelerates adoption processes and promotes animal welfare, thus contributing to a more conscious and responsible community.

Keywords: pet adoption, lost pet location, web system for adoption, adoption and pet management, digital adoption platform.

1. CAPÍTULO I

ESTADO DEL ARTE

El estado del arte constituye la base esencial para el desarrollo de cualquier investigación, pues brinda un contexto teórico y conceptual que guía el estudio y facilita la interpretación de los resultados. En este capítulo, se exponen trabajos previos, teorías y conceptos relevantes que respaldan el proyecto, contribuyendo a una mejor comprensión del problema de estudio y ofreciendo una perspectiva enriquecedora para el análisis del sistema. A través de la revisión de documentación existente, se han examinado diversas soluciones y tecnologías aplicadas en el ámbito del bienestar animal, particularmente en la adopción de mascotas y la localización de animales extraviados. Este proceso ha permitido identificar las soluciones desarrolladas hasta el momento, reconocer las carencias que persisten y destacar los enfoques y tecnologías innovadoras utilizadas. Toda la información recabada ha sido fundamental para perfeccionar y validar el sistema propuesto en este trabajo.

1.1 Aplicaciones Web realizadas

En el campo del desarrollo Web destinados en la adopción de mascotas se han desarrollado diversos trabajos, que han abordado distintas tecnologías para mejorar el sistema de adopción. A continuación, se presentan los trabajos más recientes en el campo.

(Triviño Zambrano, 2018) Desarrolló una aplicación Web cuyo propósito de este trabajo fue el de desarrollar un sistema de información con aplicaciones Web y móvil para la gestión de las adopciones de animales rescatados por la Fundación Rescate Animal. Se logró habilitar las eficiencias y ventajas que un sistema informático introduce y además normalizar, simplificar y automatizar el proceso de adopción de animales rescatados logrando un impacto positivo. Se utilizó PostgreSQL, Python, y Java.

El objetivo principal de la aplicación Web desarrollada por (Borja Machin, 2022) fue facilitar el proceso de adopción y aumentar la visibilidad de los animales que buscan un hogar. Con una plataforma digital accesible desde cualquier dispositivo, se busca que las protectoras de animales y personas particulares puedan subir y gestionar anuncios con fotos, descripciones, y características específicas de cada mascota, como su edad, tamaño, raza, y estado de salud. Para este proyecto se utilizaron para su construcción las siguientes tecnologías HTML, CSS, JavaScript, Angular y MongoDB.

En el trabajo de (Gutierrez Benitez, 2020) se implementó una plataforma Web específica para el manejo de registros de los animales y temas de adopción. Los beneficios que se buscaba era la transparencia de registros de cada animal adaptando perfiles individuales para cada animal que esté disponible para adoptar, lo cual mejorara el manejo de datos y actualizaciones de registros.

(Colcha Conejo, 2022) En su trabajo nos menciona que, los sistemas Web en la actualidad son muy importantes para simplificar muchos procesos de manera automática, es por esto que ha desarrollado el componente *frontend* de un sistema de adopción que facilita la adopción de animales rescatados de la calle, colocando a disposición varias mascotas listas para ser adoptadas, el proceso de adopción se realiza mediante un formulario que contiene preguntas de diferente índole, en donde el usuario propietario de la publicación tiene la capacidad de aceptar o denegar las solicitudes de adopción, la herramienta utilizada fue NextJs.

1.2 Trabajos realizados en aplicaciones móviles

En el ambiente de desarrollo de Software que albergan sistemas de adopción de mascotas, se han implementado diversas aplicaciones móviles que sirven como referencia para este proyecto como se mencionan a continuación.

(Arcos Arcos, 2024) El propósito de este proyecto fue implementar una aplicación móvil para la adopción de mascotas rescatadas. En este contexto se usó técnicas de recolección de datos tales como encuestas y entrevistas para identificar los problemas del refugio y de igualmente se recopiló los requerimientos para el desarrollo del aplicativo, se utilizó React Native.

(Andrade Tituaña, 2018) El Sistema desarrollado permitió cumplir una gran parte de la Ordenanza de Tenencias y Regulación de Mascotas expedida del GAD Municipal de Otavalo en el año 2014. El Sistema Georreferenciado de Registro y Control de Mascotas comenzó con el estudio y el uso de los mapas geográficos utilizando el API de Google V3, se diseñó e implementó los módulos de Registro de Propietarios, Registro de Mascotas, Registro de Esterilizaciones, Registro de Vacunas, Mapa con datos georreferenciados, Generación de carnets y fichas clínica, Emisión de Informes Estadísticos y Registro de Denuncias, Para su implementación se utilizó Flutter.

(Pérez Ruíz, 2019) Elaboró una aplicación móvil para facilitar la adopción de mascotas. La aplicación, llamada Adoptapp, está disponible para sistemas operativos Android, con una interfaz similar a una aplicación de servicios por medio de anuncios. Además, el usuario tiene la opción de elegir su publicación como persona individual o fundación/organización sin ánimo de lucro, para esta aplicación utilizó ReactNative con Android Studio para su construcción.

(Pata Vargas, 2024) Propuso un prototipo de aplicación móvil que les permite a albergues y fundaciones gestionar la adopción de mascotas, así como a los usuarios el adoptar una mascota e informar en caso de que se pierda, evitando así largos procesos de adopción por parte de los usuarios, y el ahorro de tiempo y recursos a los administradores de un albergue, las tecnologías que se utilizó en el proyecto fué Flutter.

1.3 Marco Teórico

1.3.1 Páginas y sitios Web

Una página o documento Web es todo archivo electrónico con un formato apropiado para su transmisión a través de los protocolos de la Web. Su principal característica es la utilización de enlaces o hipervínculos, referencias cruzadas que sirven para interconectar unos documentos con otros a través de la Red (Hernández Bejarano y Baquero Rey, 2020).

1.3.2 Fronted

El primer contacto por parte de nuestros usuarios, es a través de su interfaz visual, (Autentia, 2022) el *frontend* se refiere a la parte visible de una aplicación Web, lo que los usuarios ven y con lo que interactúan. El *frontend* se compone principalmente de lenguajes de programación como HTML, CSS y JavaScript, que permiten la creación de páginas Web dinámicas y atractivas.

1.3.3 Backend

El *backend* tiene un papel imprescindible en el desarrollo de una aplicación Web ya que este administra la lógica que se ejecutan detrás de las interacciones que hacen los usuarios como nos comenta (Celi Párraga et al., 2023) el *backend* se encarga de la lógica detrás de una aplicación Web. Esto incluye la gestión de la Base de Datos, la seguridad y la lógica de negocio de la aplicación. Los desarrolladores de *backend* utilizan lenguajes de programación como PHP, Ruby on Rails, Python y Java, entre otros, para crear aplicaciones Web escalables y robustas que puedan manejar grandes volúmenes de tráfico y datos.

1.3.4 HTML

HTML son las siglas de *HyperText Markup Language* (Lenguaje de Marcas de Hipertexto en español), es el lenguaje de marcado predominante para la construcción de páginas Web. Es usado para describir la estructura y el contenido en forma de texto, así como

para complementar el texto con objetos tales como imágenes. HTML se escribe en forma de “etiquetas” o “marcas”, rodeadas por corchetes angulares (< >) (Celi Párraga et al., 2023).

1.3.5 CSS

CSS (*Cascading Style Sheets* u Hojas de Estilo en Cascada) es un lenguaje de programación que permite aplicar estilos a los distintos elementos de las páginas Web, de modo que los títulos, listas y párrafos pueden verse igual en todas y cada una de las páginas (Celi Párraga et al., 2023).

1.3.6 JavaScript

Es la tercera tecnología fundamental para el desarrollo Web del lado del cliente, esta tecnología posibilita la interacción con el usuario. La estandarización de JavaScript con HTML5 en todos los navegadores nos permite crear innumerables tipos de aplicaciones Web que pueden ejecutarse en cualquier plataforma (tanto de escritorio como móvil) (Celi Párraga et al., 2023).

En resumen, HTML, CSS y JavaScript son los pilares fundamentales de la programación Web, HTML es el lenguaje de marcado utilizado para estructurar el contenido de una página Web, como texto, imágenes y enlaces, definiendo su estructura básica. CSS se encarga de como se muestra, permitiendo personalizar estilos como los colores, tipografía y la disposición de los elementos para lograr un aspecto más atractivo. JavaScript añade interactividad a las páginas, permitiendo la visualización dinámica de elementos y la respuesta a acciones del usuario como clics y desplazamientos.

1.3.7 PHP

PHP (acrónimo recursivo de PHP: *Hypertext Preprocessor*) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo Web y que puede ser incrustado en HTML (PHP, n.d.).

Lo que diferencia a PHP del lado del cliente de otros lenguajes como *Javascript* es que el código es realizado en el servidor, generando HTML y enviándolo al cliente. El cliente recibirá el resultado de ejecutar el script, aunque no se sabrá el código subyacente que era.

1.3.8 Framework

Un *framework* es un entorno de trabajo para el desarrollo de aplicaciones, ya sean Web o de escritorio, que ofrece componentes que facilitan el trabajo a los programadores, tales como bibliotecas de funciones, uso de plantillas, administración de recursos en tiempo de ejecución y otras muchas cosas. Esto permite llevar a cabo el proyecto sin tener que escribir mucho código, consiguiendo que el trabajo sea más eficiente y recursivo (Axarnet, 2018).

1.3.9 Laravel

Laravel es un *framework* de PHP y es utilizado para desarrollar aplicaciones Web, crea un entorno de trabajo y proporciona herramientas a los desarrolladores para ayudarles a desarrollar lo que se busca, es construir aplicaciones sólidas y estables, que sean fáciles de desarrollar, la utilización de parte del código preprogramada, para que pueda aprovecharse y reutilizarse, evitando así la reescritura del código en la misma aplicación (Axarnet, 2018).

1.3.10 Tailwindcss

Es un marco de trabajo (*framework*) CSS basado en clases de utilidad que se aplican directamente en las etiquetas HTML para crear cualquier diseño de interfaz de usuario, Con clases predefinidas que puedes utilizar para construir y diseñar páginas web directamente en tu marcado. Te permite escribir CSS en tu HTML en forma de clases predefinidas (Vargas Zermeño, 2022).

1.3.11 MySQL

MySQL es un sistema de administración de bases de datos relacional. Se trata de un programa capaz de almacenar una enorme cantidad de datos de gran variedad y de distribuirlos para cubrir las necesidades de cualquier tipo de organización, desde pequeños establecimientos comerciales a grandes empresas y organismos administrativos. MySQL utiliza el lenguaje de consulta estructurado (SQL). Se trata del lenguaje utilizado por todas las bases de relacionales, que presentaremos en una sección posterior. Este lenguaje permite crear bases de datos, así como agregar, manipular y recuperar datos en función de criterios específicos (Gilfilan, 2021).

1.3.12 Git

Git es un Software de control de versión, diseñado para gestionar y realizar seguimiento de los cambios en el código base de un proyecto, facilita la colaboración entre desarrolladores, donde el desarrollo está a cargo de un equipo personas, Git se convierte en una herramienta clave para planificar y de esta manera organizar, gestionar y coordinar el trabajo colaborativo de la mejor manera. Ya sea que se trabaje de manera simultánea o también mediante el uso de ramas separadas, cada desarrollador trabaja de forma independiente (Moure, 2024).

1.3.13 Control de versiones

El control de versiones es una parte esencial de cualquier proyecto de Software. Control de versiones le ayuda a rastrear cambios en su código, colaborar con otros y volver a versiones anteriores cuando sea necesario. Git es el sistema de control de versiones más utilizado en la actualidad en el desarrollo de Software. Git permite a los desarrolladores trabajar en varias ramas, fusionar sus cambios y colaborar de manera eficaz. Algunas de las plataformas populares que utilizan Git incluyen GitHub, GitLab y Bitbucket. Subversion o SVN, es menos

común pero aún se utiliza en entornos corporativos. SVN sigue un enfoque de control de versiones centralizado, a diferencia del enfoque distribuido de Git. (Coronado Garcia, 2024)

Un aporte relevante sería destacar que, además de las plataformas mencionadas como GitHub, GitLab y Bitbucket, la integración continua (CI) y la entrega continua (CD) se han vuelto prácticas fundamentales en el desarrollo de Software moderno. Estas herramientas permiten automatizar la construcción, prueba e implementación del código, mejorando la eficiencia del flujo de trabajo y minimizando los errores. Además, el uso de sistemas de control de versiones como Git no solo facilita la colaboración entre desarrolladores, sino que también ayuda a mantener la integridad del código, garantizando que se pueda realizar un seguimiento preciso de los cambios y facilitando la resolución de conflictos cuando varios desarrolladores trabajan sobre el mismo archivo o funcionalidad.

1.3.14 Entornos integrados de desarrollo (IDE)

Un entorno de desarrollo integrado (IDE) es más que un simple editor de texto; es un paquete de Software completo que incluye herramientas para escribir, depurar, compilar y probar código. Los IDE son particularmente útiles en proyectos grandes donde todas estas herramientas deben integrarse de manera eficiente. Algunos de los IDE más comunes incluyen: *Visual Studio*, desarrollado por Microsoft para aplicaciones .NET y C++, que proporciona funciones avanzadas como depuración en tiempo real, integración con *Azure* y herramientas de análisis de código IntelliJ IDEA, para usar con *Java*, *Kotlin* y *Scala*, etc. Muy popular en el desarrollo de otros lenguajes JVM, conocido por su eficiente finalización de código e integración con herramientas de desarrollo modernas, *PyCharm*, creado específicamente para *Python* por *JetBrains*, que utiliza *Django* y *Flask*; *Frameworks* como Eclipse brindan capacidades avanzadas de depuración, pruebas y desarrollo Web. Eclipse es un IDE de código abierto que soporta muchos lenguajes, aunque destaca por su compatibilidad con Java y la

amplia gama de complementos que permiten personalizar su funcionalidad (Coronado Garcia, 2024)

Un aporte importante es enfatizar que la elección del IDE depende no solo del lenguaje de programación utilizado, sino también de las necesidades específicas del proyecto y del equipo de desarrollo. Por ejemplo, en proyectos grandes, la colaboración y la integración con herramientas de control de versiones como Git son cruciales, y algunos IDE (como *Visual Studio* e *IntelliJ IDEA*) proporcionan funciones que facilitan esta integración. Además, al personalizar el IDE mediante complementos como Eclipse, los desarrolladores pueden adaptar el entorno a sus preferencias y necesidades específicas, aumentando así la productividad y la eficiencia del desarrollo.

1.4 Origen de las metodologías ágiles

Para comprender el origen y la razón de ser de las metodologías ágiles, es necesario retroceder a 1968, cuando se identificó la "Crisis del Software", un problema caracterizado por retrasos, sobrecostos y deficiencias de calidad en el desarrollo de Software. Se intentó solucionar este problema mediante la ingeniería del Software, la gestión predictiva de proyectos y la producción basada en procesos, con el objetivo de garantizar la calidad en tiempo, coste y eficiencia. Sin embargo, la gestión predictiva no fue efectiva en entornos dinámicos y cambiantes, donde los requisitos evolucionaban rápidamente. En 1986, Takeuchi y Nonaka propusieron un enfoque diferente basado en ejemplos de empresas exitosas, que aplicaban métodos flexibles frente a los modelos rígidos. Este enfoque dio paso al surgimiento del "agilismo", como respuesta a un entorno competitivo y cambiante donde la innovación y la rapidez en los lanzamientos se volvieron clave. A mediados de la década de 1990, el desarrollo ágil surgió como una reacción contra los métodos tradicionales y burocráticos, como el modelo en cascada. Estos métodos ágiles, inicialmente conocidos como "peso liviano", se consolidaron en 2001, cuando un grupo de expertos en metodologías ágiles se reunió en Snowbird, Utah, y

adoptó el término "Métodos ágiles", dando lugar a la formación de la "Alianza Ágil", una organización que promueve el desarrollo ágil de Software. (Fonseca Vargas , 2020)

El texto ofrece una visión detallada sobre el origen y evolución de las metodologías ágiles, destacando cómo surgieron como respuesta a los problemas recurrentes en el desarrollo de Software, tales como los retrasos y sobrecostos, derivados de enfoques rígidos y burocráticos como el modelo en cascada. La transición hacia métodos más flexibles y adaptativos refleja una necesidad urgente de adaptarse a entornos de trabajo dinámicos y competitivos, donde la rapidez y la innovación son cruciales para mantenerse relevante. La adopción del término "métodos ágiles" y la formación de la "Alianza Ágil" en 2001 consolidaron una nueva forma de abordar el desarrollo de Software, priorizando la colaboración, la iteración y la flexibilidad frente a la planificación estricta. Este enfoque ha demostrado ser mucho más eficiente en proyectos donde los requisitos cambian constantemente, promoviendo una mejora continua y permitiendo a las organizaciones responder de manera más ágil a las demandas del mercado.

1.4.1 Metodología XP

En este proyecto se aplica la metodología ágil *eXtreme Programming*, (XP) para aprender técnicas y evaluar los beneficios obtenidos durante y después del desarrollo.

Como (Fonseca Vargas , 2020) *eXtreme Programming*, se centra en mejorar las relaciones interpersonales, considerando el trabajo en equipo y el aprendizaje continuo de los desarrolladores como factores clave para el éxito en el desarrollo de Software. Esta metodología fomenta la retroalimentación constante entre el cliente y el equipo, una comunicación fluida entre todos los miembros, la simplicidad en las soluciones implementadas y la valentía para aceptar y adaptarse a los cambios. XP es especialmente adecuada para proyectos con requisitos vagos y cambiantes, así como en situaciones de alto riesgo técnico.

El autor en su cita destaca adecuadamente cómo la metodología XP se enfoca en la colaboración y la flexibilidad, aspectos esenciales para el éxito en entornos de desarrollo ágil. Al poner énfasis en las relaciones interpersonales y la retroalimentación continua, XP no solo favorece la calidad del producto, sino también el bienestar del equipo de trabajo. Un aporte relevante sería que, en la práctica, XP también pone gran énfasis en las pruebas automatizadas y en la integración continua, lo que permite a los equipos detectar errores tempranamente y adaptarse rápidamente a los cambios sin comprometer la calidad del Software. Esta capacidad de adaptarse a requisitos cambiantes y mantener una comunicación constante con el cliente es lo que realmente permite que XP sea una metodología eficiente en proyectos con alto nivel de incertidumbre.

1.4.2 Fase de exploración

En el caso de XP, la exploración toma la forma de tres procesos: las historias de usuario y la planificación del proyecto. El proceso comienza con las “historias de usuario”, que son relatos de las funcionalidades que los usuarios desean en la aplicación. A diferencia de los casos de uso, las historias de usuario son escritas por los usuarios y no por los analistas, y están escritas en un lenguaje verdaderamente coloquial, sin ninguna terminología especializada. Estos pensamientos se utilizan como base para crear las pruebas de aceptación, que garantizarán que la funcionalidad haya sido entendida y construida de la manera correcta. Por supuesto, esta piratería no es tan detallada como el análisis de requisitos. Dado que el cliente es parte del equipo en lugar de un árbitro externo, cualquier confusión que aparezca se puede eliminar durante el ciclo de implementación, por lo que las historias solo necesitan la cantidad adecuada de detalle (Fonseca Vargas , 2020).

1.4.3 Fase de Diseño

Una vez definidas las historias de usuario a implementar, el equipo procede a diseñar soluciones simples y eficaces. XP promueve la creación de una arquitectura flexible que facilite

la adaptación a cambios de requerimientos. En esta fase se fomenta la colaboración entre los desarrolladores, quienes se encargan de plantear diagramas, esquemas y maquetas que orienten la codificación (Fonseca Vargas , 2020).

1.4.4 Fase de Codificación

La codificación se lleva a cabo en parejas, práctica que incrementa la calidad del código y el aprendizaje compartido. Durante esta etapa, se implementan las funcionalidades descritas en las historias de usuario, manteniendo un enfoque en la simplicidad y la claridad. El trabajo colaborativo permite identificar y corregir errores de manera temprana, reduciendo los riesgos de retrabajo (Fonseca Vargas , 2020).

1.4.5 Fase de Pruebas

XP enfatiza la realización de pruebas automatizadas tanto unitarias como de aceptación desde etapas iniciales. Esto garantiza la calidad del software y la corrección de los requisitos implementados. El equipo ejecuta pruebas continuamente para detectar fallos de forma temprana, lo que contribuye a un proceso de integración continua y a una retroalimentación constante con el cliente (Fonseca Vargas , 2020).

1.4.6 Fase de Lanzamiento

Finalizadas las iteraciones planificadas y verificadas las pruebas, se realiza el lanzamiento de una versión funcional. El cliente recibe la solución en su estado actual y puede usarla o validarla, lo que permite recopilar retroalimentación real y planificar mejoras o nuevas funcionalidades para las siguientes iteraciones. Esta entrega periódica y continua del software constituye uno de los pilares fundamentales de XP, al mantener una comunicación fluida con el cliente y asegurar la satisfacción de sus necesidades (Fonseca Vargas , 2020).

2 CAPITULO II

Este capítulo presenta los materiales y métodos utilizados en la investigación para el desarrollo del sistema Web de gestión para la adopción y recuperación de mascotas extraviadas para la Fundación Fauna Urbana en Ibarra. Se detallan las generalidades de la investigación, el tipo de investigación, el lugar donde se realizó el estudio, la población involucrada, así como las técnicas e instrumentos utilizados para la recolección de datos y el desarrollo del sistema. La metodología adoptada se fundamenta en la necesidad de crear una solución efectiva que aborde los desafíos actuales en la gestión de adopciones y recuperación de mascotas.

2.1 Generalidades de la investigación

La investigación se llevó a cabo con el objetivo de diseñar y desarrollar un sistema Web que facilite la adopción y recuperación de mascotas extraviadas. Este sistema busca mejorar la eficiencia de los procesos actuales en la Fundación Fauna Urbana, permitiendo una mejor gestión de la información y una mayor visibilidad de los animales disponibles para adopción. La investigación se centró en la identificación de requerimientos, el desarrollo de la aplicación y la evaluación de su funcionalidad. Se utilizó un enfoque aplicado, orientado a resolver problemas prácticos en el contexto de la Fundación.

2.2 Tipo de la investigación

La investigación fue de tipo aplicada, ya que buscó resolver un problema específico relacionado con la gestión de adopciones y recuperación de mascotas en la Fundación Fauna Urbana. Este enfoque permitió implementar soluciones prácticas y efectivas que beneficiaran a la organización y a la comunidad. La investigación aplicada se caracteriza por su orientación hacia la resolución de problemas concretos, lo que la convierte en un enfoque adecuado para el desarrollo de sistemas informáticos que aborden necesidades específicas.

2.3 Lugar donde se realizó la investigación.

El estudio se realizó en la ciudad de Ibarra, específicamente en la Fundación Fauna Urbana. Esta organización se dedica al rescate y bienestar de animales, y su misión incluye promover la adopción responsable de mascotas. La Fundación cuenta con un equipo de profesionales y voluntarios que trabajan en la atención y cuidado de los animales rescatados. La elección de este lugar para la investigación se debió a la necesidad identificada de mejorar los procesos de adopción y recuperación de mascotas, así como a la disposición de la Fundación para colaborar en el desarrollo del sistema.

2.3.1 Población

La población de la Fundación Fauna Urbana está compuesta por el gerente, un veterinario y un equipo de al menos cinco trabajadores que desempeñan diversas funciones en la organización. A continuación, se describen los roles más relevantes dentro de la Fundación:

- **Gerente:** El gerente es responsable de la dirección general de la Fundación. Su función principal es coordinar las actividades diarias, gestionar los recursos humanos y financieros, y establecer relaciones con otras organizaciones y la comunidad. Además, el gerente supervisa el cumplimiento de las políticas y procedimientos de la Fundación, asegurando que se alcancen los objetivos establecidos.

- **Veterinario:** El veterinario se encarga de la atención médica de los animales rescatados. Esto incluye realizar exámenes de salud, administrar tratamientos y vacunas, y llevar un registro de la salud de cada animal. Además, el veterinario juega un papel crucial en la educación de los adoptantes sobre el cuidado y bienestar de las mascotas.

- **Coordinador de Adopciones:** Este rol es fundamental para gestionar el proceso de adopción. El coordinador de adopciones se encarga de evaluar a los posibles adoptantes,

realizar entrevistas y visitas a domicilio, y asegurar que los animales sean colocados en hogares adecuados. También organiza eventos de adopción y campañas de concienciación.

- **Cuidador de Animales:** Los cuidadores son responsables del bienestar diario de los animales en la Fundación. Esto incluye alimentar, bañar y proporcionar atención básica a los animales. También son responsables de mantener la limpieza de las instalaciones y asegurar que los animales tengan un ambiente seguro y saludable.

- **Voluntarios:** Los voluntarios desempeñan un papel vital en la Fundación, ayudando en diversas tareas, desde el cuidado de los animales hasta la organización de eventos y campañas de adopción. Su apoyo es fundamental para el funcionamiento diario de la organización y para aumentar la visibilidad de la Fundación en la comunidad.

- **Responsable de Comunicación y Marketing:** Este profesional se encarga de gestionar la comunicación interna y externa de la Fundación. Su trabajo incluye la creación de contenido para redes sociales, la elaboración de boletines informativos y la promoción de eventos. Su objetivo es aumentar la visibilidad de la Fundación y atraer a más adoptantes y donantes.

- **Técnico en Gestión de Datos:** Este rol es esencial para el manejo de la información relacionada con los animales y las adopciones. El técnico en gestión de datos se encarga de mantener actualizada la Base de Datos de la Fundación, asegurando que toda la información esté correctamente registrada y sea accesible para el equipo.

2.4 Técnicas e Instrumentos

2.4.1 Técnicas para la identificación de requerimientos

En el proceso de identificar los requerimientos del sistema, se implementó técnicas, que tienen como finalidad conseguir información detallada sobre los usuarios involucrados (administradores, adoptantes y público general).

A continuación, se describen las técnicas utilizadas en este proceso:

2.4.2 Entrevistas:

Se realizó entrevistas con el personal que conforman el centro temporal de rescate Fauna Urbana Ibarra para entender los problemas en la gestión de adopción de mascotas. En esta entrevista se identifican las funcionalidades y requerimientos claves que el sistema tiene que incluir, como la publicación de mascotas que llegan al centro temporal de rescate, para que de esta manera llegue a mucha más población que estén buscando adoptar mascotas, ya sean felinos o caninos, también la necesidad de elaborar una sección específica la cual la comunidad o el centro temporal de rescate permita publicar información sobre sus mascotas perdidas, de igual forma una funcionalidad que permita registrar mascotas encontradas en la calle.

2.4.3 Observación directa:

Se realizó una observación directa de los procesos actuales del centro temporal de rescate, Esta práctica permite reconocer áreas en las cuales mejorar, mediante la implementación del aplicativo Web para ayudar en el proceso de adopción de mascotas. En el contexto actual, de la era digital, está claro que crear una aplicación Web es una opción más viable y eficiente. Este planteamiento se ajusta a la tendencia mundial de empresas que se trasladan al entorno digital para agilizar sus procesos. Este análisis me ofrece una perspectiva práctica y realista de las operaciones diarias del centro temporal de rescate, con este aplicativo Web, se busca no solo modernizar los procesos, si no promover la interacción entre el centro de adopción y la comunidad.

2.5 Actores del sistema

Para el sistema Web de adopción de mascotas para el centro de rescate temporal fauna urbana Ibarra, se han identificado los siguientes actores o usuarios que interactúan con el sistema:

2.5.1 Usuario administrador

Estos usuarios son responsables de la gestión integral del sistema. Sus responsabilidades en el apartado de gestión de mascotas son registrar nuevas mascotas, actualizar su información, cambiar el estado de la mascota (adoptada, no disponible), revisar, aprobar o rechazar solicitudes de adoptantes interesados. De igual manera en el apartado de localizar mascotas, cambiar el estado de la mascota (encontrado). Los administradores tienen acceso completo a todas las funcionalidades del sistema.

2.5.2 Usuario normal

Son adoptantes o miembros de la comunidad que se registran en el sistema ya sea para publicar una mascota extraviada o localizar su mascota perdida. Sus responsabilidades y acciones son explorar mascotas disponibles, utilizando filtros para encontrar la mascota de su preferencia, enviar solicitudes de adopción, publicar reporte de mascotas perdidas, subir información detallada de la mascota perdida (fotos, características, ultima ubicación).

2.6 Especificación de requisitos del Software

La especificación de requisitos del Software para el sistema Web de adopción de mascotas para el centro de rescate temporal fauna urbana Ibarra, se basa en la recaudación a través de entrevistas, observaciones directas y análisis de procesos. Estos requisitos se fraccionan en las siguientes categorías.

2.6.1 Historias de usuario

Desde la tabla 1 hasta la tabla 15, se presenta las historias de usuario (requisitos funcionales) del sistema.

Tabla 1

Registro de usuarios en el sistema

Identificación del requerimiento: HU001
Nombre: Registro de usuarios en el sistema
Características: Permite que los usuarios (ciudadanos y administradores) se registren en el sistema para poder interactuar con la plataforma de adopción.
Responsable: Jairo Hidalgo
Descripción: Los usuarios podrán registrarse a través de un formulario para iniciar sesión y empezar a utilizar el sistema de adopción de mascotas.
Prioridad del requisito: 100
Duración: 10 horas
Criterio de aceptación: Los usuarios deben poder completar el registro correctamente y recibir un mensaje de confirmación.

Tabla 2

Inicio y cierre de sesión

Identificación del requerimiento: HU002
Nombre: Inicio y cierre de sesión
Características: Permite a los usuarios iniciar y cerrar sesión en la plataforma utilizando su email y contraseña.
Responsable: Andy Usiña
Descripción: Los ciudadanos y administradores podrán acceder al sistema ingresando su email y contraseña, y podrán cerrar sesión cuando lo deseen.
Prioridad del requisito: 100
Duración: 10 horas
Criterio de aceptación: Los usuarios deben poder iniciar y cerrar sesión sin problemas y recibir mensajes de error si los campos están vacíos o incorrectos.

Tabla 3*Visualización de la página principal*

Identificación del requerimiento: HU003
Nombre: Visualización de la página principal
Características: La página principal muestra información resumida sobre el sistema de adopción y las mascotas disponibles.
Responsable: Jairo Hidalgo
Descripción: Los ciudadanos podrán ver la página principal, que incluirá secciones sobre tipos de mascotas, un buscador de mascotas y una sección informativa sobre adopción.
Prioridad del requisito: 80
Duración: 12 horas
Criterio de aceptación: Los usuarios deben ver claramente todas las secciones mencionadas, con información organizada y accesible.

Tabla 4*Visualización de instrucciones de adopción*

Identificación del requerimiento: HU004
Nombre: Visualización de instrucciones de adopción
Características: Los ciudadanos podrán consultar una sección con instrucciones detalladas sobre el proceso de adopción.
Responsable: Andy Usiña
Descripción: Los ciudadanos podrán acceder a una sección de instrucciones con pasos detallados para adoptar una mascota.
Prioridad del requisito: 50
Duración: 8 horas
Criterio de aceptación: La sección debe mostrar instrucciones claras y fáciles de seguir para el usuario.

Tabla 5*Publicación de mascotas en adopción*

Identificación del requerimiento: HU005
Nombre: Publicación de mascotas en adopción
Características: Los ciudadanos podrán ofrecer mascotas para adopción mediante un formulario detallado.
Responsable: Jairo Hidalgo
Descripción: Los ciudadanos podrán publicar detalles sobre una mascota, incluyendo nombre, edad, sexo y tamaño, para que otros usuarios puedan solicitarlas.
Prioridad del requisito: 80
Duración: 15 horas
Criterio de aceptación: Las publicaciones deben incluir toda la información solicitada y ser validadas correctamente.

Tabla 6*Visualización de la lista de mascotas en adopción*

Identificación del requerimiento: HU006
Nombre: Visualización de la lista de mascotas en adopción
Características: Los ciudadanos podrán consultar una lista de mascotas disponibles para adopción, filtrada por tipo de animal.
Responsable: Andy Usiña
Descripción: Los ciudadanos podrán ver una lista de mascotas clasificadas por tipo (perros, gatos, otros), con sus características relevantes.
Prioridad del requisito: 100
Duración: 18 horas
Criterio de aceptación: La lista debe mostrarse correctamente filtrada, con la información detallada de cada mascota.

Tabla 7*Visualización del perfil de la mascota*

Identificación del requerimiento: HU007
Nombre: Visualización del perfil de la mascota
Características: Los ciudadanos podrán acceder al perfil de una mascota específica para solicitar su adopción.
Responsable: Jairo Hidalgo
Descripción: Los ciudadanos podrán ver el perfil completo de una mascota, incluyendo nombre, características, y una pequeña descripción para decidir si quieren adoptarla.
Prioridad del requisito: 70
Duración: 12 horas
Criterio de aceptación: El perfil debe incluir toda la información relevante de la mascota de manera clara y accesible.

Tabla 8*Edición y visualización del perfil de usuario*

Identificación del requerimiento: HU008
Nombre: Edición y visualización del perfil de usuario
Características: Los ciudadanos podrán editar su perfil y ver la información de las publicaciones realizadas.
Responsable: Andy Usiña
Descripción: Los ciudadanos podrán modificar su información personal (nombre, correo, teléfono) y consultar las publicaciones que han realizado.
Prioridad del requisito: 90
Duración: 15 horas
Criterio de aceptación: Los cambios en el perfil deben reflejarse correctamente y las publicaciones deben ser fácilmente accesibles desde el perfil del usuario.

Tabla 9*Solicitud de adopción*

Identificación del requerimiento: HU009
Nombre: Solicitud de adopción
Características: Los ciudadanos podrán completar un formulario para solicitar la adopción de una mascota.
Responsable: Jairo Hidalgo
Descripción: Los ciudadanos podrán seleccionar una mascota y llenar un formulario con detalles sobre su idoneidad para la adopción.
Prioridad del requisito: 70
Duración: 12 horas
Criterio de aceptación: El formulario debe ser validado correctamente y los datos deben almacenarse de manera segura.

Tabla 10*Aprobación de solicitudes de adopción*

Identificación del requerimiento: HU010
Nombre: Aprobación de solicitudes de adopción
Características: Los ciudadanos podrán aprobar o rechazar solicitudes de adopción de sus mascotas.
Responsable: Andy Usiña
Descripción: Los ciudadanos podrán ver todas las solicitudes recibidas para sus mascotas y decidir si las aprueban o rechazan.
Prioridad del requisito: 100
Duración: 15 horas
Criterio de aceptación: El ciudadano debe poder aprobar o rechazar solicitudes y ver la información del solicitante.

Tabla 11*Visualización de sección de adopciones*

Identificación del requerimiento: HU011
Nombre: Visualización de sección de adopciones
Características: Los ciudadanos podrán ver imágenes con las últimas publicaciones de mascotas en adopción.
Responsable: Jairo Hidalgo
Descripción: Los ciudadanos podrán ver imágenes de perros, gatos y otros animales disponibles para adopción.
Prioridad del requisito: 60
Duración: 10 horas
Criterio de aceptación: Deben funcionar correctamente, mostrando imágenes actualizadas de las mascotas.

Tabla 12*Gestión de perfil de mascota*

Identificación del requerimiento: HU012
Nombre: Gestión de perfil de mascota
Características: Los ciudadanos podrán editar o eliminar las publicaciones de sus mascotas.
Responsable: Andy Usiña
Descripción: Los ciudadanos podrán acceder a sus publicaciones para modificar o eliminar la información de la mascota.
Prioridad del requisito: 70
Duración: 12 horas
Criterio de aceptación: Los usuarios deben poder editar o eliminar sus publicaciones sin errores, con confirmación antes de eliminar.

Tabla 13*Gestión de notificaciones*

Identificación del requerimiento: HU013
Nombre: Gestión de notificaciones
Características: Los ciudadanos recibirán notificaciones sobre acciones relevantes en el sistema, como comentarios, solicitudes de adopción y cambios.
Responsable: Jairo Hidalgo
Descripción: Los ciudadanos recibirán notificaciones cuando haya interacción en sus publicaciones, solicitudes de adopción o cuando se les apruebe o rechace una solicitud.
Prioridad del requisito: 100
Duración: 15 horas
Criterio de aceptación: Las notificaciones deben ser claras, contener información relevante y permitir la eliminación si no son necesarias.

Tabla 14*Gestión de perfiles de usuario (Administrador)*

Identificación del requerimiento: HU014
Nombre: Gestión de perfiles de usuario (Administrador)
Características: Los administradores podrán ver y gestionar los perfiles de los ciudadanos.
Responsable: Andy Usiña
Descripción: Los administradores podrán acceder a los perfiles de los ciudadanos para realizar tareas de mantenimiento o seguridad en el sistema.
Prioridad del requisito: 100
Duración: 15 horas
Criterio de aceptación: Los administradores deben poder ver la información completa de cada usuario, incluyendo las publicaciones realizadas.

Tabla 15*Gestión de publicaciones de mascotas (Administrador)*

Identificación del requerimiento: HU015
Nombre: Gestión de publicaciones de mascotas (Administrador)
Características: Los administradores podrán gestionar las publicaciones de mascotas, editando, eliminando o verificando información errónea.
Responsable: Jairo Hidalgo
Descripción: Los administradores podrán revisar las publicaciones de mascotas y eliminar o editar la información cuando sea necesario.
Prioridad del requisito: 100
Duración: 15 horas
Criterio de aceptación: Las mascotas deben aparecer correctamente en el sistema y cualquier modificación o eliminación debe realizarse con confirmación.

2.7 Identificación de requerimientos no funcionales

A continuación, desde la tabla 16 hasta la tabla 21 se presenta los requisitos funcionales del sistema.

Tabla 16*Rendimiento*

Identificación del requerimiento: RNF01
Nombre: Rendimiento.
Características: Tiempos de respuesta rápidos y eficientes.
Descripción: El sistema debe tener tiempos de respuesta rápidos, mantener una alta disponibilidad.

Tabla 17*Usabilidad*

Identificación del requerimiento: RNF02
Nombre: Usabilidad.
Características: Interfaz intuitiva y fácil de usar.
Descripción: El sistema Web debe proporcionar una experiencia de usuario simple, consistente, permitiendo a los usuarios realizar tareas sin complicaciones y con mínima curva de aprendizaje.

Tabla 18*Seguridad*

Identificación del requerimiento: RNF03
Nombre: Seguridad.
Características: Implementar medidas de seguridad.
Descripción: El sistema protege los datos de los usuarios, utilizando encriptación en las contraseñas de los usuarios.

Tabla 19*Compatibilidad*

Identificación del requerimiento: RNF04
Nombre: Compatibilidad.
Características: Funcionalidad multiplataforma.
Descripción: La aplicación Web debe ser completamente responsiva y funcionar correctamente en diferentes navegadores modernos.

Tabla 20*Escalabilidad*

Identificación del requerimiento: RNF05
Nombre: Escalabilidad.
Características: Capacidad de crecimiento del sistema.
Descripción: La arquitectura del sistema debe permitir un crecimiento futuro, soportando la incorporación de nuevas funcionalidades.

Tabla 21*Mantenibilidad*

Identificación del requerimiento: RNF06
Nombre: Mantenibilidad.
Características: Facilidad de mantenimiento y actualización.
Descripción: El código fuente debe estar bien estructurado, documentado y seguir las mejores prácticas de desarrollo para permitir actualizaciones.

2.8 Determinación de roles y responsabilidades

Acorde a la metodología XP a continuación, en la tabla 22, se detallan los roles y las responsabilidades de los integrantes involucrados en el desarrollo del sistema Web.

Tabla 22*Roles y responsabilidades de los integrantes*

Nombre	Rol	Categoría Profesional	Responsabilidad
Claudio Vaca	Administrador	Ingeniero	Proponer información para el desarrollo del sistema y dar aceptación al cumplimiento de los requisitos del Software
Andy Usiña	Desarrollador	Estudiante de Tecnología Superior en Desarrollo de Software	Diseñar, programar y probar el sistema
Jairo Hidalgo	Desarrollador	Estudiante de Tecnología Superior en Desarrollo de Software	Estudiante de Tecnología Superior en Desarrollo de Software
José Luis Ibarra	Líder	Magíster	Responsable de hacer seguimiento al desarrollo del proyecto

Tabla 23*Análisis y refinamiento de las historias de usuario*

PRIORIDAD	NUMERO DE HISTORIAS DE USUARIO	ID DE TAREA	TAREAS	HORAS – ESFUERZO
900	6	T6-1	Registro de usuarios en el sistema	10
		T6-2	Inicio y cierre de sesión	12
		T6-3	Visualización de la página principal	15
		T6-4	Visualización de instrucciones de adopción	10
		T6-5	Publicación de mascotas en adopción	18
750	3	T3-1	Visualización de la lista de mascotas	15
		T3-2	Visualización del perfil de la mascota	12
		T3-3	Edición del perfil de usuario	10
700	4	T4-1	Solicitud de adopción	15
		T4-2	Aprobación de solicitudes de adopción	20
		T4-3	Implementación de proceso de pre-adopción	25
		T4-4	Gestión de notificaciones	18
600	2	T2-1	Gestión de perfil de mascota	12
		T2-2	Gestión de publicaciones de mascotas	18
300	5	T5-1	Visualización de sección de adopciones	10
		T5-2	Gestión de perfiles de usuario	12
250	1	T1-1	Validación de datos de registro	8
		T1-2	Validación de datos de adopción	8

Una vez definido *el product backlog* con la duración y prioridad de las historias de usuario para el sistema Web de adopción de mascotas, se procedió a distribuir las historias de usuario en 6 Sprints con una duración de 2 semanas cada uno. El proyecto será desarrollado por un equipo de dos programadores, Andy Usiña y Jairo Hidalgo, trabajando 5 días a la semana por 6 horas diarias.

Características de la planificación:

- **Número de desarrolladores: 2**

- **Duración del Sprint:** 2 semanas = 10 días
- **Días de trabajo por semana:** 5 días
- **Horas de trabajo diarias:** 6 horas
- **Velocidad de desarrollo por Sprint:** 120 horas (60 horas × 2 programadores)

Tabla 24*Distribución de Sprints*

SPRINT	TAREA
1	T6-1, T6-2, T6-3
2	T6-4, T6-5, T3-1
3	T3-2, T3-3, T3-4, T4-1
4	T4-2, T4-3, T4-4, T4-5
5	T2-1, T2-2, T2-3, T2-4
6	T5-1, T5-2, T1-1, T1-2, T4-5

Después de completar los Sprints en cada iteración, a lo largo del proceso, se trabaja para incorporar mejoras y abordar los requisitos específicos del proyecto, asegurándose de que cada componente está diseñado de manera óptima para cumplir los objetivos de cada sprint.

2.9 Iteraciones

El desarrollo del sistema Web de adopción y recuperación de mascotas extraviadas se lleva mediante iteraciones siguiendo la metodología ágil *eXtreme Programming (XP)*. Cada iteración incluye diligencias de diseño, codificación, pruebas y entrega parcial de funcionalidades. A continuación, se presenta un resumen de las iteraciones, y su repercusión en el desarrollo del sistema.

2.9.1 Iteración 1: Planificación Inicial

Planificación

En la primera iteración, se priorizan las funcionalidades esenciales para la gestión de usuarios y registro de mascotas disponibles para adopción:

T6-1: Registro de usuarios

T6-2: Inicio y cierre de sesión

T6-3: Visualización de la página principal

Diseño

Se diseñan formularios para el registro de usuarios y para el inicio de sesión, con el objetivo de permitir un acceso fácil y seguro al sistema. Además, se crea una pantalla principal que muestra información sobre la adopción de mascotas, ofreciendo a los usuarios una vista clara de las opciones disponibles. En cuanto a la Base de Datos (BDD), se diseñaron tablas para almacenar los datos de los usuarios y las mascotas disponibles para adopción, asegurando una estructura organizada y accesible. Finalmente, se definieron los controladores de la arquitectura de Software, los cuales gestionan el flujo de datos entre la interfaz de usuario y la Base de Datos, permitiendo una interacción eficiente y fluida entre el sistema y el usuario.

Codificación

Se implementa la lógica *backend* para crear y almacenar recetas electrónicas, asegurando que los datos sean procesados correctamente. Además, se integra la tabla de recetas en la Base de Datos, permitiendo el almacenamiento organizado y accesible de la información relacionada con las prescripciones.

Pruebas

Se realizan pruebas unitarias para verificar la funcionalidad de los formularios de registro y login. Además, se llevan a cabo pruebas de carga para verificar la correcta respuesta del sistema al registrar y autenticar usuarios.

Entrega

Módulo funcional para el registro de usuarios, inicio de sesión y visualización inicial de la página principal.

2.9.2 Iteración 2: Publicación de Mascotas

Planificación

En la segunda iteración, se priorizan las funcionalidades para la publicación de mascotas en adopción:

T6-4: Visualización de instrucciones de adopción

T6-5: Publicación de mascotas en adopción

T3-1: Visualización de la lista de mascotas

Diseño

Se diseña una pantalla para mostrar instrucciones detalladas sobre el proceso de adopción, lo que permite a los usuarios entender claramente los pasos a seguir. Además, se crea un formulario para que los usuarios publiquen detalles sobre las mascotas disponibles para adopción. En la Base de Datos (BDD), se amplía el modelo para incluir la información relacionada con las publicaciones de mascotas, asegurando una correcta relación con los datos de los usuarios.

Codificación

Se implementa la funcionalidad que permite a los usuarios crear nuevas publicaciones de mascotas, incluyendo detalles como nombre, edad, tamaño, sexo y tipo. Además, se desarrolla la lógica *backend* para almacenar las publicaciones de mascotas en la Base de Datos.

Pruebas

Se realizan pruebas unitarias para verificar que las mascotas publicadas aparezcan correctamente en la lista. Además, se validan los formularios de publicación para asegurar que los datos se ingresen correctamente y sin errores.

Entrega

Se entrega un módulo funcional que permite a los usuarios publicar mascotas en adopción y visualizar instrucciones claras para el proceso de adopción.

2.9.3 Iteración 3: Visualización y gestión de perfiles

Planificación

En la tercera iteración, se priorizan las funcionalidades para la visualización del perfil de la mascota y la gestión del perfil del usuario:

T3-2: Visualización del perfil de la mascota

T3-3: Edición del perfil de usuario

Diseño

Se diseña una pantalla para mostrar el perfil completo de cada mascota, incluyendo información relevante como nombre, edad, características y una pequeña descripción. Además, se crea un formulario para que los usuarios puedan editar su perfil, actualizando detalles como la imagen de perfil, nombre, correo y número de contacto.

Codificación

Se implementa la lógica para la visualización de perfiles de mascotas y la edición de perfiles de usuario. La funcionalidad se conecta con la Base de Datos para actualizar la información de manera eficiente.

Pruebas

Se realizan pruebas unitarias para asegurar que los perfiles de las mascotas se visualicen correctamente. Además, se verifica que los cambios en el perfil del usuario se actualicen correctamente en el sistema.

Entrega

Se entrega un módulo que permite a los usuarios visualizar el perfil de la mascota y editar su propio perfil de usuario.

2.9.4 Iteración 4: Solicitud y aprobación de adopción

Planificación

En la cuarta iteración, se priorizan las funcionalidades para solicitar y aprobar adopciones de mascotas:

T4-1: Solicitud de adopción

T4-2: Aprobación de solicitudes de adopción

Diseño

Se crea un formulario para que los usuarios soliciten una mascota en adopción, donde podrán ingresar datos personales y detalles sobre por qué desean adoptar esa mascota. Además, se diseña una pantalla para que los propietarios de las mascotas puedan ver las solicitudes y gestionarlas, aprobándolas o rechazándolas.

Codificación

Se desarrolla el *backend* para manejar la solicitud de adopción y la lógica de aprobación/rechazo de solicitudes. También se implementa la Base de Datos para registrar las solicitudes y las decisiones de adopción.

Pruebas

Se realizan pruebas unitarias para asegurar que los perfiles de las mascotas se visualicen correctamente. Además, se verifica que los cambios en el perfil del usuario se actualicen correctamente en el sistema.

Entrega

Se entrega un módulo funcional que permite solicitar y gestionar adopciones de mascotas, incluyendo la aprobación o rechazo de solicitudes.

2.9.5 Iteración 5: Visualización y gestión de adopciones

Planificación

En la quinta iteración, se priorizan las funcionalidades para visualizar y gestionar las adopciones:

T5-1: Visualización de sección de adopciones

T5-2: Gestión de perfiles de usuario

Diseño

Se diseña una interfaz visual para mostrar las últimas publicaciones de adopciones, organizadas en carruseles de imágenes que faciliten la navegación. Además, se mejora la gestión del perfil de usuario, mostrando las publicaciones de adopción realizadas y permitiendo la edición de los datos personales.

Codificación

Se implementa el *backend* para generar los carruseles de adopciones y la lógica para gestionar las publicaciones y perfiles de usuario. Además, se optimiza la Base de Datos para mejorar la visualización de las adopciones y la interacción con el perfil.

Pruebas

Se realizan pruebas de integración para verificar que la visualización de las adopciones y la gestión del perfil se realicen correctamente. También se valida la actualización del perfil de usuario.

Entrega

Se entrega un módulo completo que permite a los usuarios visualizar y gestionar sus adopciones, así como su perfil personal dentro del sistema.

2.9.6 Iteración 6: Gestión de adopciones y historial de solicitudes

Planificación

En la sexta iteración, se priorizan las funcionalidades clave para gestionar las solicitudes de adopción y mantener un historial de las adopciones realizadas:

T6-1: Solicitud de adopción

T6-2: Historial de adopciones

Diseño

Se diseña un formulario para que los usuarios puedan realizar solicitudes de adopción, capturando detalles como la razón de la solicitud y su información de contacto. Además, se crea una vista para mostrar el historial de adopciones pasadas de cada usuario y mascota,

permitiendo que tanto los adoptantes como los administradores accedan fácilmente a esta información.

Codificación

Se implementa la lógica *backend* para gestionar las solicitudes de adopción y almacenar el historial de adopciones. Se crean los *endpoints* necesarios en la Base de Datos para registrar las solicitudes y el historial, manteniendo la información organizada y accesible.

Pruebas

Se realizan pruebas de validación para garantizar que las solicitudes de adopción se registren correctamente y que el historial de adopciones sea accesible. Se verifica que los datos ingresados por los usuarios en las solicitudes sean válidos y estén correctamente almacenados.

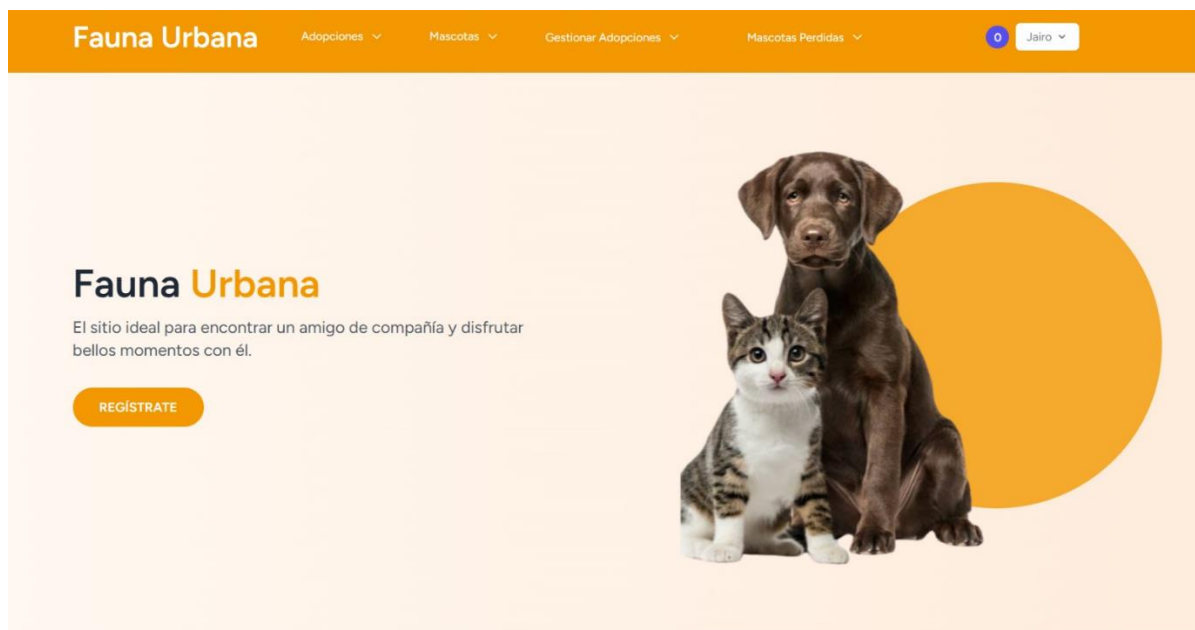
Entrega

Se entrega un módulo funcional que permite a los usuarios realizar solicitudes de adopción y consultar el historial de adopciones de manera fácil y rápida, asegurando que todo el proceso de adopción esté debidamente registrado y gestionado.

2.10 Diseño de la Interfaz Gráfica (GUI)

Figura 1

Página de Inicio



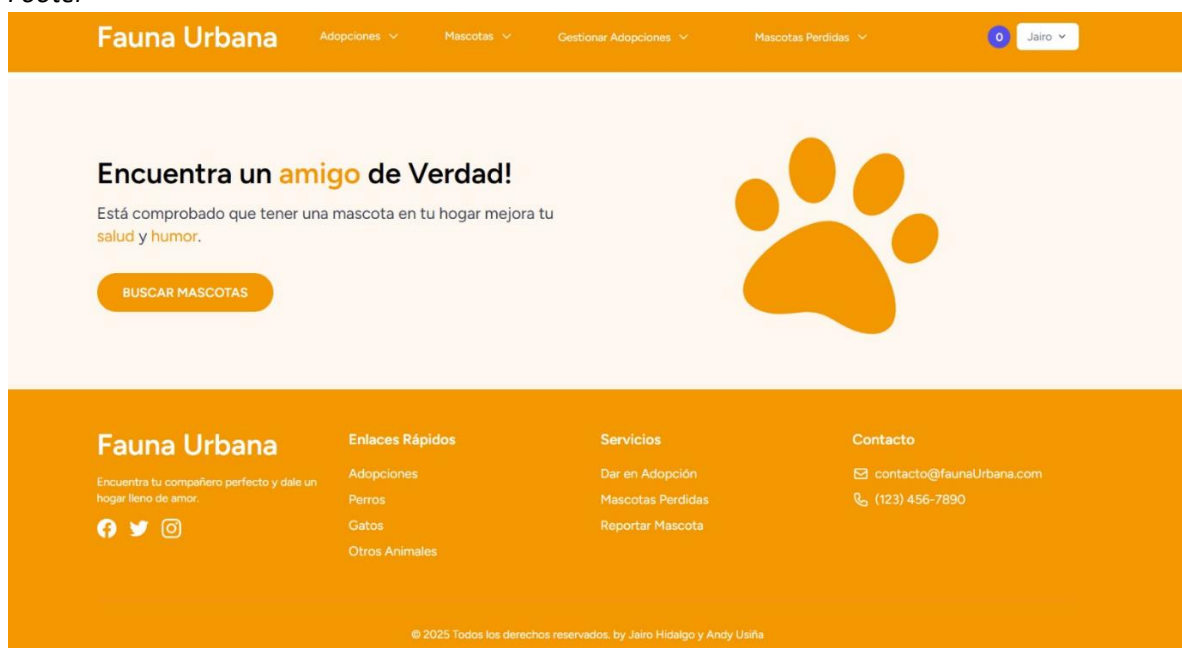
La página de inicio del sistema Web da la bienvenida a los usuarios con una imagen de mascotas del centro de rescate temporal Fauna Urbana Ibarra, representando el objetivo principal del sitio: promover la adopción de mascotas. En la parte superior, se encuentra un menú de navegación.

Figura 2
Área principal



En la figura 2, la sección principal tiene un mensaje destacado que invita a los usuarios a adoptar una mascota, con el texto "¿Quieres adoptar una mascota?" en un tono naranja (similar al del encabezado), lo que atrae la atención del usuario y mantiene la coherencia visual.

Figura 3
Footer



En la figura 3, El pie de página contiene información de derechos de autor y los enlaces a las redes sociales de la plataforma, como Facebook, X e Instagram. Esto ayuda a generar confianza y permite a los usuarios conectarse a la comunidad de Fauna Urbana a través de distintas plataformas.

La interfaz gráfica de la página está cuidadosamente diseñada para ser amigable, accesible y funcional. Los colores cálidos y los mensajes emocionales promueven la adopción de mascotas, mientras que el diseño estructurado facilita la navegación entre las diferentes secciones del sitio. Además, los botones de acción bien destacados y los enlaces rápidos aseguran que los usuarios puedan acceder fácilmente a la información y servicios que necesitan.

2.11 Diseño de la Base de Datos

El diseño de la base de datos fue un pilar esencial para garantizar la integridad y el correcto manejo de la información. Se diseñaron modelos conceptuales, lógicos y físicos para asegurar la adecuada representación y almacenamiento de los datos del sistema.

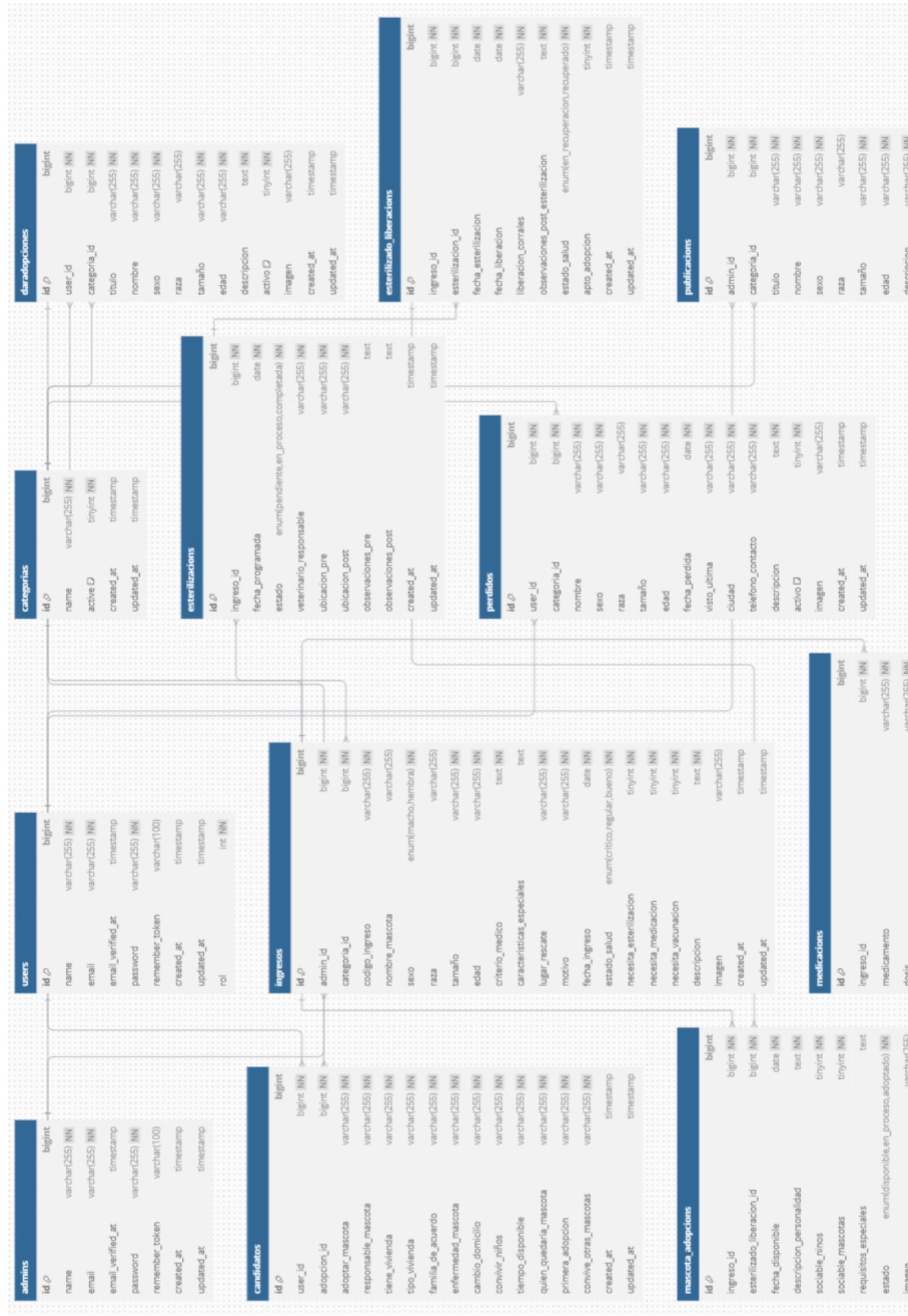
2.11.1 Modelo relacional

Se diseñó una Base de Datos relacional utilizando MySQL, con tablas para usuarios, mascotas, solicitudes adopción, perfiles usuarios, historial adopciones y notificaciones.

2.11.2 Diagrama relacional

Figura 4

Diagrama relacional de la Base de Datos



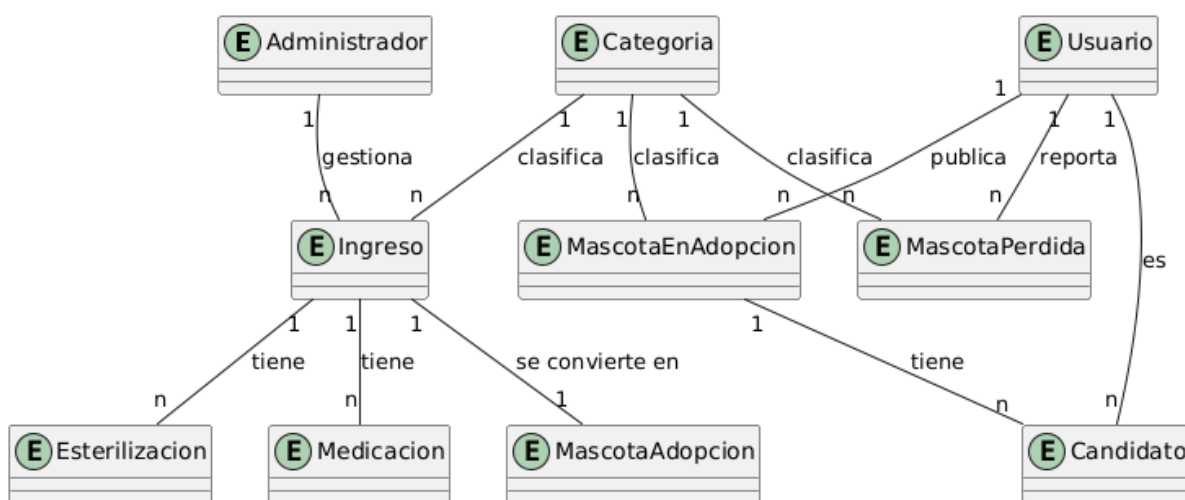
Relaciones Clave

1. **admins - ingresos:** Un administrador puede registrar múltiples ingresos de mascotas.
2. **users y daradopciones:** Un usuario puede publicar múltiples mascotas para adopción.
3. **users y candidatos:** Un usuario puede ser candidato para adoptar múltiples mascotas.
4. **ingresos y esterilizaciones:** Un ingreso puede tener un proceso de esterilización asociado.
5. **ingresos y mascota_adopcions:** Un ingreso puede resultar en una mascota disponible para adopción.
6. **categorias y daradopciones:** Una categoría puede tener múltiples mascotas en adopción.

2.11.3 Modelo conceptual

Figura 5

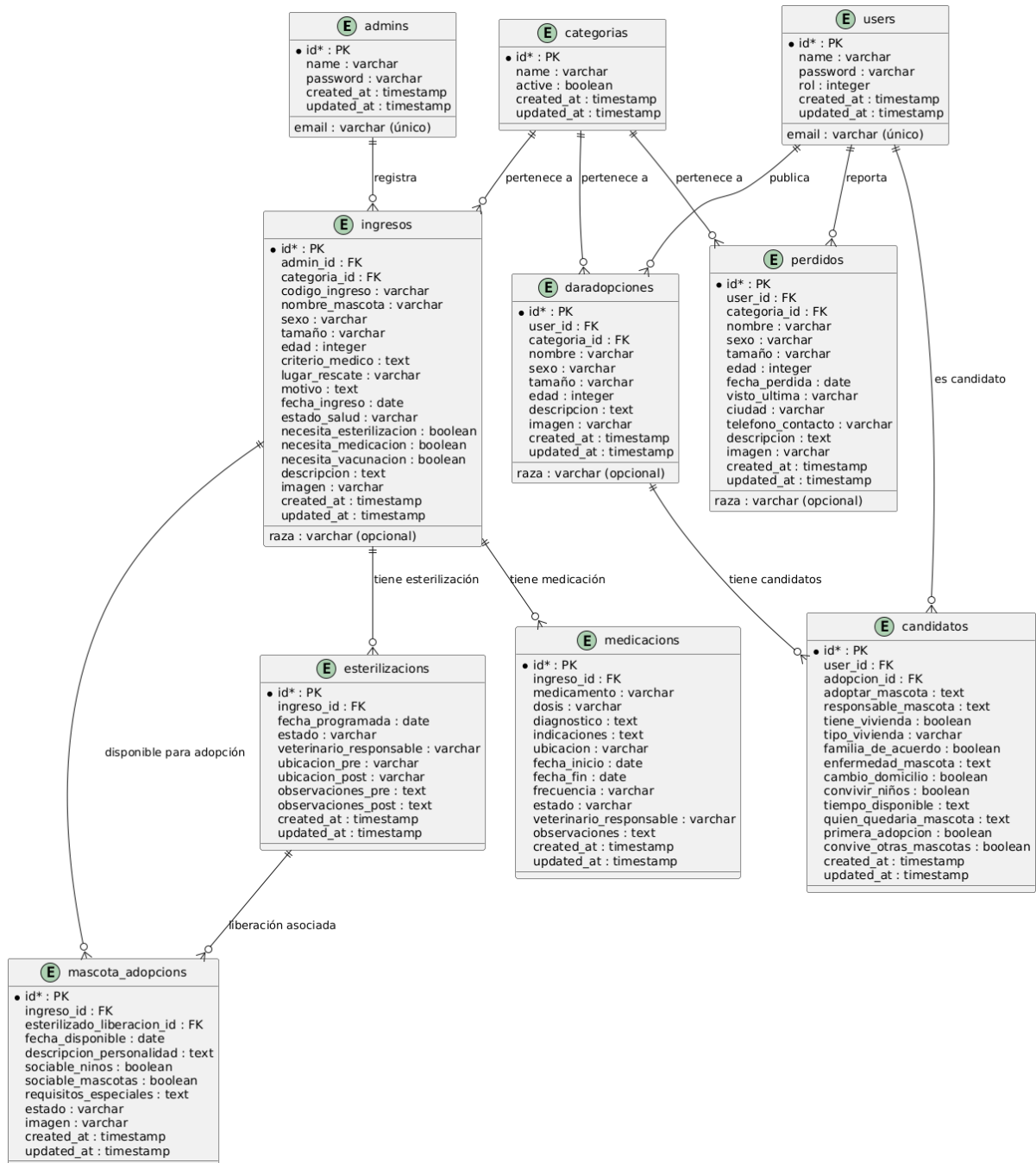
Modelo conceptual fauna urbana



2.11.4 Modeló Lógico

Figura 6

Modelo lógico fauna urbana



2.12 Diseño de la arquitectura de Software

2.12.1 Modelo arquitectónico

Se utiliza el modelo MVC (Modelo-Vista-Controlador) para separar responsabilidades, el modelo se encarga de los datos, consultando a la Base de Datos, actualizaciones, consultas, búsquedas, el controlador como su nombre lo dice controla, recibe las órdenes del usuario el cual se encarga de solicitar los datos al modelo, Las vistas son la representación visual de los datos.

2.12.2 Estilos de capas

Presentación

Interfaz gráfica (HTML, Tailwind CSS, JavaScript). Tailwind CSS es utilizado para facilitar el diseño de una interfaz visual moderna y adaptativa, permitiendo una experiencia fluida en dispositivos móviles y de escritorio.

Lógica de Negocio

Para la lógica de negocio, se utiliza Laravel, un marco de desarrollo PHP que maneja la mayoría de la lógica empresarial del sistema. Laravel es ideal para construir aplicaciones robustas con una arquitectura clara. En la parte del administrador, se implementa Filament, una herramienta que simplifica la construcción de paneles administrativos complejos, permitiendo una interfaz intuitiva para gestionar las mascotas, usuarios y adopciones.

Acceso a Datos

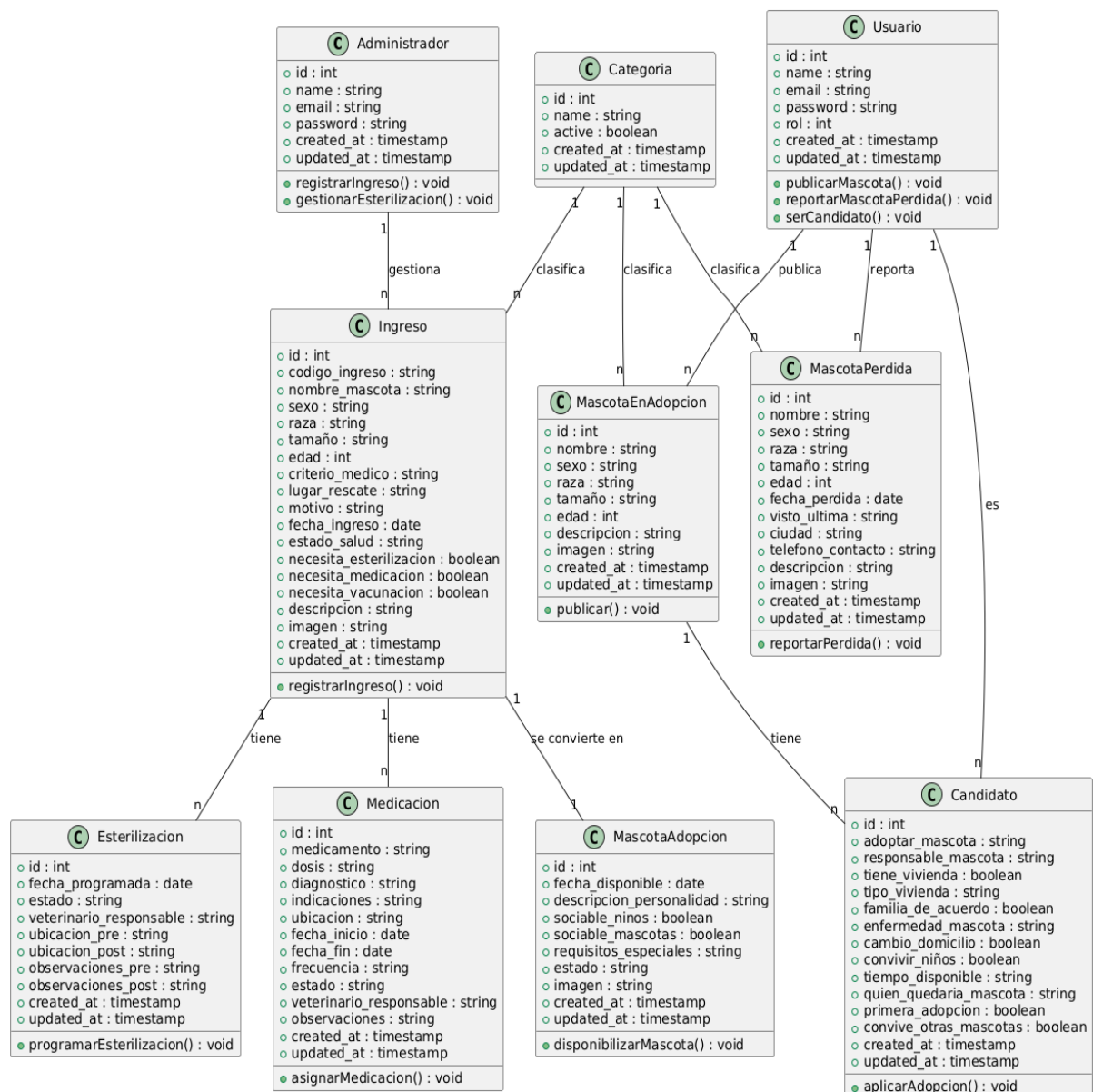
MySQL se utiliza como sistema de gestión de bases de datos para almacenar la información relacionada con usuarios, mascotas, solicitudes de adopción, historial y más.

Laravel facilita la interacción con MySQL a través de migraciones, lo que permite una gestión eficiente de la Base de Datos, asegurando la integridad de los datos.

2.12.3 Diagrama de clases:

Figura 7

Diagrama de clases



2.12.4 Descripción de las clases principales

Clase administrador

Gestiona las operaciones administrativas del sistema, como registrar ingresos de mascotas y supervisar procesos. **Relaciones:** Se relaciona con Ingreso en una relación uno a muchos.

Clase usuario

Representa a los usuarios del sistema, que pueden ser adoptantes o personas que publican mascotas para adopción. **Relaciones:** Se relaciona con MascotaEnAdopcion, Candidato y MascotaPerdida en relaciones uno a muchos.

Clase Categoría

Clasifica las mascotas según su tipo (por ejemplo, perros, gatos, aves). **Relaciones:**

Se relaciona con MascotaEnAdopcion, Ingreso y MascotaPerdida en relaciones uno a muchos.

Clase MascotaEnAdopcion

Representa una mascota disponible para adopción, publicada por un usuario.

Relaciones: Se relaciona con Candidato en una relación uno a muchos.

Clase Candidato

Representa a un usuario interesado en adoptar una mascota, con información sobre su idoneidad. **Relaciones:** Se relaciona con MascotaEnAdopcion en una relación muchos a uno.

Clase Ingreso

Representa una mascota que ingresa al sistema, como una mascota rescatada.

Relaciones: Se relaciona con Esterilizacion, Medicacion y MascotaAdopcion en relaciones uno a muchos.

Clase Esterilizacion

Gestiona el proceso de esterilización de una mascota. **Relaciones:** Se relaciona con Ingreso en una relación muchos a uno.

Clase Medicacion

Maneja los tratamientos médicos asociados a una mascota. **Relaciones:** Se relaciona con Ingreso en una relación muchos a uno.

Clase MascotaAdopcion

Representa una mascota disponible para adopción después de pasar por los procesos necesarios. **Relaciones:** Se relaciona con Ingreso en una relación uno a uno.

Clase MascotaPerdida

Representa una mascota reportada como perdida por un usuario. **Relaciones:** Se relaciona con Usuario en una relación muchos a uno.

2.13 Diseños de casos de prueba

Pruebas Funcionales

Se realizan pruebas funcionales para validar cada módulo del sistema Web de adopción de mascotas.

Tabla 25

Casos de Prueba

N°	Caso de Prueba	Parámetros de Entrada	Resultado Esperado
01	Registro de usuario	Nombre, Correo, Teléfono, Dirección, Contraseña.	El usuario debe ser registrado correctamente, con un mensaje de confirmación.
02	Inicio de sesión	Correo, Contraseña.	El usuario debe ser redirigido a la página principal y poder acceder a su perfil.
03	Cierre de sesión	N/A (usuario ya ha iniciado sesión).	El usuario debe ser desconectado y redirigido a la página de inicio.
04	Publicación de Mascotas	Nombre de la mascota, Edad, Sexo, Tamaño, Tipo, Descripción.	La mascota debe ser publicada y visible en la lista de adopciones.
05	Visualización de Mascotas	Filtro por tipo de mascota (perro, gato, otros).	La lista debe mostrar correctamente las mascotas filtradas según el tipo seleccionado.
06	Solicitud de Adopción	Seleccionar una mascota, Razón para adoptar.	La solicitud debe ser enviada correctamente y reflejada en la Base de Datos.
07	Aprobación de Solicitudes	Solicitud de adopción recibida.	El dueño debe poder aprobar o rechazar la solicitud, y el estado de la solicitud debe actualizarse.
08	Edición del Perfil de Usuario	Nombre, Correo, Teléfono.	El perfil debe ser actualizado correctamente, y la información debe reflejarse en el sistema.
09	Visualización del Perfil de Mascota	Mascota seleccionada de la lista.	La página de perfil debe mostrar correctamente la información de la mascota seleccionada.
10	Notificación de Solicitud	Solicitud de adopción realizada y su estado actualizado.	El usuario debe recibir una notificación de acuerdo con el estado de la solicitud (aprobada o rechazada).
11	Eliminación de Publicación de Mascota	Mascota publicada previamente.	La publicación de la mascota debe ser eliminada correctamente de la lista de adopciones.

3 CAPITULO III

RESULTADOS Y DISCUSIONES

En este capítulo se muestran los resultados obtenidos tras el desarrollo del sistema Web de gestión para adopción y recuperación de mascotas extraviadas para la el centro temporal de rescate Fauna Urbana Ibarra, a continuación, se describe las principales interfaces y el funcionamiento del sistema.

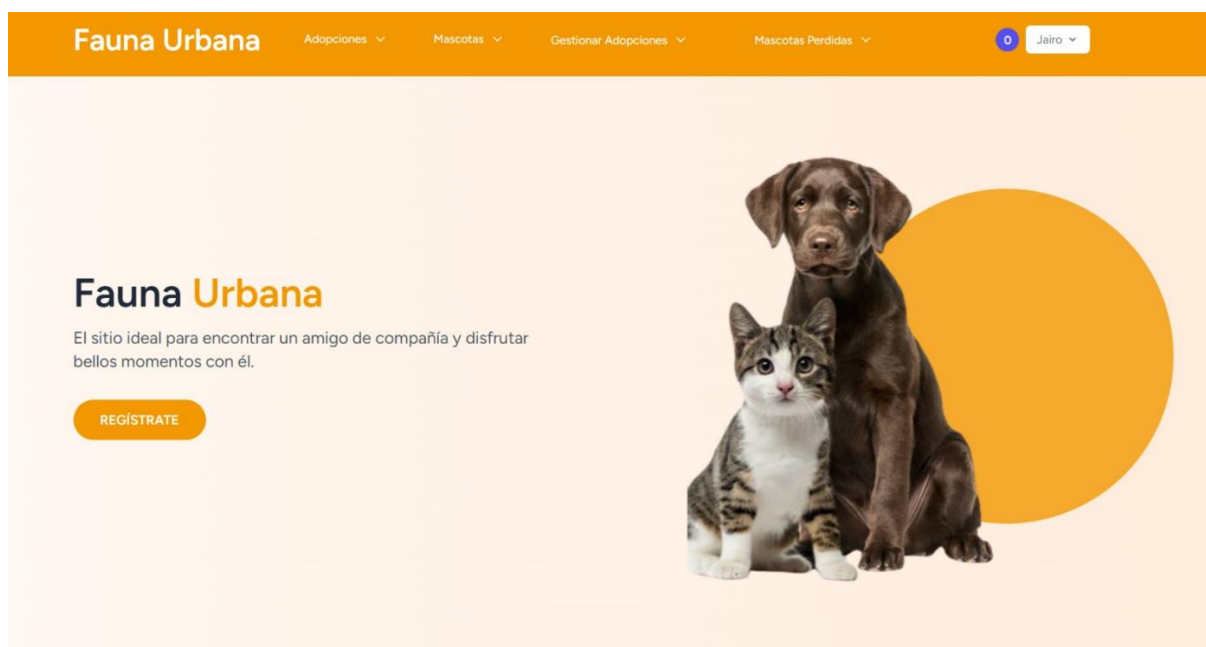
3.1 Resultados de la implementación del sistema

El sistema Web se compone por los siguientes módulos principales, diseñados para atender las necesidades del centro temporal de rescate Fauna Urbana Ibarra.

3.1.1 Módulo de interfaz principal

Figura 8

Módulo de interfaz principal



El Módulo de Interfaz Principal es la puerta de acceso del usuario al sistema como se muestra en la **Figura 8**, donde se visualizan las primeras opciones para interactuar con las funcionalidades clave del sitio. Esta interfaz ha sido diseñada para ser intuitiva y fácil de usar,

con un enfoque en la experiencia del usuario, lo que permite acceder rápidamente a la información sobre las mascotas disponibles para adopción y las mascotas perdidas.

3.1.2 Módulo de login

Figura 9

Interfaz de login



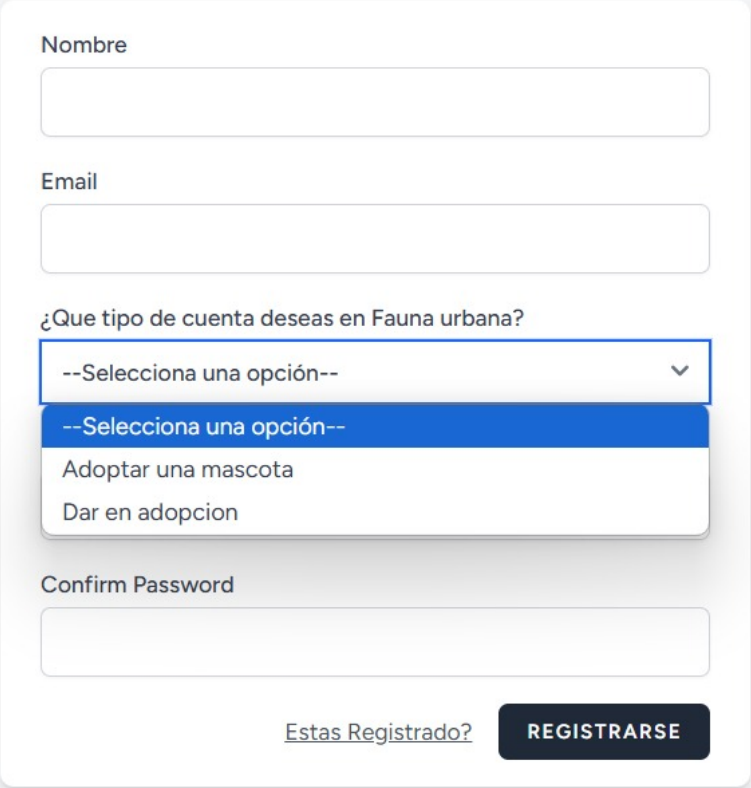
The image shows a login form for 'Fauna Urbana'. At the top, it says 'Fauna Urbana' and 'Entre a su cuenta'. Below that is a link 'Abrir una cuenta'. The form has two input fields: 'Correo electrónico*' and 'Contraseña*'. The password field has a toggle icon for visibility. There is a 'Recordarme' checkbox and an 'Entrar' button.

La **Figura 9**: Interfaz de Login corresponde a la pantalla de inicio de sesión del sistema Fauna Urbana, esta interfaz es crucial para permitir que los usuarios accedan a sus cuentas, y de esta manera tenga acceso al catálogo de mascotas disponibles, o en tal caso que esté buscando a su mascota perdida.

3.1.3 Módulo de registro

Figura 10

Interfaz de registro



The image shows a registration form with the following elements:

- Nombre**: A text input field.
- Email**: A text input field.
- ¿Que tipo de cuenta deseas en Fauna urbana?**: A dropdown menu with the following options:
 - Selecciona una opción-- (highlighted in blue)
 - Adoptar una mascota
 - Dar en adopcion
- Confirm Password**: A text input field.
- Estas Registrado?**: A link.
- REGISTRARSE**: A dark blue button.

En la **Figura 10** se muestra el formulario que se tiene que llenar cuando se está creando su cuenta para formar parte del sistema Web de adopción y localización de mascotas, esta consta de lo siguiente: nombre, correo electrónico, tipo de cuenta que desea en fauna urbana (Adoptar una mascota o Dar en adopción) contraseña y repetir contraseña. De esta manera, se asegura que el usuario ingrese datos básicos de identificación y protección de la cuenta, facilitando el proceso de registro y garantizando la seguridad de la información proporcionada.

3.1.4 Módulo de mascotas

Figura 11

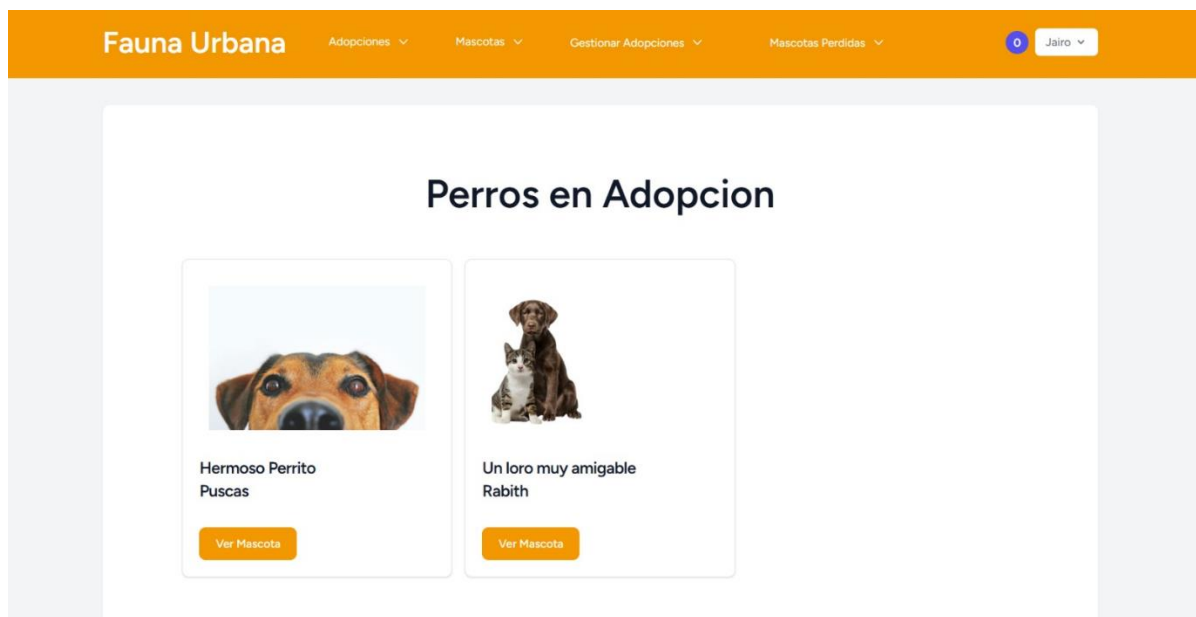
Interfaz de selección de mascota



En la **Figura 11** se muestra el apartado de selección en el que el usuario selecciona el tipo de mascota que desea adoptar. Dependiendo de la elección realizada, el sistema habilita las pestañas correspondientes, ofreciendo opciones específicas para cada categoría de animal (por ejemplo, perros, gatos u otros). De esta manera, el usuario puede explorar de forma sencilla las distintas opciones de adopción, adaptando su búsqueda a sus preferencias particulares.

Figura 12

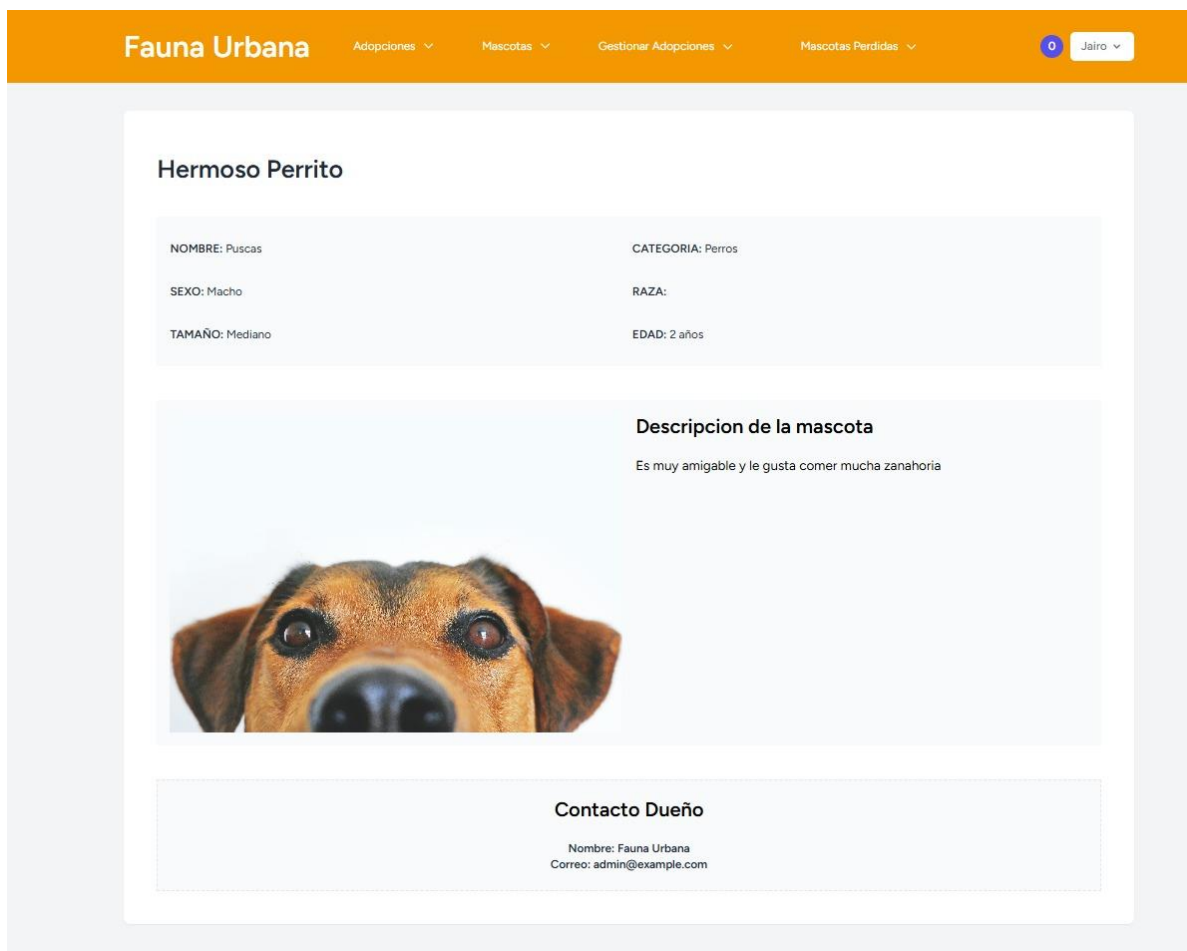
Interfaz de catálogo de perros



La **Figura 12** constituye uno de los componentes centrales de la plataforma Fauna Urbana, ya que permite a los usuarios visualizar de manera ordenada y atractiva a los caninos disponibles para adopción. En esta sección, el usuario encuentra una lista de perros, cada uno acompañado de su nombre, edad, fotografía y breve descripción, lo cual facilita la comparación y la toma de decisiones.

Figura 13

Interfaz de ver mascota



En la **Figura 14**, la interfaz muestra información general sobre las mascotas disponibles para adopción. Se presenta el nombre de la mascota, su sexo, tamaño, categoría (por ejemplo, perros o gatos), raza (si está disponible) y edad. Además, se incluye una descripción de la mascota que detalla su personalidad, comportamiento y preferencias, lo que ayuda a los usuarios a conocerla mejor. En la parte inferior de la interfaz, se proporciona la información de contacto del dueño o responsable, que puede incluir el nombre de la organización o persona a cargo y un correo electrónico. La interfaz también cuenta con un botón de "Contactar al dueño" que redirige al usuario a WhatsApp del dueño o responsable, facilitando la comunicación directa y rápida para obtener más detalles o iniciar el proceso de adopción.

Figura 14

Interfaz de catálogo de gatos

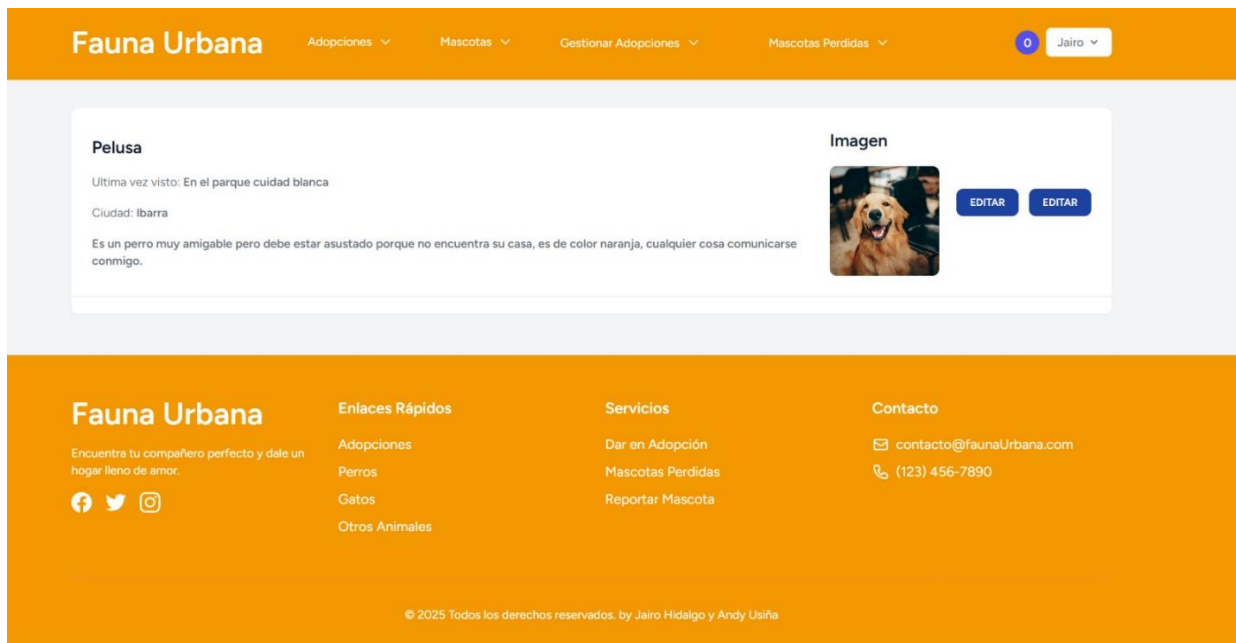


Constituye un módulo esencial del sistema Fauna Urbana, ya que facilita la exploración y adopción de felinos. En esta sección, el usuario encuentra información detallada sobre cada gato, como su nombre, edad, descripción y fotografía, permitiendo comparar distintas opciones de adopción de manera ágil, promoviendo así una adopción responsable y satisfactoria como se muestra en la **Figura 15**.

3.1.5 Módulo de localización de mascotas

Figura 15

Interfaz de mascotas perdidas



Esta vista corresponde a la sección de mascotas perdidas, donde se muestra un catálogo de mascotas que han sido reportadas como extraviadas por parte de los usuarios de la comunidad. En esta sección, los dueños o personas que han encontrado una mascota perdida pueden publicar información detallada.

Figura 16*Formulario publicar mascota perdida*

The image shows a web form titled "Publicar Pérdida de Mascota" on the "Fauna Urbana" website. The form is set against a white background with a light gray border. At the top, there is an orange navigation bar with the site name "Fauna Urbana" and several menu items: "Adopciones", "Mascotas", "Gestionar Adopciones", and "Mascotas Perdidas". A user profile icon for "Jairo" is visible in the top right corner. The form itself contains the following fields:

- Nombre:** A text input field with the placeholder "Ingresa el nombre de la mascota".
- Categoría:** A dropdown menu with "--Selecciona--" and a downward arrow.
- Sexo:** A dropdown menu with "--Selecciona--" and a downward arrow.
- Raza:** A text input field with the placeholder "Ingresa la raza de la mascota".
- Tamaño:** A dropdown menu with "--Selecciona--" and a downward arrow.
- Edad:** A text input field with the placeholder "Ingresa la edad de la mascota en meses y años eje. 1 año 4 meses".
- Fecha de Pérdida:** A date input field with the placeholder "dd/mm/aaaa" and a calendar icon.
- Última vez visto en:** A text input field with the placeholder "Lugar donde se vio por última vez".
- Ciudad:** A text input field with the placeholder "Ciudad donde se perdió".
- Teléfono de contacto:** A text input field with the placeholder "Teléfono para contactarte".
- Descripción:** A large text area for a detailed description of the pet.
- Imagen:** A dashed box with a plus sign and a camera icon, indicating where to upload a representative image.

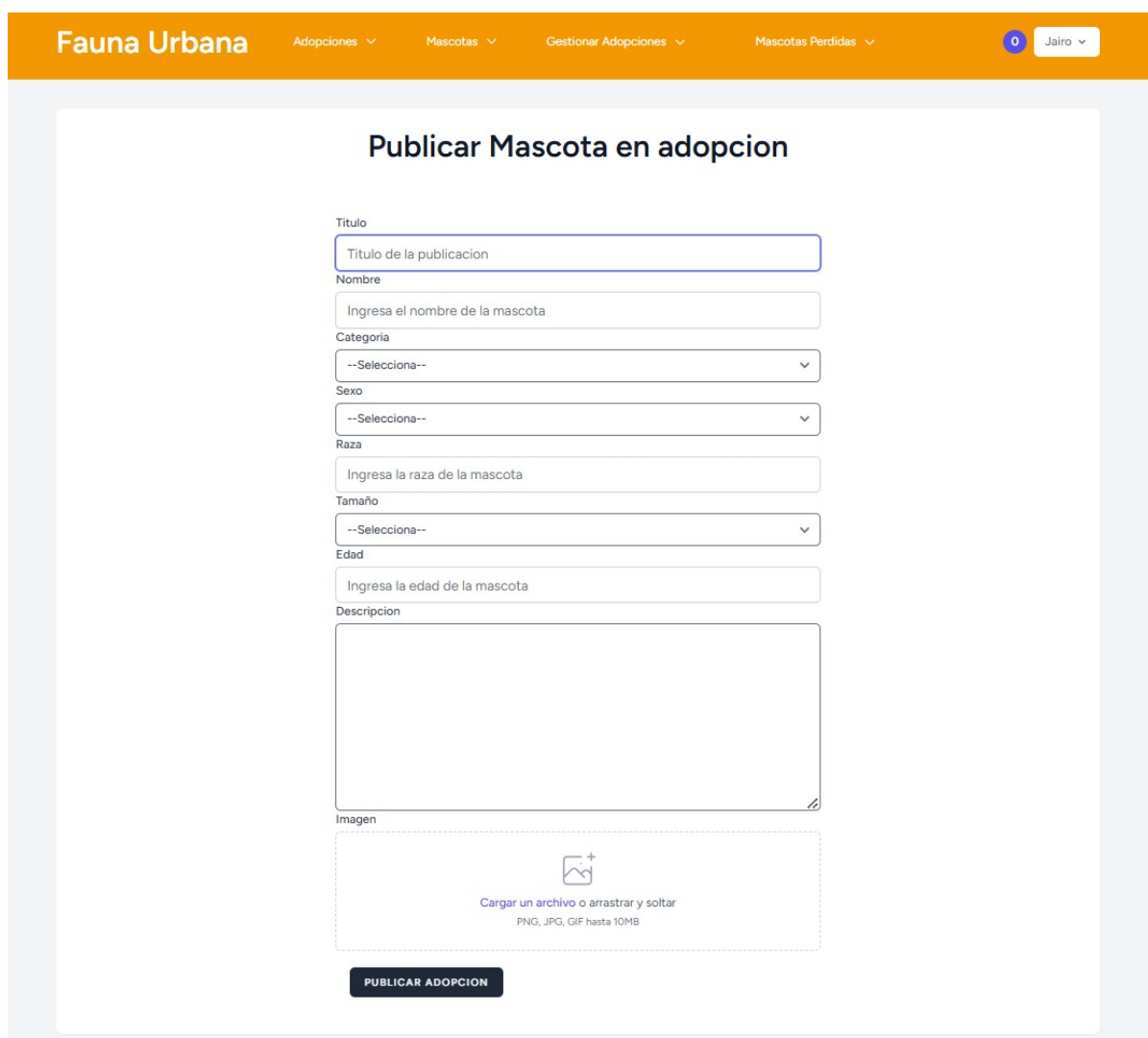
El formulario que se muestra en la Figura 15 permite al usuario registrar todos los datos necesarios para reportar una mascota perdida. En esta sección, el usuario ingresa el nombre de la mascota, selecciona la categoría (perro, gato u otra), define el sexo, la raza, el tamaño y la edad, e indica la fecha en que se perdió. Además, el usuario proporciona detalles sobre el lugar donde se vio por última vez a la mascota, la ciudad donde ocurrió la pérdida y un número de teléfono de contacto para que las personas puedan comunicarse en caso de tener información. También se incluye un campo para una descripción detallada de la mascota, lo que ayuda a identificarla mejor, y la opción de adjuntar una imagen representativa. Con toda esta

información, el sistema almacena los detalles y habilita la publicación, de modo que la mascota perdida quede visible en la plataforma para que la comunidad pueda colaborar en su búsqueda.

3.1.6 Módulo crear adopción

Figura 17

Formulario publicar mascota en adopción



The screenshot shows a web interface for 'Publicar Mascota en adopción' (Publish Pet for Adoption) within the 'Fauna Urbana' system. The header is orange and contains the site name and navigation links: 'Adopciones', 'Mascotas', 'Gestionar Adopciones', and 'Mascotas Perdidas'. A user profile 'Jairo' is visible in the top right. The form itself is white and contains the following fields:

- Título:** A text input field with the placeholder 'Titulo de la publicacion'.
- Nombre:** A text input field with the placeholder 'Ingresa el nombre de la mascota'.
- Categoría:** A dropdown menu with the placeholder '--Selecciona--'.
- Sexo:** A dropdown menu with the placeholder '--Selecciona--'.
- Raza:** A text input field with the placeholder 'Ingresa la raza de la mascota'.
- Tamaño:** A dropdown menu with the placeholder '--Selecciona--'.
- Edad:** A text input field with the placeholder 'Ingresa la edad de la mascota'.
- Descripción:** A large text area for entering details.
- Imagen:** A dashed box containing an upload icon and the text 'Cargar un archivo o arrastrar y soltar' and 'PNG, JPG, GIF hasta 10MB'.

At the bottom of the form is a dark blue button labeled 'PUBLICAR ADOPCION'.

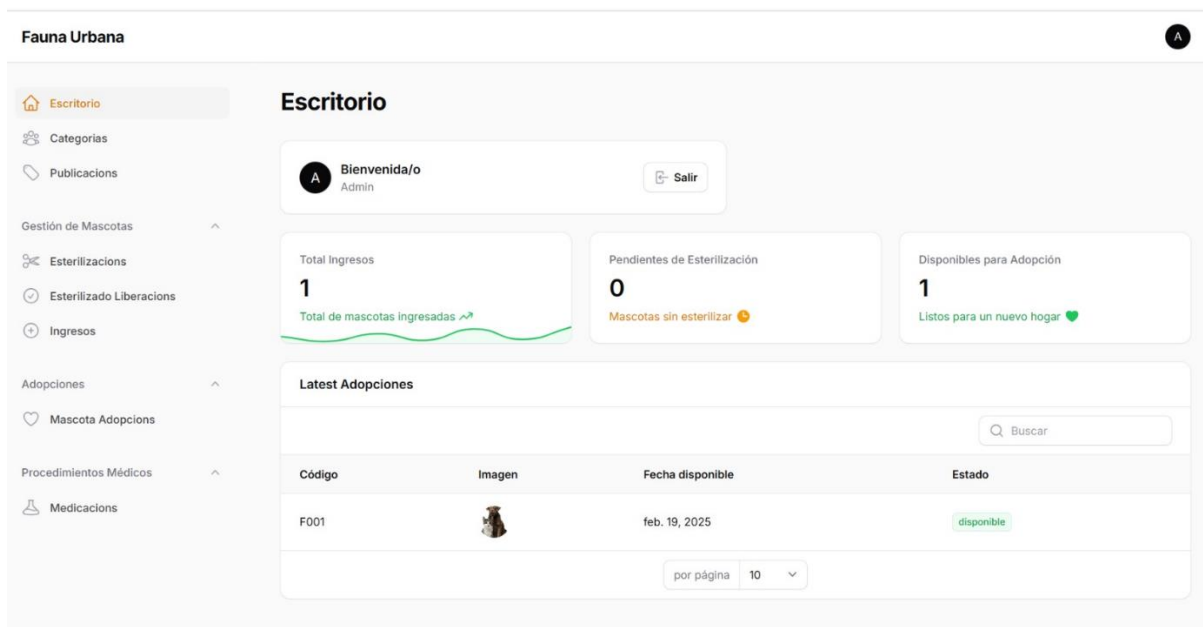
El formulario que se muestra en la **Figura 15** permite al administrador registrar todos los datos necesarios para publicar una mascota en adopción. En esta sección, el administrador ingresa el título y el nombre de la mascota, selecciona la categoría (perro, gato u otra), define el sexo, la raza, el tamaño y la edad, y proporciona una breve descripción que ayude a los

usuarios a conocer mejor al animal. Además, el administrador adjunta una imagen representativa para que la mascota sea fácilmente reconocible. Con toda esta información, el sistema almacena los detalles y habilita la publicación, de modo que la mascota quede disponible para ser adoptada a través de la plataforma.

3.1.7 Módulo de administrador

Figura 18

Interfaz de Administrador



En la **Figura 19** ofrece un panel de control completo para gestionar el sistema Fauna Urbana. En la parte superior se muestra el nombre del usuario administrador y la opción de Salir, facilitando la identificación y la salida segura de la sesión. A la izquierda, se encuentra un menú lateral con diversas secciones, entre ellas:

1. **Escritorio:** Resumen general del sistema, donde se presentan indicadores rápidos de estado, como:
2. **Total, Ingresos:** Número de mascotas registradas.

3. **Pendientes de Esterilizar:** Cantidad de mascotas que requieren esterilización.
4. **Disponibles para Adopción:** Mascotas listas para ser adoptadas.
5. **Categorías y Publicaciones:** Opciones para clasificar y revisar las publicaciones que realizan todos los usuarios.
6. **Sección de Mascotas:** Con subapartados para gestionar el estado de cada mascota (esterilizadas, extraviadas o ingresadas).
7. **Adopciones:** Visualización y administración de las solicitudes y procesos de adopción.

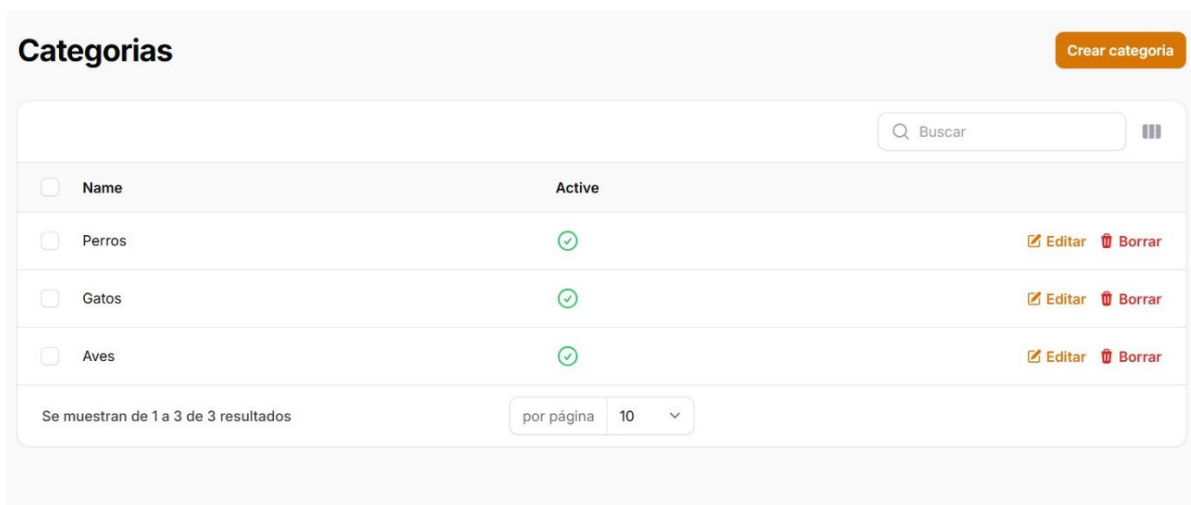
En el centro de la pantalla, se observa un panel principal que muestra: Una bienvenida personalizada para el administrador, un cuadro de estadísticas que reflejan datos clave, como el total de mascotas ingresadas, cuántas están pendientes de esterilizar, cuántas están disponibles para adopción, y las últimas adopciones realizadas.

Un listado de las adopciones más recientes, con columnas que incluyen código, imagen, fecha disponible y estado de la mascota, además de un campo de búsqueda para filtrar rápidamente la información.

Esta interfaz está diseñada para que el administrador controle todas las publicaciones y pueda gestionar de forma ágil la información de las mascotas (ingresos, extravíos, adopciones), asegurando que el sistema se mantenga actualizado y organizado.

Figura 19

Administrador - agregar categorías



Se muestra la interfaz de gestión de categorías, donde el administrador del sistema, en este caso Fauna Urbana, tiene la capacidad de agregar, editar y eliminar categorías relacionadas con las mascotas. Esta sección permite al administrador organizar y clasificar las mascotas en grupos específicos, como Perros, Gatos y Aves, lo que facilita la navegación y búsqueda de los usuarios dentro de la plataforma. Cada categoría cuenta con opciones para editar o borrar, lo que brinda flexibilidad al administrador para mantener actualizada la lista de categorías según las necesidades del sistema. Además, la interfaz muestra un indicador de estado (representado por un círculo) que permite visualizar si una categoría está activa o no.

Figura 20

Administrador - ingreso de nuevas mascotas

Ingresos > Listado

Ingresos

Crear ingreso

Buscar

<input type="checkbox"/>	Codigo ingreso	Categoría	Nombre mascota	Estado salud	Necesita esterilizacion	Necesita medicacion	Necesita vacunacion	Imagen
<input type="checkbox"/>	F001	Perros	Tingo	regular	✓	✗	✗	
<input type="checkbox"/>	F002	Perros	Andy	bueno	✓	✗	✗	

Se muestran de 1 a 2 de 2 resultados

por página 10

Esta sección permite al administrador registrar y monitorear el estado de las mascotas que ingresan al sistema, lo que facilita la gestión de sus necesidades médicas y de cuidado. Además, proporciona una visión general de los ingresos, lo que ayuda a mantener un control organizado y eficiente de las mascotas rescatadas o en proceso de adopción.

Figura 21

Administrador - formulario de ingreso nuevas mascotas

Ingresos > Crear

Crear Ingreso

Admin*

Categoría*

Código de Ingreso*

Nombre mascota

Sexo*

Raza

Tamaño*

Edad Aproximada (meses)*

Criterio medico*

Características especiales

Lugar rescate*

Motivo*

Fecha ingreso*

Estado salud*

¿Necesita Esterilización?

¿Necesita Vacunación?

¿Necesita Medicación?

Descripción*

Imagen

En la interfaz de Crear Ingreso, el administrador debe completar varios campos para registrar una mascota que ingresa al sistema. A continuación, se describen los campos que se deben llenar:

1. **Admin:** Se selecciona al administrador responsable del registro del ingreso.
2. **Código de Ingreso:** Se ingresa un identificador único para el ingreso de la mascota.

3. **Sexo:** Se selecciona el sexo de la mascota (macho o hembra).
4. **Tamaño:** Se elige el tamaño de la mascota (pequeño, mediano o grande).
5. **Criterio Médico:** Se describe cualquier condición médica o criterio de salud relevante para la mascota.
6. **Lugar de Rescate:** Se indica el lugar donde fue rescatada la mascota.
7. **Motivo:** Se especifica el motivo por el cual la mascota ingresó al sistema (por ejemplo, rescate, abandono, etc.).
8. **Fecha de Ingreso:** Se ingresa la fecha en que la mascota ingresó al sistema, en formato dd/mm/aaaa.
9. **Estado de Salud:** Se selecciona el estado de salud de la mascota (por ejemplo, crítico, regular, bueno).
10. **Necesita Esterilización:** Se indica si la mascota requiere esterilización.
11. **Necesita Medicación:** Se especifica si la mascota necesita tratamiento médico.
12. **Necesita Vacunación:** Se determina si la mascota requiere vacunación.
13. **Descripción:** Se proporciona una descripción detallada de la mascota, incluyendo su comportamiento, antecedentes o cualquier información adicional relevante.
14. **Imagen:** Se adjunta una imagen de la mascota, ya sea arrastrando y soltando el archivo o seleccionándolo desde el dispositivo.

Finalmente, el administrador tiene las opciones de Crear para guardar el registro, Crear y crear otro para guardar y continuar registrando más ingresos, o Cancelar para descartar el proceso. Estos campos permiten registrar de manera completa y organizada la información de

cada mascota que ingresa al sistema, asegurando que se cubran sus necesidades médicas y de cuidado.

Figura 22

Administrador - esterilizaciones

<input type="checkbox"/>	Código Ingreso	Fecha programada	Estado	Veterinario responsable	
<input type="checkbox"/>	F001	feb. 18, 2025	en_proceso	Andy Usiña	Editar Borrar Ver

Se muestra un resultado

por página 10

Se muestra la interfaz de Esterilizaciones, donde el administrador del sistema puede gestionar y visualizar los procesos de esterilización programados para las mascotas. Esta sección está diseñada para registrar y monitorear las esterilizaciones, asegurando que cada mascota reciba el procedimiento necesario.

Figura 23

Administrador - formulario esterilizaciones

Esterilizaciones > Crear

Crear Esterilizacion

Código Ingreso*
Seleccione una opción

Fecha programada*
dd/mm/aaaa

Estado*
Pendiente

Veterinario responsable*

Ubicacion pre*

Ubicacion post*

Observaciones pre

Observaciones post

Crear **Crear y crear otro** **Cancelar**

1. **Código de Ingreso:** Se selecciona el código único asociado al ingreso de la mascota que será esterilizada.
2. **Estado:** Se indica el estado actual del proceso de esterilización.
3. **Ubicación Pre:** Se especifica la ubicación donde se encuentra la mascota antes del procedimiento de esterilización.
4. **Observaciones Pre:** Se ingresan observaciones relevantes sobre la mascota antes de la esterilización.
5. **Fecha Programada:** Se ingresa la fecha en la que está programada la esterilización, en formato dd/mm/aaaa.
6. **Veterinario Responsable:** Se indica el nombre del veterinario encargado de realizar el procedimiento.

7. **Ubicación Post:** Se especifica la ubicación donde se encontrará la mascota después de la esterilización.
8. **Observaciones Post:** Se ingresan observaciones relevantes sobre la mascota después de la esterilización.

Opciones de Acción

- **Crear:** Guarda el registro de la esterilización.
- **Crear y crear otro:** Guarda el registro y permite continuar registrando más esterilizaciones.
- **Cancelar:** Descarta el proceso sin guardar los cambios.

Figura 24

Administrador - esterilización liberación

The screenshot shows a web interface for managing sterilized animals. At the top, there is a breadcrumb 'Esterilizado Liberacions > Listado' and a title 'Esterilizado Liberacions'. A search bar is present with the text 'Buscar'. Below the search bar is a table with the following columns: 'Código de Mascota', 'Fecha esterilizacion', 'Fecha liberacion', 'Estado salud', and 'Apto adopcion'. There is a single row of data for a pet with ID 'F001', sterilized on 'feb. 18, 2025', and scheduled for release on 'feb. 23, 2025'. The health status is 'en_recuperacion' (highlighted in yellow) and it is 'Apto adopcion' (marked with a green checkmark). Action buttons for 'Editar' and 'Ver' are visible. At the bottom, it indicates 'Se muestra un resultado' and a pagination control set to 'por página 10'.

<input type="checkbox"/>	Código de Mascota	Fecha esterilizacion	Fecha liberacion	Estado salud	Apto adopcion	
<input type="checkbox"/>	F001	feb. 18, 2025	feb. 23, 2025	en_recuperacion	✓	Editar Ver

El administrador del sistema puede gestionar y visualizar el estado de las mascotas que han sido esterilizadas y están en proceso de ser liberadas para adopción.

Figura 25

Administrador - formulario esterilización liberación

Esterilizado Liberacions > Crear

Crear Esterilizado Liberacion

Código Ingreso*

Fecha esterilización*

Fecha liberación*

Liberación corrales*

Observaciones post esterilización*

Estado salud*

Apto adopcion*

1. **Código de Ingreso:** Se selecciona el código único asociado al ingreso de la mascota que ha sido esterilizada.
2. **Fecha de Esterilización:** Se ingresa la fecha en la que se realizó la esterilización, en formato dd/mm/aaaa.
3. **Fecha de Liberación:** Se especifica la fecha en la que la mascota será liberada, en formato dd/mm/aaaa.
4. **Liberación en Corrales:** Se indica si la mascota será liberada en corrales o en otro lugar específico.
5. **Estado de Salud:** Se selecciona el estado de salud actual de la mascota después de la esterilización.
6. **Observaciones Post Esterilización:** Se ingresan observaciones relevantes sobre la mascota después de la esterilización.
7. **Apto para Adopción:** Se indica si la mascota está apta para ser puesta en adopción.

Opciones de Acción:

- **Crear:** Guarda el registro de la liberación.
- **Crear y crear otro:** Guarda el registro y permite continuar registrando más liberaciones.
- **Cancelar:** Descarta el proceso sin guardar los cambios.

Figura 26*Administrador - medicación*

<input type="checkbox"/>	Código de Mascota	Medicamento	Fecha inicio	Fecha fin	Estado	Frecuencia	
<input type="checkbox"/>	F001	Ibuprofeno	feb. 19, 2025	feb. 27, 2025	activo	diaria	Editar Ver

Se muestra la interfaz de Listado de Medicaciones, donde el administrador del sistema puede registrar y gestionar los medicamentos asignados a las mascotas.

Figura 27

Administrador - formulario medicación

The screenshot shows a web form titled 'Crear Medicación' under the breadcrumb 'Medicaciones > Crear'. The form is organized into two columns. The left column contains fields for 'Mascota' (a dropdown menu), 'Dosis' (a text input), 'Indicaciones' (a large text area), 'Fecha inicio' (a date picker), 'Frecuencia' (a dropdown menu), and 'Veterinario responsable' (a text input). The right column contains fields for 'Medicamento' (a text input), 'Diagnostico' (a text input), 'Ubicacion' (a text input), 'Fecha fin' (a date picker), 'Estado' (a dropdown menu), and 'Observaciones' (a large text area). At the bottom of the form, there are three buttons: 'Crear' (orange), 'Crear y crear otro', and 'Cancelar'.

Esta interfaz permite gestionar de manera eficiente los tratamientos médicos de las mascotas, asegurando que cada medicamento esté correctamente registrado con la información necesaria. Ayuda a mantener un control organizado sobre las medicaciones activas y facilita la supervisión del estado de salud de las mascotas.

1. **Mascota:** Selección de la mascota a la que se le asignará la medicación.
2. **Medicamento:** Nombre del medicamento que se administrará.
3. **Dosis:** Cantidad y forma de administración del medicamento.
4. **Diagnóstico:** Razón o condición médica por la que se receta el medicamento.
5. **Indicaciones:** Instrucciones detalladas sobre cómo administrar la medicación.
6. **Ubicación:** Lugar donde se encuentra la mascota al momento de recibir la medicación.

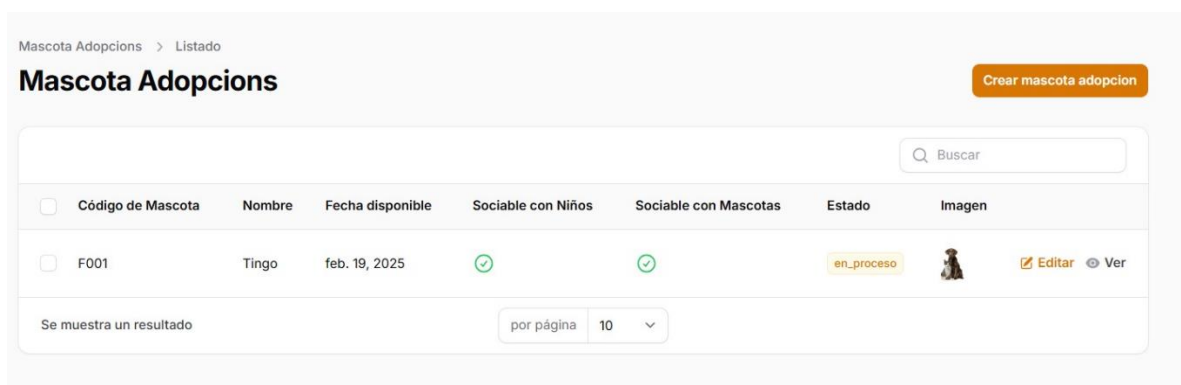
7. **Fecha de inicio:** Día en que comienza la administración del medicamento (formato dd/mm/aaaa).
8. **Fecha de fin:** Día en que termina la administración del medicamento (formato dd/mm/aaaa).
9. **Frecuencia:** Periodicidad con la que se debe administrar el medicamento (diaria, semanal, etc.).
10. **Estado:** Estado de la medicación (activa o inactiva).
11. **Veterinario responsable:** Nombre del veterinario encargado del tratamiento.
12. **Observaciones:** Comentarios adicionales sobre la medicación o el estado de la mascota.

Opciones de acción:

- **Crear:** Guarda el registro de la medicación en el sistema.
- **Crear y crear otro:** Guarda el registro actual y permite registrar una nueva medicación de inmediato.
- **Cancelar:** Descarta los datos ingresados y regresa a la lista de medicaciones.

Figura 28


Administrador - mascotas para adopción



Mascota Adopcions > Listado

Mascota Adopcions Crear mascota adopcion

🔍 Buscar

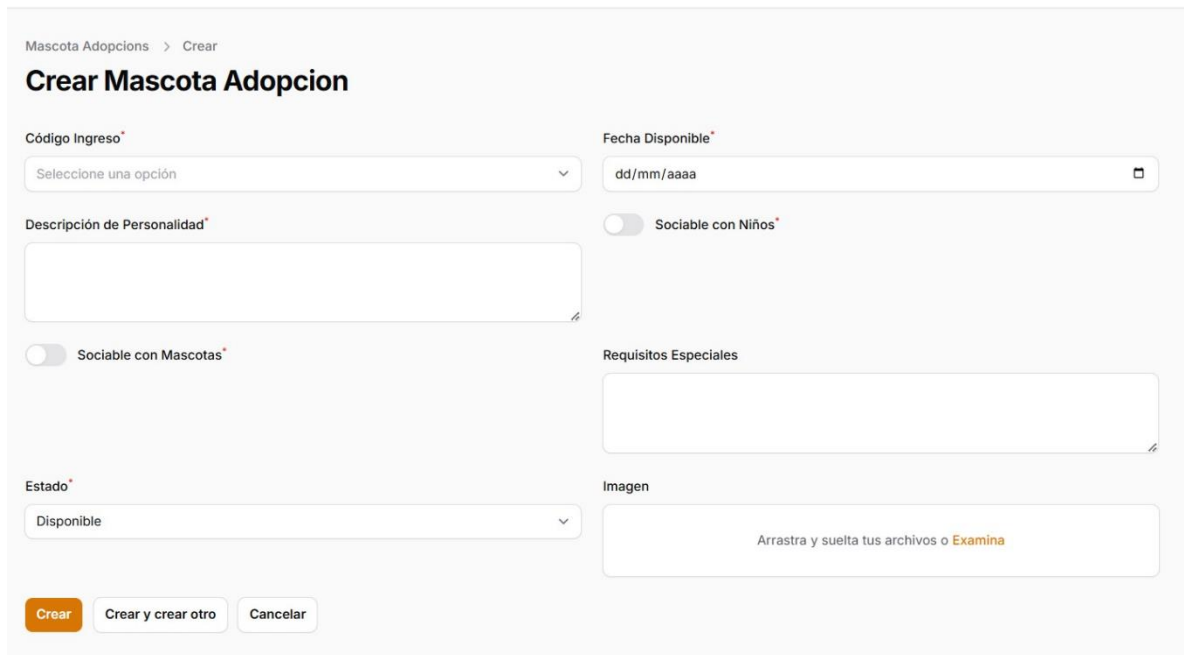
<input type="checkbox"/>	Código de Mascota	Nombre	Fecha disponible	Sociable con Niños	Sociable con Mascotas	Estado	Imagen
<input type="checkbox"/>	F001	Tingo	feb. 19, 2025	✔	✔	en_proceso	 ✎ Editar 👁 Ver

Se muestra un resultado por página 10 ▼

Finalmente, las mascotas que han pasado por el proceso de ingreso, esterilización y tratamiento médico, quedan listas para ser publicadas en el sistema de adopción.

Figura 29

Administrador - formulario mascotas para adopción



The screenshot shows a web form titled "Crear Mascota Adopción" within a breadcrumb trail "Mascota Adopcions > Crear". The form contains the following fields and controls:

- Código Ingreso***: A dropdown menu with the placeholder text "Seleccione una opción".
- Fecha Disponible***: A date input field with the placeholder "dd/mm/aaaa" and a calendar icon.
- Descripción de Personalidad***: A large text area for describing the pet's personality.
- Sociable con Niños***: A toggle switch, currently turned off.
- Sociable con Mascotas***: A toggle switch, currently turned off.
- Requisitos Especiales**: A text area for specifying special requirements.
- Estado***: A dropdown menu with "Disponible" selected.
- Imagen**: A file upload area with the text "Arrastra y suelta tus archivos o Examina".

At the bottom of the form, there are three buttons: "Crear" (highlighted in orange), "Crear y crear otro", and "Cancelar".

La interfaz de "Crear Mascota Adopción", donde el administrador del sistema puede registrar una nueva mascota lista para adopción.

Descripción de los Campos

1. **Código de Ingreso:** Selección del código único asociado a la mascota ingresada al sistema.
2. **Fecha Disponible:** Día a partir del cual la mascota estará disponible para adopción.
3. **Descripción de Personalidad:** Espacio para detallar el comportamiento y características de la mascota.
4. **Sociable con Mascotas / Niños:** Botones de activación para indicar si la mascota puede convivir con otras mascotas y niños.

5. **Estado:** Indica si la mascota está disponible para adopción.
6. **Requisitos Especiales:** Campo para especificar si la mascota tiene condiciones o necesidades especiales.
7. **Imagen:** Espacio para adjuntar una foto de la mascota.

Opciones de Acción

- **Crear:** Guarda el registro y finaliza el proceso.
- **Crear y crear otro:** Guarda el registro actual y permite registrar otra mascota.
- **Cancelar:** Descarta el proceso sin guardar cambios.

3.2 Resultados funcionales

Este sistema Web de adopción de mascotas cuenta con una interfaz sencilla de usar, de esta manera pueden comprender y utilizar de la mejor manera este aplicativo Web que busca promover el bienestar de todas las mascotas.

3.3 Resultados de las pruebas realizadas

Tabla 26

Resultados de las pruebas realizadas

Número de Caso de Prueba	Módulo	Parámetros de Entrada	Pasos a Seguir	Resultado Esperado	Resultado Obtenido	Observaciones
CP-01	Registro de Mascotas	Nombre, raza, descripción, foto	1. Ingresar al formulario de creación de mascota. 2. Completar todos los campos obligatorios. 3. Presionar "Guardar".	Mascota registrada exitosamente	Éxito	Sin errores detectados

CP-02	Búsqueda de Mascotas	Filtros: especie, tamaño, edad	1. Acceder al módulo de búsqueda. 2. Introducir filtros. 3. Presionar "Buscar".	Mostrar resultados relevantes	Éxito	Resultados rápidos y claros
CP-03	Registro de Usuarios	Nombre, correo, contraseña	1. Acceder al formulario de registro. 2. Completar los campos de información. 3. Presionar "Registrar".	Usuario registrado correctamente	Éxito	Registro sin fallos
CP-04	Panel de Administrador	ID de usuario, ID de mascota, estado	1. Ingresar como administrador. 2. Gestionar mascotas y solicitudes de adopción. 3. Aprobar o rechazar solicitudes.	Gestión eficiente de adopciones	Éxito	Interfaz fluida

3.3.1 Análisis de los resultados

El sistema Web ha demostrado ser eficiente para optimizar los procesos de adopción y localización de mascotas. Los principales beneficios identificados son:

3.3.2 Facilidad de uso

La interfaz de usuario se ha creado para que sea intuitiva y permita navegar fácilmente por el sistema incluso a quienes no tienen experiencia. Los módulos de registro, búsqueda y gestión de adopciones han sido evaluados positivamente por los usuarios.

3.3.3 Eficiencia:

El sistema ha demostrado ser eficiente en la optimización de los procesos de adopción y localización de mascotas. La automatización de la gestión de registros y solicitudes, junto con la implementación de filtros para la búsqueda de mascotas, permite una reducción

significativa en los tiempos de espera y facilita la comunicación entre los usuarios y los administradores.

3.3.4 Visibilidad y alcance:

El sistema ha incrementado la visibilidad de las mascotas disponibles ya sean por parte del centro temporal de rescate y también por parte de la comunidad, contribuyendo a un mayor índice de adopción. Además, la posibilidad de que los usuarios informen de mascotas perdidas ha mejorado la recuperación de animales extraviados, permitiendo a la comunidad participar activamente en la localización de mascotas y en la adopción.

3.3.5 Escalabilidad y mantenimiento:

El sistema ha sido diseñado de manera que se pueda escalar para incluir nuevas funcionalidades y gestionar un mayor volumen de datos conforme el centro temporal de rescate crece. La arquitectura modular y el uso de tecnologías estándar aseguran que el sistema pueda ser mantenido y actualizado con facilidad.

CONCLUSIONES

El sistema Web diseñado para la adopción y localización de mascotas ha demostrado ser una herramienta eficiente para agilizar los procesos administrativos en el centro temporal de rescate Fauna Urbana. Ha facilitado la gestión de adopciones y el seguimiento de mascotas perdidas, mejorando significativamente la visibilidad de los animales disponibles para adopción y promoviendo la interacción entre la comunidad y los administradores.

Los usuarios han expresado una valoración positiva sobre la simplicidad de la interfaz, destacando la facilidad con la que pueden navegar por el sistema y realizar las acciones necesarias, independientemente de su nivel de experiencia técnica. Este diseño accesible ha sido clave para la rápida adopción de la plataforma.

En cuanto a la protección de datos, el sistema implementa medidas de seguridad confiables, como el cifrado de contraseñas, lo cual garantiza la privacidad de la información de los usuarios.

La estructura del sistema, diseñada de forma flexible, ha permitido que el Software se adapte fácilmente a futuras actualizaciones o ampliaciones de funcionalidad, lo que asegura que se pueda escalar sin dificultades conforme aumenten las necesidades del proyecto.

RECOMENDACIONES

Una posible mejora sería ofrecer mayores opciones de personalización para las publicaciones de adopción. Permitir que los administradores incluyan información detallada sobre el comportamiento y necesidades de cada mascota podría ayudar a los adoptantes a tomar decisiones más informadas.

La incorporación de un sistema de notificaciones sería útil para mantener a los adoptantes actualizados sobre nuevas mascotas en adopción, cambios en el estado de sus solicitudes y novedades sobre animales perdidos, lo que optimizaría la comunicación y la experiencia de los usuarios.

Aunque el sistema de búsqueda es eficaz, sería beneficioso añadir más filtros específicos, como la búsqueda por necesidades especiales de las mascotas o por características de comportamiento, para que los adoptantes puedan encontrar rápidamente a la mascota que mejor se adapte a sus preferencias.

Es recomendable realizar pruebas de carga adicionales para evaluar el rendimiento del sistema en situaciones de alto tráfico. Estas pruebas ayudarían a garantizar que el sistema pueda soportar un número creciente de usuarios y registros sin afectar su funcionamiento.

Para garantizar que el personal aproveche al máximo todas las funcionalidades del sistema, se recomienda ofrecer sesiones de capacitación periódicas a los administradores. Esto no solo asegurará el uso adecuado del sistema, sino también la mejora continua en la gestión de adopciones.

Integrar el sistema con otras plataformas de adopción o redes sociales podría ampliar la visibilidad de las mascotas y fomentar una mayor participación de la comunidad en la adopción y recuperación de animales.

Establecer un mecanismo de retroalimentación donde los usuarios puedan expresar sus opiniones y sugerencias contribuiría a una mejora constante del sistema, asegurando que se mantenga alineado con las expectativas y necesidades de los usuarios.

BIBLIOGRAFÍA

- Andrade Tituaña, K. E. (agosto de 2018). Desarrollo de un sistema web y móvil de registro y control de mascotas del gobierno autónomo descentralizado municipal de Otavalo, para las plataformas Ios Y Android. Ibarra, Imbabura, Ecuador.
- Arcos Arcos, J. D. (febrero de 2024). APLICACIÓN MÓVIL PARA LA GESTIÓN DE ADOPCIÓN DE MASCOTAS RESCATADAS EN EL REFUGIO ANIMAL KODAH DEL CANTÓN CEVALLOS. Ambato, Tungurahua, Ecuador.
- Asencio, H. L. (Junio de 2014). Seguridad en aplicaciones web: Una vision practica. Leganes, España.
- Autentia. (2022). *Guía Completa Front*. Madrid, España: Autentia. Retrieved 14 de noviembre de 2024, from https://ahorasomos.izertis.com/autentia/wp-content/uploads/libros/Front_GuiaCompleta-Autentia.pdf
- Axarnet, A. (18 de abril de 2018). *Laravel el Framework de PHP - Qué es y para qué sirve*. <https://www.scribbr.es/citar/generador/citar/entrada-de-blog/?searchQuery=https%3A%2F%2Faxarnet.es%2Fblog%2Fque-es-laravel>
- Borja Machin, M. (3 de enero de 2022). Desarrollo de una aplicación web para adoptar perros y gatos. Cataluña, España.
- Celi Párraga, R. J., Boné Andrade, M. F., & Mora Olivero, A. P. (2023). *Programación Web del Frontend al Backend*. Santo Domingo, Santo Domingo, Ecuador: Grupo AEA. Retrieved 14 de noviembre de 2024, from file:///C:/Users/andy_/Downloads/Dialnet-ProgramacionWebDelFrontendAlBackend-933116.pdf
- Colcha Conejo, J. L. (octubre de 2022). Desarrollo de un sistema web y aplicacion móvil para gestionar la adopción de mascotas en la ciudad de quito: Desarrollo del Frontend. Quito, Pichincha, Ecuador.
- Coronado Garcia, B. (Septiembre de 2024). Desarrollo de aplicaciones web en el entorno servidor.
- Fonseca Vargas , M. A. (2020). La programacion extrema aplicada al desarrollo del sistema de informacion web para la gestion de presupuestos en CEM comunicaciones S.A. *Metodologias Agiles de Desarrollo de Software*. Nicaragua.
- Gilfilan, I. (2021). *La Biblia del MySQL*. Madrid, España: Anaya Multimedia. Retrieved 15 de diciembre de 2025, from <https://persefone20.wordpress.com/wp-content/uploads/2011/09/la-biblia-de-mysql.pdf>
- GoDaddy, E. d. (11 de septiembre de 2024). *Páginas web: Definición, tipos y características*. GoDaddy Resources - LATAM: <https://www.godaddy.com/resources/latam/stories/pagina-web-que-es-tipos#h-que-es-una-pagina-web>
- Gutierrez Benitez, F. A. (14 de agosto de 2020). Sistema web de adopción de animales CEMP. Santiago de Chile, Chile.

- Hernández Bejarano, M., & Baquero Rey, L. E. (2020). *Web, Fundamentos de Programacion*. (C. García, Ed.) Cali, Valle del Cauca, Colombia: Universidad ECCI. Retrieved 12 de diciembre de 2024, from <https://www.ecci.edu.co/wp-content/uploads/2022/02/Fundamentos-de-Programacion-Web-version-1.0-EDITORIAL-ECCI.pdf>
- Mendez Muñoz, V. (30 de Diciembre de 2019). Analisis de sistema de autenticacion y autorizacion para entornos web distribuidos. Cataluña, España.
- Moure, B. (2024). *Git y GitHub desde cero*. Madrid, España: Independently published. Retrieved 14 de diciembre de 2024, from <https://www.amazon.com/-/es/Git-GitHub-desde-cero-te%C3%B3rico-pr%C3%A1ctica/dp/B0C1J3FG6Q>
- Pata Vargas, P. S. (junio de 2024). Mejorando el bienestar animal y promoviendo la sostenibilidad a través de la tecnología: desarrollo de una aplicación móvil para la adopción de mascotas. Esmeraldas, Esmeraldas, Ecuador.
- Pérez Ruíz, J. S. (septiembre de 2019). Adoptapp. Aplicación móvil de adopción de animales domesticos que habitan las calles de Bogotá. Bogotá, Colombia.
- PHP. (s.f.). *PHP: Hypertext Preprocessor*. <https://www.scribbr.es/citar/generador/folders/6a9gFoF16uwrSH6NHfpnuk/lists/5zJ5SXA7jqVSGNOhWJonWS/>
- Triviño Zambrano, F. O. (Mayo de 2018). Aplicación web-móvil para la gestión de adopciones de animales rescatados en la fundación rescate animal. *Aplicación web-móvil para la gestión de adopciones de animales rescatados en la fundación rescate animal*. Ambato, Tungurahua, Ecuador.
- Vargas Zermeño, E. (2022). *Implementación de Tailwindcss en proyecto web*. Visibilidad Web UNAM. Retrieved 16 de diciembre de 2024, from https://www.visibilidadweb.unam.mx/sites/default/files/2023-01/Guia_de_instalacion_Tailwind.pdf#page=2&zoom=100,96,594

4 ANEXOS

Centro temporal de Rescate Fauna Urbana

Parque Ciudad Blanca, junto al segundo Puente. Entre Avda. Camino Ponce y Avda. Galo Plaza Lasso.

(+593) 063700200

faunaurbana@ibarra.gob.ec

Fecha: 13 de febrero del 2025

A quien corresponda:

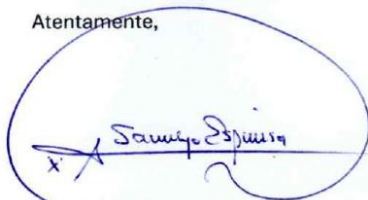
Nos dirigimos a ustedes para comunicar que, tras la finalización del desarrollo de la aplicación web **Fauna Urbana**, han cumplido con los compromisos adquiridos inicialmente, tal y como se estableció en las historias de usuarios definidas al inicio del proyecto.

El equipo de desarrollo, compuesto por **Jairo Hidalgo** y **Andy Usiña**, se encargó de implementar las funcionalidades necesarias para el correcto funcionamiento de la plataforma, según lo especificado en las historias de usuarios. Nos complace comunicar que todas las funcionalidades han sido entregadas y validadas conforme a lo estipulado.

Durante todo el proceso, se han seguido las especificaciones acordadas y se ha trabajado en estrecha colaboración con el equipo de desarrollo, con el objetivo de cumplir con los plazos y expectativas de calidad establecidos. Las funcionalidades solicitadas, tanto en términos de desarrollo como de pruebas, se han implementado exitosamente.

Las historias de usuarios detalladas en la documentación del proyecto han sido completadas, y hemos asegurado que cada requerimiento esté reflejado en la versión final de la aplicación, brindando una solución funcional y efectiva. Además, hemos realizado las pruebas correspondientes para verificar que todas las funcionalidades operan según lo esperado.

Atentamente,



Ing. Claudio Vaca

Persona a Cargo

Centro temporal de Rescate Fauna Urbana

Script de la Base de Datos:

-- Estructura de tabla para la tabla admins

```
CREATE TABLE admins (  
  
  id bigint(20) UNSIGNED NOT NULL,  
  
  name varchar(255) NOT NULL,  
  
  email varchar(255) NOT NULL,  
  
  email_verified_at timestamp NULL DEFAULT NULL,  
  
  password varchar(255) NOT NULL,  
  
  remember_token varchar(100) DEFAULT NULL,  
  
  created_at timestamp NULL DEFAULT NULL,  
  
  updated_at timestamp NULL DEFAULT NULL  
  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

-- Estructura de tabla para la tabla candidatos

```
CREATE TABLE candidatos (  
  
  id bigint(20) UNSIGNED NOT NULL,  
  
  user_id bigint(20) UNSIGNED NOT NULL,  
  
  adopcion_id bigint(20) UNSIGNED NOT NULL,  
  
  adoptar_mascota varchar(255) NOT NULL,  
  
  responsable_mascota varchar(255) NOT NULL,
```

```
tiene_vivienda varchar(255) NOT NULL,  
  
tipo_vivienda varchar(255) NOT NULL,  
  
familia_de_acuerdo varchar(255) NOT NULL,  
  
enfermedad_mascota varchar(255) NOT NULL,  
  
cambio_domicilio varchar(255) NOT NULL,  
  
convivir_niños varchar(255) NOT NULL,  
  
tiempo_disponible varchar(255) NOT NULL,  
  
quien_quedaria_mascota varchar(255) NOT NULL,  
  
primera_adopcion varchar(255) NOT NULL,  
  
convive_otras_mascotas varchar(255) NOT NULL,  
  
created_at timestamp NULL DEFAULT NULL,  
  
updated_at timestamp NULL DEFAULT NULL,  
  
FOREIGN KEY (user_id) REFERENCES users(id),  
  
FOREIGN KEY (adopcion_id) REFERENCES daradopciones(id)  
  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

-- Estructura de tabla para la tabla categorias

```
CREATE TABLE categorias (  
  
id bigint(20) UNSIGNED NOT NULL,  
  
name varchar(255) NOT NULL,
```

```
active tinyint(1) NOT NULL DEFAULT 1,  
  
created_at timestamp NULL DEFAULT NULL,  
  
updated_at timestamp NULL DEFAULT NULL  
  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

-- Estructura de tabla para la tabla daradopciones

```
CREATE TABLE daradopciones (  
  
id bigint(20) UNSIGNED NOT NULL,  
  
user_id bigint(20) UNSIGNED NOT NULL,  
  
categoria_id bigint(20) UNSIGNED NOT NULL,  
  
titulo varchar(255) NOT NULL,  
  
nombre varchar(255) NOT NULL,  
  
sexo varchar(255) NOT NULL,  
  
raza varchar(255) DEFAULT NULL,  
  
tamaño varchar(255) NOT NULL,  
  
edad varchar(255) NOT NULL,  
  
descripcion text NOT NULL,  
  
activo tinyint(1) NOT NULL DEFAULT 1,  
  
imagen varchar(255) DEFAULT NULL,  
  
created_at timestamp NULL DEFAULT NULL,
```

```
updated_at timestamp NULL DEFAULT NULL,  
  
FOREIGN KEY (user_id) REFERENCES users(id),  
  
FOREIGN KEY (categoria_id) REFERENCES categorias(id)  
  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

```
-- Estructura de tabla para la tabla notifications
```

```
CREATE TABLE notifications (  
  
id char(36) NOT NULL,  
  
type varchar(255) NOT NULL,  
  
notifiable_type varchar(255) NOT NULL,  
  
notifiable_id bigint(20) UNSIGNED NOT NULL,  
  
data text NOT NULL,  
  
read_at timestamp NULL DEFAULT NULL,  
  
created_at timestamp NULL DEFAULT NULL,  
  
updated_at timestamp NULL DEFAULT NULL,  
  
FOREIGN KEY (notifiable_id) REFERENCES users(id)  
  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

```
-- Estructura de tabla para la tabla perdidos
```

```
CREATE TABLE perdidos (  

```

```
id bigint(20) UNSIGNED NOT NULL,  
  
user_id bigint(20) UNSIGNED NOT NULL,  
  
categoria_id bigint(20) UNSIGNED NOT NULL,  
  
nombre varchar(255) NOT NULL,  
  
sexo varchar(255) NOT NULL,  
  
raza varchar(255) DEFAULT NULL,  
  
tamaño varchar(255) NOT NULL,  
  
edad varchar(255) NOT NULL,  
  
fecha_perdida date NOT NULL,  
  
visto_ultima varchar(255) NOT NULL,  
  
ciudad varchar(255) NOT NULL,  
  
telefono_contacto varchar(255) NOT NULL,  
  
descripcion text NOT NULL,  
  
activo tinyint(1) NOT NULL DEFAULT 1,  
  
imagen varchar(255) DEFAULT NULL,  
  
created_at timestamp NULL DEFAULT NULL,  
  
updated_at timestamp NULL DEFAULT NULL,  
  
FOREIGN KEY (user_id) REFERENCES users(id),  
  
FOREIGN KEY (categoria_id) REFERENCES categorias(id)  
  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

-- Estructura de tabla para la tabla publicacions

```
CREATE TABLE publicacions (  
  
  id bigint(20) UNSIGNED NOT NULL,  
  
  admin_id bigint(20) UNSIGNED NOT NULL,  
  
  categoria_id bigint(20) UNSIGNED NOT NULL,  
  
  titulo varchar(255) NOT NULL,  
  
  nombre varchar(255) NOT NULL,  
  
  sexo varchar(255) NOT NULL,  
  
  raza varchar(255) DEFAULT NULL,  
  
  tamaño varchar(255) NOT NULL,  
  
  edad varchar(255) NOT NULL,  
  
  descripcion varchar(255) NOT NULL,  
  
  slug varchar(255) NOT NULL,  
  
  activo tinyint(1) NOT NULL DEFAULT 1,  
  
  views bigint(20) NOT NULL DEFAULT 0,  
  
  likes bigint(20) NOT NULL DEFAULT 0,  
  
  dislikes bigint(20) NOT NULL DEFAULT 0,  
  
  asset_url varchar(255) DEFAULT NULL,  
  
  created_at timestamp NULL DEFAULT NULL,
```

```
updated_at timestamp NULL DEFAULT NULL,  
  
FOREIGN KEY (admin_id) REFERENCES admins(id),  
  
FOREIGN KEY (categoria_id) REFERENCES categorias(id)  
  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;  
  
  
-- Estructura de tabla para la tabla users  
  
CREATE TABLE users (  
  
id bigint(20) UNSIGNED NOT NULL,  
  
name varchar(255) NOT NULL,  
  
email varchar(255) NOT NULL,  
  
email_verified_at timestamp NULL DEFAULT NULL,  
  
password varchar(255) NOT NULL,  
  
remember_token varchar(100) DEFAULT NULL,  
  
created_at timestamp NULL DEFAULT NULL,  
  
updated_at timestamp NULL DEFAULT NULL,  
  
rol int(11) NOT NULL  
  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```