



Sheltra

Elba Calvo García
2º Desarrollo de Aplicaciones Multiplataforma

Índice:

1. Introducción	2
1.1. Contextualización	2
1.2. Estructura organizativa y funciones de la empresa	3
2. Tecnologías usadas	4
2.1. Explicación de las tecnologías usadas	4
2.2. Motivos de la elección de las tecnologías	5
3. Arquitectura	7
3.1. Arquitectura usada	7
4. Análisis de requisitos	8
4.1. Captación de requisitos	8
4.2. Información relativa a la realización del proyecto	9
5. Diseño estático	13
5.1. Creación de diagramas	13
6. Modelo de datos	14
6.1. Definición de acceso a datos	14
7. Manual	16
7.1. Creación de manuales de usuario e instalación	16
8. Viabilidad	27
8.1. Estudio de mercado	27
8.2. Análisis económico	27
8.3. Estudio de riesgos	28
9. Calidad	29
9.1. Pruebas	29
10. Conclusiones y mejoras futuras	35
10.1. Conclusiones	35
11. Bibliografía	36
11.1. Bibliografía	36

1. Introducción

Sheltra es una plataforma web orientada a facilitar la adopción responsable de animales domésticos. El proyecto surge como respuesta a varios problemas sociales urgentes, como el alto número de animales sin hogar, la dificultad que tienen muchas personas para hacerse cargo de crías no deseadas, y la falta de plataformas accesibles, seguras y éticas que conecten a quienes necesitan dar en adopción con adoptantes.

A diferencia de un refugio normal, Sheltra ni cuenta con un espacio físico para alojar animales. En su lugar, funciona como un punto de encuentro digital entre personas que, por distintos motivos, no pueden seguir cuidando de un animal, y aquellas que buscan darle un nuevo hogar. Todo el proceso se plantea desde un enfoque legal y gran compromiso con el bienestar animal.

Además, para quienes no pueden adoptar pero quieren ayudar igualmente, la plataforma ofrece una sección de donaciones. A través de esta, los usuarios pueden colaborar económicamente con refugios y organizaciones con las que trabajamos. Las donaciones se transfieren directamente a estas organizaciones, asegurando que la ayuda llegue sin intermediarios a quienes más lo necesitan.

1.1. Contextualización

El abandono animal sigue siendo una problemática en muchas regiones, y aunque existen refugios y protectoras, sus recursos y capacidad de acogida son limitados. Por otro lado, muchas personas que no pueden hacerse cargo de animales de forma permanente se ofrecen como cuidadores temporales, pero no siempre cuentan con medios adecuados para encontrar adoptantes responsables. Sheltra surge como una alternativa accesible.

Sheltra se presenta como una respuesta a esta necesidad social. La plataforma está diseñada para personas que, por cuenta propia, necesitan una vía legal, ética y segura para dar en adopción a los animales que están bajo su cuidado.

- ❖ Identificación de la empresa en el sector y mercado objetivo

Sheltra se ubica en el sector de servicios web orientados a la protección y bienestar animal. A diferencia de otras webs del sector, no está directamente vinculada a protectoras o refugios, sino que funciona como un intermediario entre particulares, promoviendo la adopción responsable.

Sheltra se presenta como una empresa digital con enfoque social, especializada en facilitar la adopción responsable de animales domésticos. Su estructura empresarial responde a un modelo que busca equilibrar la sostenibilidad operativa con un fuerte compromiso ético, facilitando un

entorno seguro y confiable tanto para adoptantes como para cuidadores.

- Mercado objetivo:

- ❖ Personas que necesitan dar en adopción a un animal y buscan una vía segura para hacerlo.
- ❖ Adoptantes potenciales que buscan una adopción responsable y transparente.
- ❖ Personas comprometidas con la causa animal que desean colaborar económicamente mediante donaciones.
- ❖ Usuarios de entre 18 y 45 años, familiarizados con herramientas digitales, principalmente en áreas urbanas.

1.2. Estructura organizativa y funciones de la empresa

Sheltra es una empresa legalmente constituida sin ánimo de lucro. Su estructura organizativa está diseñada para garantizar una gestión eficiente, transparente y coherente con sus valores éticos.

Dado su carácter sin fines lucrativos, Sheltra se financia a través de donaciones particulares, subvenciones públicas, convenios de colaboración con entidades protectoras y, en menor medida, apoyos puntuales de empresas privadas con compromiso social. Esto permite cubrir los costes esenciales de operación y, cuando es posible, contratar personal o recurrir a colaboradores externos.

- ❖ Dirección General

Funciones:

- Coordinación general del proyecto.
- Toma de decisiones estratégicas y supervisión operativa.
- Representación legal y de relaciones institucionales.
- Búsqueda de financiación y gestión de convenios.

- ❖ Departamento de Desarrollo Web

Funciones:

- Diseño, programación y mantenimiento de la plataforma.
- Desarrollo de nuevas funcionalidades y mejoras técnicas.
- Control de calidad, seguridad y pruebas.

- ❖ Departamento de Comunicación y Marketing

Funciones:

- Gestión de redes sociales y contenido digital.
- Difusión de campañas de adopción, concienciación y recaudación.
- Atención al usuario.
- Fortalecimiento de la imagen.

- ❖ Departamento Legal y Ético
 - Funciones:
 - Garantizar el cumplimiento legal de los procesos de adopción.
 - Redacción y revisión de términos legales.
 - Asesoramiento en la protección de datos y derechos de los usuarios.
 - Mediación en caso de conflictos o irregularidades.
- ❖ Departamento de Soporte y Comunidad
 - Funciones:
 - Soporte técnico y resolución de incidencias.
 - Gestión de la comunidad de usuarios adoptantes y donantes.

2. Tecnologías usadas

2.1. Explicación de las tecnologías usadas

Para la creación de la aplicación:

- **HTML:** Es el lenguaje base que se utiliza para estructurar las páginas web. Con él se definen los elementos fundamentales de la interfaz, como los textos, imágenes, botones, formularios o enlaces. Se ha usado para construir la parte visual y organizativa de la plataforma.
- **CSS:** Se ha utilizado para dar estilo y personalidad a los elementos definidos con HTML. Gracias a CSS se ha podido adaptar la apariencia del sitio (colores, tamaños, tipografía, espaciado...).
- **PHP:** Es el lenguaje que se ha encargado de la lógica del lado del servidor. Se ha usado principalmente para procesar formularios, conectar con la base de datos, gestionar sesiones de usuario y generar contenido dinámico en la web.
- **XAMPP:** Para poder trabajar de forma local sin necesidad de un servidor externo, he utilizado XAMPP, un paquete que incluye todo lo necesario: **Apache** como servidor web, **MySQL** como gestor de base de datos, y PHP como lenguaje de servidor. Esto ha permitido desarrollar y probar la aplicación directamente desde mi ordenador.
- **phpMyAdmin:** Herramienta basada en web que facilita la administración de bases de datos MySQL. Ofrece una interfaz gráfica intuitiva para crear tablas, ejecutar consultas SQL, modificar datos y gestionar usuarios.
- **Github:** Se ha utilizado para llevar un control de versiones del código y documentar el proyecto en un entorno colaborativo. Además, facilita

compartir el proyecto y mantener un historial de los cambios realizados.

- **Visual Studio Code:** Ha sido el editor de código principal durante el desarrollo. Es ligero, personalizable y tiene muchas extensiones útiles que han facilitado tanto la escritura de código como la revisión y depuración.

Para la creación de la documentación:

- **Google Docs:** He utilizado Google Docs como herramienta principal para redactar y organizar la memoria del proyecto.
- **Google Drive:** Ha servido para almacenar los documentos del proyecto en la nube, facilitando el acceso desde distintos dispositivos
- **Canva:** Herramienta de diseño gráfico en línea para crear materiales visuales, como la portada del proyecto, logos o banners.
- **Figma:** Antes de comenzar con la programación, utilicé Figma para diseñar los primeros prototipos de la interfaz. Esta herramienta permite crear esquemas visuales (wireframes) y diseños más detallados de las pantallas de la web.
- **GitHub (README.md):** Dentro del repositorio del proyecto, se ha incluido un archivo README.md donde se explica de forma breve el funcionamiento general de la aplicación, sus objetivos y las instrucciones para su instalación.

2.2. Motivos de la elección de las tecnologías

Para la creación de la aplicación:

- **HTML:** Elegí HTML porque es el estándar universal para estructurar páginas web. A diferencia de otras alternativas como editores visuales o sistemas de plantillas cerrados, HTML permite tener control total sobre el contenido y la estructura. Además, es compatible con todos los navegadores y no depende de frameworks externos.
- **CSS:** Usé CSS para el diseño visual porque es el lenguaje más extendido y flexible para personalizar interfaces web. Comparado con soluciones como Bootstrap o Tailwind (que también se basan en CSS), preferí escribir estilos propios para tener más libertad estética y mantener la página ligera. CSS también permite aplicar diseño responsive sin necesidad de bibliotecas adicionales.

- **PHP:** Aunque existen otros lenguajes de backend como Python (con Django o Flask) o JavaScript (con Node.js), opté por PHP porque se integra directamente con HTML y es ideal para proyectos pequeños o medianos. Funciona bien con bases de datos MySQL y además, al estar incluido en XAMPP, facilita mucho el desarrollo en local sin configuraciones complicadas.
- **XAMPP:** Frente a otras opciones como WAMP o MAMP, XAMPP destaca por su compatibilidad multiplataforma (Windows, Linux, macOS) y por incluir todo lo necesario (Apache, PHP y MySQL) en un único instalador. Esto ahorra tiempo y evita conflictos entre dependencias.
- **phpMyAdmin:** Para trabajar con la base de datos, preferí phpMyAdmin antes que usar directamente la consola de MySQL u otras herramientas como DBeaver. La razón es su interfaz gráfica intuitiva, que permite hacer tareas comunes como crear tablas o ejecutar consultas sin necesidad de escribir todo a mano.
- **Github:** GitHub fue la plataforma elegida para control de versiones, por encima de otras como GitLab o Bitbucket, principalmente por su popularidad, facilidad de uso y herramientas de documentación. Publicar el proyecto en GitHub permite mantenerlo organizado, compartirlo fácilmente y demostrar buenas prácticas de desarrollo.
- **Visual Studio Code:** Usé Visual Studio Code como entorno de desarrollo porque es ligero, rápido y muy personalizable. Frente a editores más pesados como Eclipse o entornos limitados como Notepad++, VS Code ofrece autocompletado, resaltado de sintaxis, depuración integrada y soporte para múltiples lenguajes en un solo editor.

Para la creación de la documentación:

- **Google Docs:** Elegí Google Docs frente a procesadores como Microsoft Word porque permite trabajar online, guardar automáticamente los avances y acceder desde cualquier lugar. Su compatibilidad con PDF también hace más sencilla la exportación final del documento.
- **Google Drive:** Usar Drive me resultó más cómodo que depender de almacenamiento local o memorias USB. Además, permite organizar carpetas, compartir archivos y tener todo el proyecto centralizado y respaldado.
- **Canva:** Canva es más accesible que herramientas como Photoshop o Illustrator. Su interfaz es más sencilla, tiene plantillas listas para usar y no requiere instalación ni conocimientos avanzados de diseño.
- **Figma:** Para diseñar la interfaz antes de programar, Figma me pareció

la mejor opción frente a otras herramientas. Al ser una aplicación basada en la web, permite trabajar desde cualquier dispositivo sin necesidad de instalación. Además, su sistema de prototipado interactivo facilita visualizar cómo funcionará la plataforma antes de implementarla, lo que mejora tanto el diseño como la experiencia de usuario.

- **GitHub (README.md):** Frente a la documentación en PDFs o archivos separados, el README es más directo, accesible y estándar en proyectos de software. Incluye instrucciones de instalación, objetivos del proyecto y detalles técnicos, lo que lo hace ideal para presentar el trabajo de forma clara.

3. Arquitectura

3.1. Arquitectura usada

Sheltra se basa en una arquitectura cliente-servidor, una de las arquitecturas más utilizadas en el desarrollo de aplicaciones web. En este modelo el sistema se divide en dos partes.

- **Cliente:** Es el navegador del usuario, que envía peticiones (solicitar ver el perfil de un animal, enviar un formulario de adopción...).
- **Servidor:** Es el que procesa esas peticiones. En este caso, el servidor está simulado localmente mediante XAMPP, que incluye Apache como servidor web y MySQL como gestor de base de datos. El lenguaje PHP se encarga de procesar la lógica del lado servidor.

Cuando un usuario interactúa con la web, su navegador (cliente) envía una solicitud al servidor. PHP procesa esa solicitud, accede a la base de datos si es necesario mediante MySQL y devuelve una respuesta.

Ventajas de la arquitectura:

- El cliente se encarga de la presentación (HTML/CSS) y el servidor se ocupa de la lógica y los datos (PHP/MySQL).
- Aunque el proyecto está en un entorno local, esta arquitectura permite crecer fácilmente con un servidor remoto.
- Al estar la lógica de negocio y la base de datos en el servidor, el usuario no tiene acceso directo a los datos ni al código interno.
- Es un modelo muy conocido y bien documentado.
- Al funcionar sobre HTTP, es accesible desde cualquier dispositivo con navegador.

Además de la arquitectura cliente-servidor, el proyecto también cuenta con una estructura interna basada en el patrón **MVC (Modelo Vista)**

Controlador). Este modelo permite separar la lógica del negocio, la presentación de la información y el control de las acciones del usuario.

Esta organización mejora la claridad del código y facilita su mantenimiento y escalabilidad. En el caso de Sheltra, la aplicación de MVC se refleja de la siguiente forma:

- **Modelo:** Archivos PHP que se encargan de acceder y gestionar la base de datos, por ejemplo para obtener información de los animales o guardar solicitudes de adopción.
- **Vista:** Páginas HTML/CSS que muestran la información al usuario de forma visual y accesible.
- **Controlador:** Archivos PHP que actúan como intermediarios entre el modelo y la vista, gestionando las peticiones del usuario y coordinando las respuestas.

Ventajas del uso de MVC:

- Favorece un desarrollo ordenado y limpio.
- Permite separar claramente la presentación del contenido y la lógica de negocio.
- Facilita el mantenimiento y futuras ampliaciones del proyecto.
- Mejora la reutilización de componentes.

La arquitectura cliente-servidor sigue siendo una de las más utilizadas en aplicaciones web, especialmente en entornos donde se combina **PHP con MySQL**. Muchas plataformas populares están basadas en esta misma arquitectura (como WordPress, Moodle, Prestashop...). A parte, su simplicidad y eficacia la convierten en una muy buena opción para proyectos pequeños, medianos o educativos.

Aunque han aparecido arquitecturas más modernas como, el modelo **cliente-servidor** sigue teniendo una fuerte presencia en muchos sitios web y aplicaciones profesionales en todo el mundo.

Por otro lado, el patrón **MVC** es una arquitectura de software ampliamente adoptada en el desarrollo web. Es la base de numerosos frameworks modernos (por ejemplo, en PHP, **Laravel**, **Symfony**, **CodeIgniter**...), y se considera un estándar tanto en entornos educativos como en proyectos profesionales. Su uso facilita el desarrollo estructurado, modular y escalable de aplicaciones web, lo que lo convierte en una opción muy valorada en la industria.

4. Análisis de requisitos

4.1. Captación de requisitos

La definición de los requisitos para el desarrollo de la plataforma Sheltra se ha realizado siguiendo un enfoque práctico, centrado en las necesidades reales de los usuarios y en la misión de la empresa. Al tratarse de un proyecto impulsado desde una organización con un propósito social claro, la recogida de requisitos no se ha limitado a supuestos teóricos, sino que ha combinado análisis funcional con observación de referentes del sector y conocimiento del entorno.

- Investigación de plataformas similares (como AdoptaUnAmigo, Miwuki, etc.), analizando las funcionalidades más comunes.
- Empatía con posibles usuarios reales (personas que quieren dar en adopción o adoptar), planteando sus necesidades, objetivos y problemas.
- Análisis personal de funcionalidad mínima viable (MVP) para asegurar que la plataforma fuera útil, legal, sencilla y ética.

Historias de usuario:

Como técnica de definición de requisitos, se emplearon **historias de usuario**, que ayudan a expresar de forma sencilla qué espera conseguir un usuario típico. Algunos ejemplos utilizados en el desarrollo de Sheltra son:

- "*Como visitante de la web, quiero ver animales disponibles para adoptar sin necesidad de registrarme.*"
- "*Como usuario registrado, quiero poder subir un animal para darlo en adopción o adoptar*"
- "*Como adoptante, quiero que el proceso de adopción sea sencillo e intuitivo.*"
- "*Como persona solidaria, quiero hacer una donación a un refugio sin tener que adoptar.*"

Para complementar la especificación de requisitos, se ha elaborado un diagrama de casos de uso UML que representa gráficamente las principales interacciones entre los usuarios y el sistema (apartado 5).

4.2. Información relativa a la realización del proyecto

Ciclo de vida del proyecto

Para el desarrollo de Sheltra se ha seguido un modelo de ciclo de vida en **cascada**, caracterizado por una secuencia lineal de fases. Este modelo permite completar cada etapa antes de pasar a la siguiente, facilitando un avance claro.

Durante el desarrollo se enseñó un borrador que permitió identificar y corregir errores. Esta práctica resulta especialmente útil en proyectos académicos, donde el alcance y los requisitos suelen estar bien definidos.

La elección del modelo en cascada se basa en los siguientes motivos:

- Facilita la planificación y ejecución con etapas claramente delimitadas.
- Proporciona un control riguroso del progreso del proyecto.
- Es adecuado para proyectos con requisitos estables y poco propensos a cambios durante el desarrollo.

Fases del proyecto

El proyecto se ha dividido en las siguientes fases, con una duración estimada y sus respectivos objetivos:

1. Fase de análisis y captación de requisitos

- Duración: 7 días
- Objetivos: Identificar funcionalidades necesarias mediante análisis de otras plataformas y definición de historias de usuario y definir el alcance y propósito de Sheltra.

2. Diseño visual en Figma

- Duración: 6 días
- Objetivos: Crear un prototipo visual de la web que sirva como referencia para el desarrollo front-end, facilitando la codificación del diseño y haciendo que sea más rápido.

3. Diseño de la arquitectura y estructura de datos

- Duración: 5 días
- Objetivos: Crear la base de datos con sus tablas y relaciones y planificar la estructura del código MVC y preparar la inserción inicial de datos en la BD.

4. Desarrollo de la parte visual (HTML y CSS)

- Duración: 14 días
- Objetivos: Codificar la interfaz gráfica de la web.

5. Corrección de la parte visual

- Duración: 2 días
- Objetivos: Optimizar la experiencia de usuario, mejorar la usabilidad y aumentar la intuitividad de la interfaz.

6. Desarrollo del código de la lógica de la web

- Duración: 14 días
- Objetivos: Programar la funcionalidad de la aplicación codificando la lógica necesaria para que la web sea operativa.

7. Pruebas

- Duración: 5 días
- Objetivos: Detectar y corregir errores, validar formularios y mejorar el diseño mediante pruebas funcionales que simulan casos de uso reales.

8. Documentación del proyecto

- Duración: 7 días
- Objetivos: Redactar la memoria y la documentación para explicar el desarrollo, la arquitectura, las decisiones tomadas y los resultados obtenidos.

9. Presentación

- Duración: 7 días
- Objetivos: Preparar y exponer los resultados del proyecto, demostrando el funcionamiento de la web, destacando las funcionalidades clave y respondiendo a posibles dudas del jurado o público.

10. Repaso final

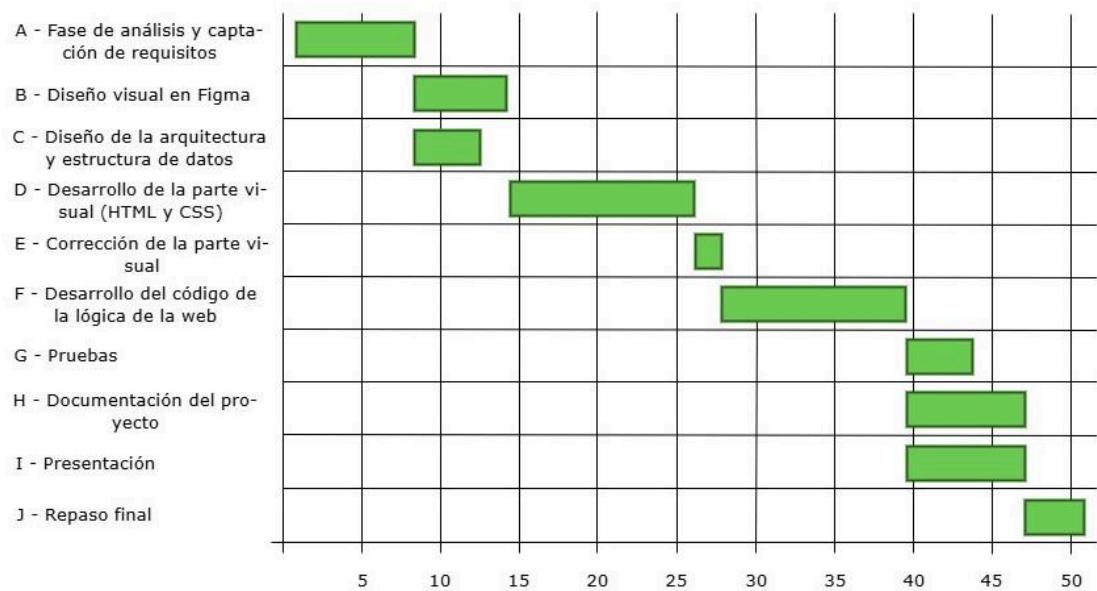
- Duración: 4 días
- Objetivos: Realizar una revisión completa del proyecto, corregir detalles finales, mejorar la presentación y asegurarse de que todo el contenido y funcionamiento estén listos para la entrega o exposición final.

A continuación representaré los datos anteriores con un diagrama Gantt, adjuntando también una tabla.

	Fase	Fecha inicio	Fecha final	Duración	Depende de
A	Fase de análisis y captación de requisitos	24/03/25	31/04/25	7 días	—
B	Diseño visual en Figma	01/04/25	07/04/25	6 días	A
C	Diseño de la arquitectura y estructura de datos	07/04/25	12/04/25	5 días	A
D	Desarrollo de la parte visual (HTML y CSS)	12/04/25	26/04/25	14 días	B
E	Corrección de la parte visual	26/04/25	28/04/25	2 días	D
F	Desarrollo del código de la lógica de la web	28/04/25	12/05/25	14 días	E

G	Pruebas	12/05/25	17/05/25	5 días	F
H	Documentación del proyecto	17/05/25	24/05/25	7 días	F
I	Presentación	25/05/25	30/05/25	7 días	F
J	Repaso final	30/05/25	02/06/25	4 días	I

- Diagrama Gantt:



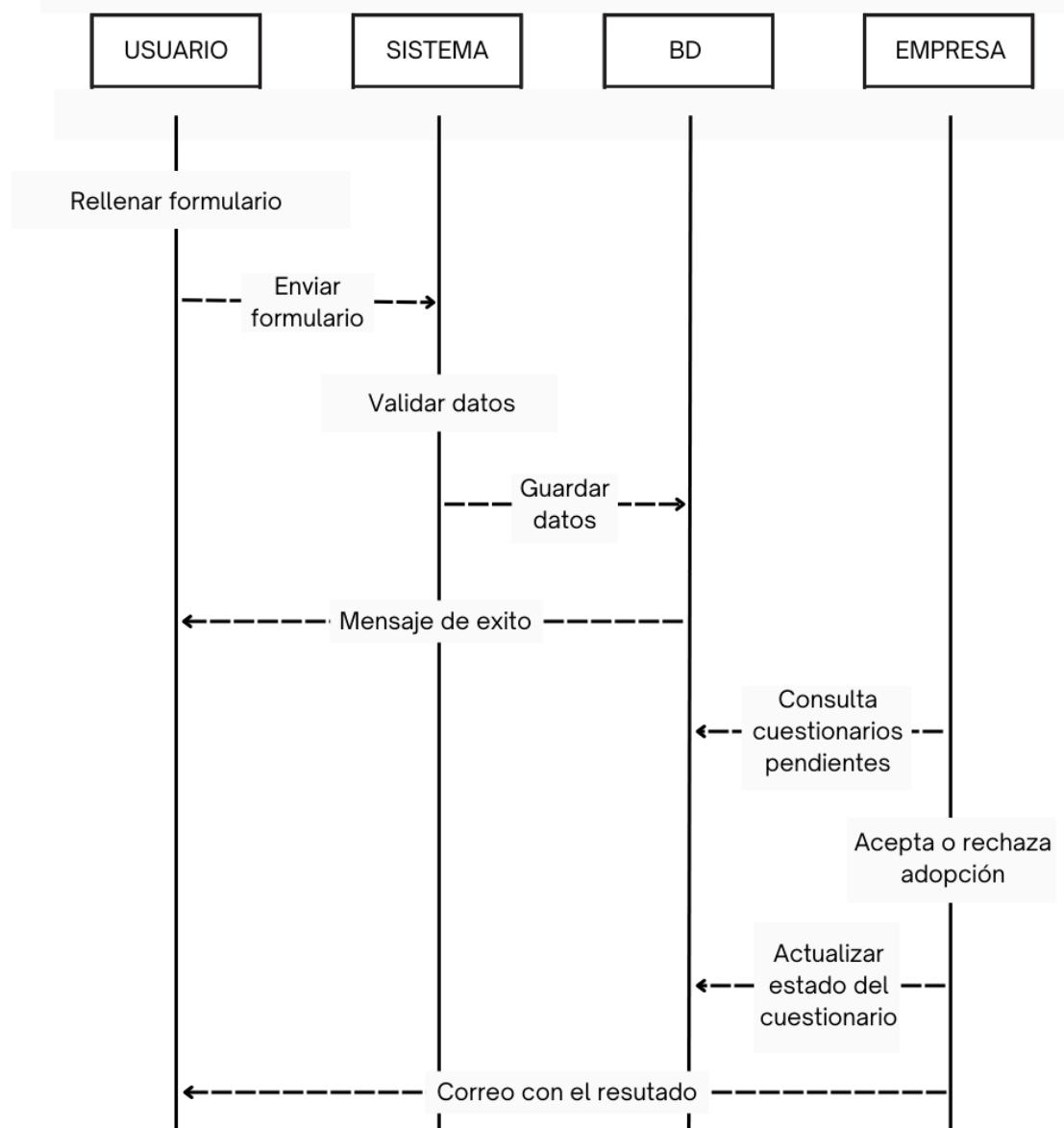
5. Diseño estático

5.1. Creación de diagramas

- Diagrama UML:



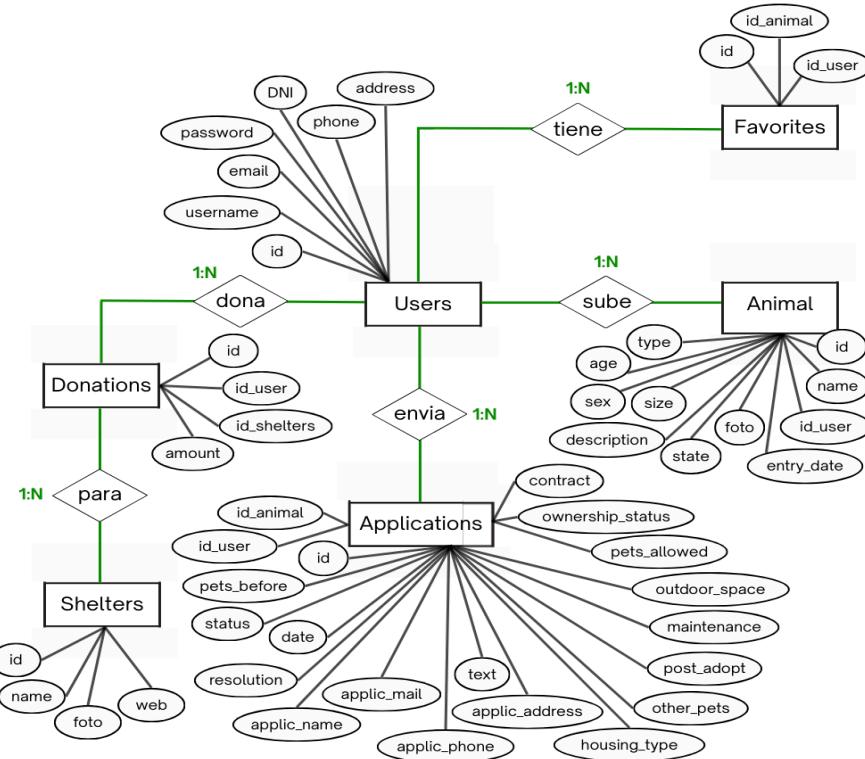
- Diagrama de secuencia del caso de "Enviar cuestionario de adopción":



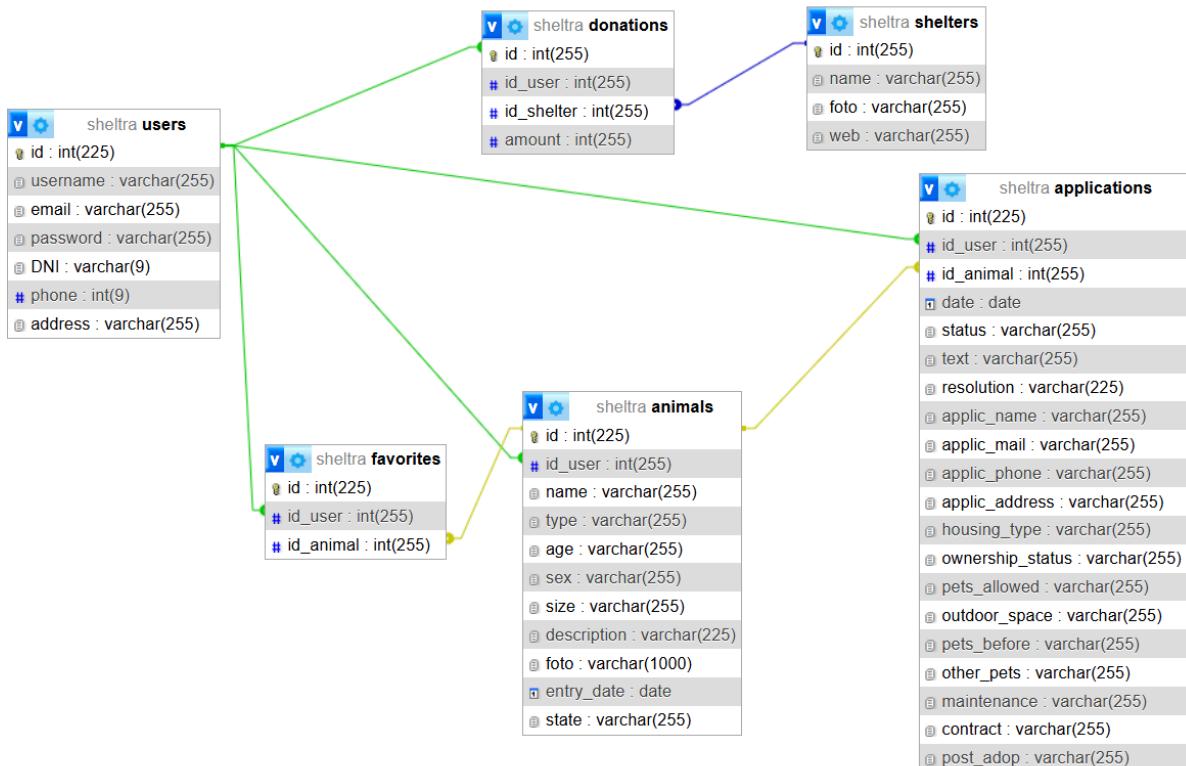
6. Modelo de datos

6.1. Definición de acceso a datos

- Diagrama E-R:



- Diagrama de tablas: que describen la organización de la base de datos, representando de forma básica la función de cada tabla



7. Manual

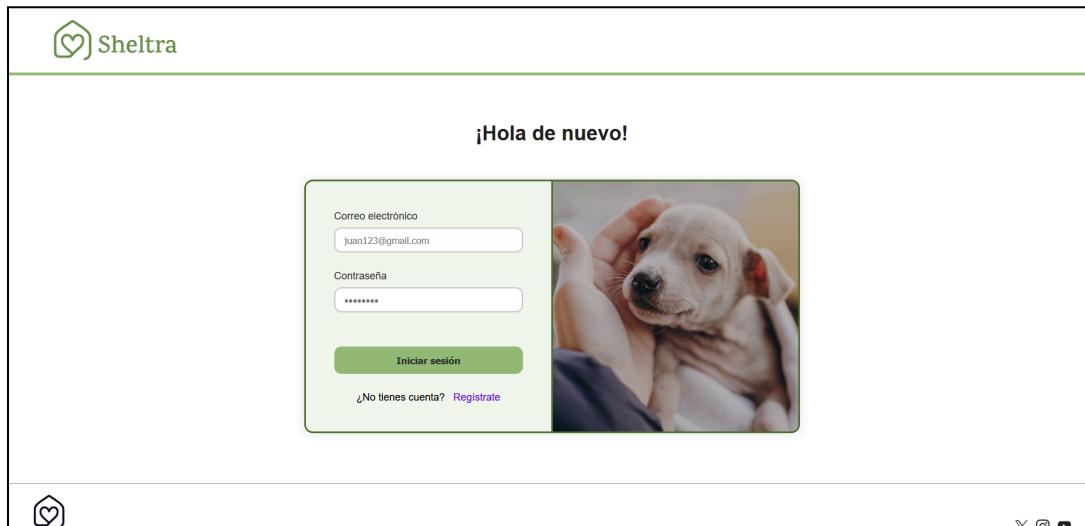
7.1. Creación de manuales de usuario e instalación

A continuación voy a presentar el **manual de usuario** de la aplicación, acompañado de un mapa de pantallas con capturas que muestran las funcionalidades principales. Este manual está diseñado para facilitar la navegación y comprensión de la interfaz, ayudando al usuario.

Mapa de pantallas (navegación):

Iniciar sesión: La pantalla de login permite al usuario acceder a su cuenta dentro de la aplicación. Consta de los siguientes elementos:

- **Campo de correo electrónico:** donde el usuario introduce el email con el que se registró.
- **Campo de contraseña:** donde debe escribir su contraseña de acceso.
- **Botón "Iniciar sesión":** al pulsarlo, el sistema verifica las credenciales y, si son correctas, redirige al usuario a la página principal.
- **Enlace "¿No tienes cuenta? Regístrate":** para acceder rápidamente al formulario de registro.



Registrarse: La pantalla de **registro** permite a nuevos usuarios crear una cuenta en la plataforma. Incluye los siguientes campos:

- Nombre del usuario
- Correo electrónico: debe ser único y válido.
- Teléfono: Debe ser un teléfono válido español.

- Contraseña: Debe tener mínimo 8 caracteres.
- Confirmación de contraseña: para evitar errores de escritura.
- DNI: Debe ser un DNI válido español.
- Dirección
- Botón "Registrarse": envía los datos al servidor y crea la cuenta si todo es correcto.

The screenshot shows the registration form for the Sheltra website. At the top, there's a green header with the 'Sheltra' logo. Below the header, a banner features the text '¡Bienvenido/a!' and a photograph of a woman holding a dog. The main form area contains several input fields: 'Nombre de usuario' (with 'Maria' entered), 'Teléfono' (with '111 11 11 11' entered), 'Correo electrónico' (with 'maria123@gmail.com' entered), 'Contraseña' (with '*****' entered), 'DNI/NIF' (with '1111111A' entered), 'Confirma contraseña' (with '*****' entered), and 'Dirección' (with 'Calle Ejemplo 123' entered). At the bottom of the form is a green 'Registrarme' button.

Pantalla de inicio: La pantalla de inicio consta de cuatro secciones, la primera es un banner donde hay un botón de "Adoptar ahora", este botón te redirigirá a una pantalla donde se encuentran todos los animales que hay disponibles para adoptar.

La siguiente sección muestra los tres últimos animales añadidos a la base de datos. La tercera es un apartado donde se pueden ver todos los tipos de animales que hay en la web, estos son gatos, perros, aves, peces, roedores y reptiles. Al clicar cualquier tipo de animales se redirigirá a una pantalla que mostrará todos los animales que haya del tipo escogido en la base de datos.

Por último está la sección de donar, si un usuario no puede permitirse adoptar a uno de nuestros animales pero aun así quiere contribuir a la causa, puede visitar esta pantalla donde se visualizan todos los refugios con los que colaboramos para aportar una donación económica.

A continuación muestro las pantallas:

Apartado de “Adoptar ahora”:

The screenshot shows the Sheltra website's adoption section. At the top left is the Sheltra logo (a green house with a heart inside). At the top right are social media sharing icons and the word "Vega". The main image is a close-up of a puppy being petted, with the text "Cada vida cuenta, desde el pico hasta la cola" overlaid. Below the image is a subtitle: "Nos dedicamos a dar una nueva oportunidad a cada animal, brindándoles un hogar lleno de amor y cuidado. Ayúdanos a transformar vidas a través de la adopción responsable." A "Adopta ahora" button is visible. At the bottom of the section, the text "Nuestras tres últimas incorporaciones" is displayed.

Apartado de “Tres últimas incorporaciones”:

The screenshot shows the Sheltra website's section featuring three recent animal admissions. The title "Nuestras tres últimas incorporaciones" is at the top. Three cards are shown, each with a photo and details: "Vainilla" (a rabbit), "Lucy" (a dog), and "Pedro" (a duck). Each card includes a "Ver más" button and a paw print icon.

Apartado de “Tipos de animales”:

The screenshot shows the Sheltra website's section on different types of animals. The title "¿Cuál es tu compañero ideal?" is at the top. Six categories are displayed in a grid: "Gatos" (cat), "Perros" (dog), "Aves" (bird), "Peces" (fish), "Reptiles" (reptile), and "Roedores" (rodent). Each category has a representative image and the name of the animal type.

Apartado de "Donaciones":



Todos los animales: Esta pantalla muestra un catálogo con todos los anuncios de adopción disponibles en la plataforma. Está diseñada para que los usuarios puedan explorar fácilmente a los animales en adopción.

Todos nuestros animales

Nube
Nube es un conejo juguetón y curioso. Le encanta explorar y...

Max
Max es un perro leal y protector. Le encanta jugar al aire libre y es...

Simba
Simba es un gato curioso y juguetón. Le encanta explorar y...

Animales por tipo: Permite a los usuarios visualizar únicamente los animales filtrados por su especie, facilitando así una búsqueda más específica según sus preferencias. Mejora la experiencia de navegación, sobre todo cuando hay una gran cantidad de animales publicados.

The screenshot shows a mobile application interface for animal adoption. At the top, there is a header with the Sheltra logo and a search bar. Below the header, there is a large image of several cats. Underneath the image, the word "Gato" is displayed. Below this, there are three smaller thumbnail images of cats, each with a green "Ver más" button below it.

Gato

Ver más

Gato

Simba
Tom
Mia

Ver más

Ver más

Ver más

Donaciones: Esta pantalla permite a los usuarios ver un listado de refugios colaboradores con el objetivo de realizar donaciones voluntarias que apoyen el cuidado de los animales si no se pueden permitir adoptar.

The screenshot shows a mobile application interface for donations. At the top, there is a header with the Sheltra logo and a search bar. Below the header, the word "DONACIONES" is displayed. A message below it reads: "Aunque no puedas adoptar, aún puedes cambiar vidas. Conoce a los refugios que lo dan todo por ellos." Below this, there is a large image of a person holding a cat's paw. At the bottom, the word "Refugios" is displayed, followed by three logos of合作机构: APADAN, Tepa Fundación, and FADA.

DONACIONES

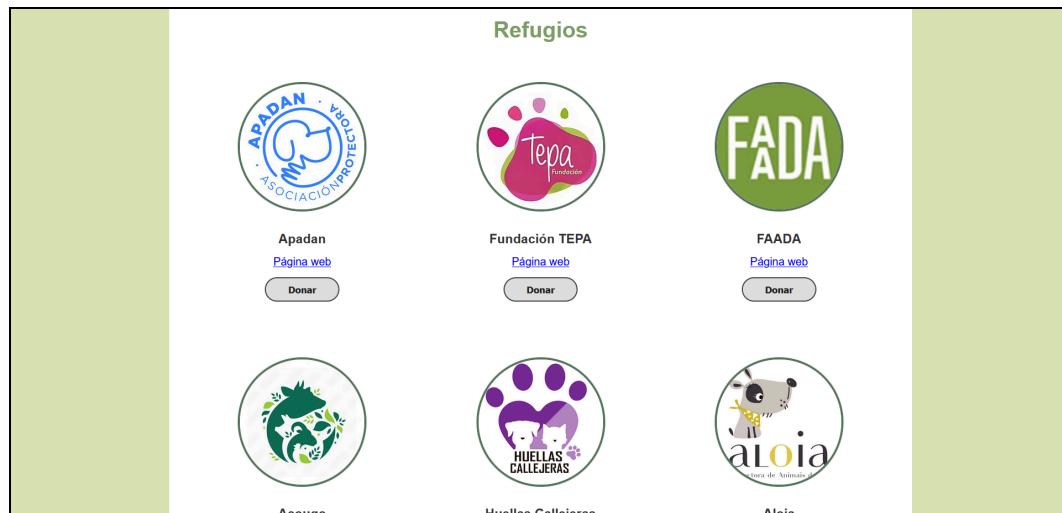
Aunque no puedas adoptar, aún puedes cambiar vidas.
Conoce a los refugios que lo dan todo por ellos.

Refugios

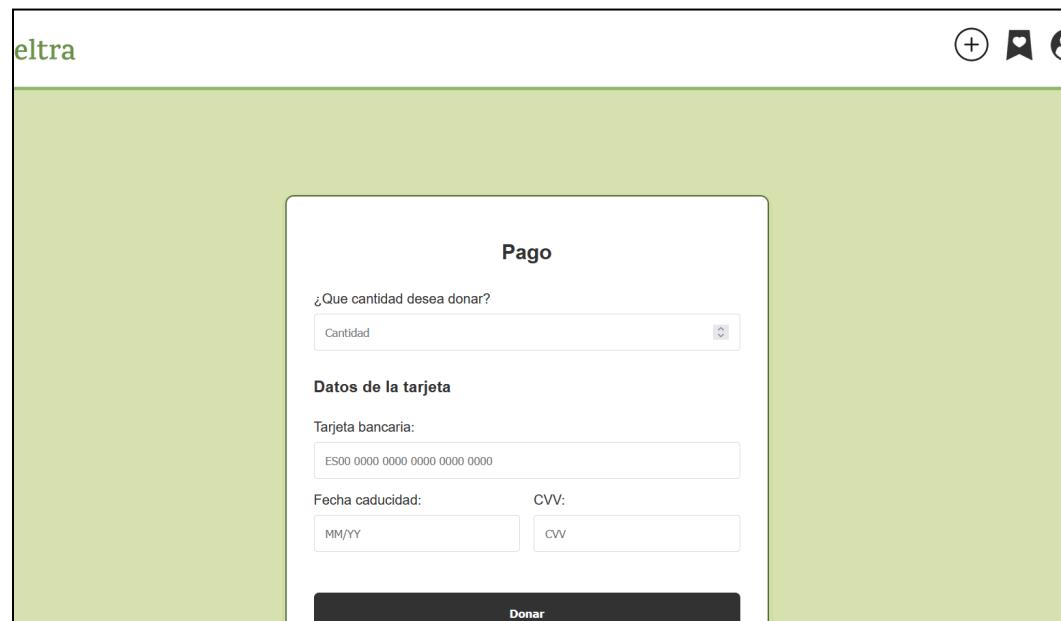
APADAN

Tepa Fundación

FADA



Pantalla de pago de las donaciones: Esta pantalla aparece cuando un usuario selecciona un refugio y decide realizar una donación. Su objetivo es permitir al usuario introducir los datos necesarios para completar el pago de forma sencilla y segura.



Ficha del animal: Esta pantalla muestra toda la información relacionada con un animal en concreto que está disponible para adopción. Es el lugar donde los usuarios pueden consultar datos detallados y decidir si desean enviar el formulario de adopción.

En la parte de abajo hay un carrusel de animales del mismo tipo por si el usuario busca otra cosa.

The screenshot shows a profile for a rabbit named "Vainilla". The main image is a close-up of a light brown lop-eared rabbit sitting outdoors. To the right of the image is a detailed profile card:

- Vainilla** (name)
- Calle Vega, 33 (address)
- Vainilla es una conejita a la que le gusta comer y dormir, perfecta para familias tranquilas (description)
- Información adicional:**
- Edad: 2
- Tamaño: Pequeño
- Sexo: Hembra
- Fecha ingreso: 2025-05-15
- A green "Adoptar" button at the bottom.

Below this, there is a section titled "También te pueden interesar:" (Also might interest you:) featuring three more small animal profiles:

- Nube**: Roedor (Rodent). Image shows a brown and white rabbit-like rodent.
- Chester**: Roedor (Rodent). Image shows a guinea pig.
- Luna**: Roedor (Rodent). Image shows a hamster.

Each animal profile includes a "Ver más" button and a small paw icon.

Cuestionario de adopción: Esta pantalla permite a los usuarios interesados en adoptar un animal enviar una solicitud formal respondiendo un formulario con preguntas que nos ayudarán a evaluar si el usuario es un buen perfil como adoptante.

Una vez el usuario envía un formulario el estado del animal pasa de estar a "En adopción" a "No en adopción" y ya no se muestra visible en la web.

 Sheltra

+ 📌 🎯 🌐 Vega



Cuestionario de solicitud de adopción

Cada animalito necesita una segunda oportunidad.

Cuéntanos un poco sobre ti y el hogar que estás dispuesto a ofrecer.

Ellos solo necesitan amor... ¿Lo tienes tú?

Datos personales

Nombre completo
Lorena Gómez López

Correo electrónico lorena123@gmail.com	Teléfono 111 11 11 11
Dirección Calle Ejemplo, N°3	

Sobre tu hogar

Tipo de vivienda:
 Casa Piso Otro

¿Es vivienda propia o alquilada?
 Propia

¿Permiten mascotas? (en caso de alquiler)
 Sí No

¿Tienes patio o jardín?
 Sí No

Compromiso

¿Estás dispuesto/a a asumir los cuidados?
 Sí No

¿Estás de acuerdo en firmar un contrato de adopción?
 Sí No

¿Aceptarias seguimiento post-adopción si es necesario?
 Sí No

Enviar

Subir un animal: Esta pantalla permite a los usuarios crear un nuevo anuncio de adopción para que otras personas puedan ver al animal.

Subir animal

Nombre	Edad
Bigotes	3 meses
Sexo	Tamaño
Macho	Pequeño
Tipo	Fecha ingreso
Gato	dd / mm / aaaa
Foto	URL de la imagen
Estado de la adopción	
Adopción activa	
Descripción del animal	
Bigotes es un gato tranquilo y le encanta dormir...	
Subir animal	

© 2026 Sheltra. Todos los derechos reservados

Favoritos: Esta pantalla permite a los usuarios registrados ver los animales que han marcado como favoritos, facilitando el seguimiento de aquellos en los que están interesados para una posible adopción futura.

Tus favoritos

	Max Perro	
Max es un perro leal y protector. Le encanta jugar al aire libre y es perfecto para familias activas que buscan un compañero energético.		
Ver más		
	Coco Gato	
Coco es juguetón y se lleva bien con perros pequeños.		
Ver más		

Editar perfil: Esta pantalla permite al usuario gestionar su cuenta personal, actualizando su información básica, y accediendo a opciones clave como cerrar sesión o eliminar la cuenta.

The screenshot shows the 'Editar perfil' (Edit Profile) screen of the Sheltra application. At the top, there is a header bar with the Sheltra logo on the left and a user icon labeled 'Vega' on the right. Below the header, the title 'Editar perfil' is centered. The main content area contains a form with the following fields:

- Nombre de usuario: Vega
- Correo electrónico: juan@gmail.com
- Contraseña: (redacted)
- Confirmar contraseña: (redacted)
- DNI/NIF: 11111111A
- Teléfono: 111 11 11 11
- Dirección: Calle ejemplo, 123

At the bottom of the form, there are three buttons: a green 'Guardar' (Save) button, a grey 'Cerrar sesión' (Logout) button, and a red 'Eliminar cuenta' (Delete account) button. The footer of the page contains the copyright notice: '© 2025 Sheltra. Todos los derechos reservados'.

Manual de Instalación

Ahora se detallarán los requisitos del sistema, la configuración del entorno de desarrollo, la instalación de dependencias necesarias y los pasos para ejecutar correctamente la aplicación tanto a nivel local como en un servidor.

- Requisitos del sistema:

- Servidor local como XAMPP, WAMP o MAMP (con Apache y MySQL).
- PHP 7.4 o superior

- MySQL/MariaDB como gestor de base de datos.
- Navegador web actualizado (Google Chrome, Firefox, etc.)
- Editor de código recomendado: Visual Studio Code.

- **Pasos de instalación:**

1. Descargar o clonar el proyecto: Clonar el repositorio con el comando **git clone git@github.com:ElbaCalvo/Sheltra.git**

2. Colocar el proyecto en el servidor local: Mueve la carpeta del proyecto a la ruta correspondiente:

- En XAMPP → htdocs/
- En WAMP → www/
- En MAMP → htdocs/

Por ejemplo: C:/xampp/htdocs/sheltra/

3. Instalar Composer: Para comprobar si Composer está instalado, abre la terminal y ejecuta **composer -- version**

Si no está instalado, descárgalo desde <https://getcomposer.org/>

4. Instalar dependencias del proyecto: Una vez dentro de la carpeta raíz del proyecto (donde está el archivo composer.json), ejecuta en la terminal el comando **composer install**. Esto descargará las dependencias necesarias en la carpeta vendor/

5. Configurar la base de datos: Inicia **phpMyAdmin** desde el panel de control de tu servidor local y crea una nueva base de datos, por ejemplo:

Nombre: sheltra

Collation: utf8mb4_unicode_ci

Después de crearla, importa el archivo .sql que se encuentra en la carpeta del proyecto con la estructura y datos iniciales, usa la opción "Importar" de phpMyAdmin y selecciona el archivo.

6. Configurar la conexión a la base de datos: Abre el archivo de configuración y asegúrate de que los datos coincidan con tu servidor local:

```
$host = "localhost";
$user = "root";
$password = "";
$dbname = "sheltra";
```

7. Iniciar Apache y MySQL: Desde el panel de XAMPP (o

WAMP/MAMP), activa Apache y MySQL y accede al navegador. Después entra a la dirección **<http://localhost/Sheltra/app/view/HomeScreen.php>**, la aplicación debería cargarse correctamente en el navegador.

8. Viabilidad

8.1. Estudio de mercado

El abandono animal y la saturación de los refugios siguen siendo problemas persistentes en muchas regiones. A pesar de la existencia de algunas plataformas que promueven la adopción, muchas de ellas están vinculadas a refugios específicos o tienen procesos poco intuitivos. Sheltra detecta una oportunidad clara al ofrecer una solución digital independiente, transparente y accesible para facilitar la adopción.

La demanda identificada incluye:

- Personas que no pueden seguir cuidando a un animal y necesitan encontrarle un nuevo hogar responsable.
- Adoptantes que buscan una forma sencilla y segura de adoptar.
- Usuarios sensibilizados con la protección animal que quieren colaborar económicamente sin adoptar.
- Refugios y protectoras que necesitan un canal adicional para dar visibilidad a los animales bajo su cuidado.

El mercado objetivo de Sheltra se centra principalmente en personas de entre 18 y 45 años, ya que este grupo suele tener una mayor familiaridad con el uso de herramientas digitales y plataformas online. Además, dentro de este rango de edad es común encontrar perfiles tanto de personas jóvenes interesadas en adoptar por primera vez, como de adultos que ya han tenido experiencia previa con animales.

La mayoría de estos usuarios residen en zonas urbanas, donde las restricciones de espacio y las condiciones de vida hacen que el proceso de adopción responsable requiera un canal claro, accesible y seguro como el que ofrece Sheltra.

También se tiene en cuenta que este grupo suele estar más sensibilizado con temas sociales y medioambientales, lo que los hace más receptivos a iniciativas con enfoque ético y sin ánimo de lucro.

8.2. Análisis económico

Para evaluar la viabilidad económica de Sheltra, se ha hecho un estudio que tiene en cuenta los recursos necesarios para su puesta en marcha y su mantenimiento básico. La naturaleza sin ánimo de lucro del proyecto no implica ausencia de ingresos, sino que estos se reinvierten íntegramente en el mantenimiento, mejora y expansión del servicio.

En cuanto a **recursos humanos**, el equipo está compuesto inicialmente por cinco personas contratadas a tiempo parcial: una persona responsable de la dirección del proyecto, un desarrollador web (encargado de la programación y mantenimiento de la plataforma), un responsable de comunicación y redes, una persona en el departamento de soporte y una persona encargada del área legal y ética. La idea es mantener una estructura compacta pero funcional, centrada en garantizar el correcto funcionamiento de la plataforma y la atención a los usuarios.

A nivel de **recursos temporales**, el desarrollo inicial de la plataforma se ha estimado en aproximadamente seis meses, incluyendo fases de análisis, diseño, programación, pruebas y puesta en producción. Posteriormente, se prevé un mantenimiento continuo y mejoras progresivas a medida que se amplíe la base de usuarios.

Respecto a los **costes económicos**, se han contemplado los siguientes gastos aproximados para el primer año de funcionamiento:

- Dominio y alojamiento web: 120€ anuales.
- Servicios externos (correo, seguridad, backups): 180€.
- Sueldos (5 personas a media jornada durante 12 meses): 60.000€.
- Campañas de difusión y publicidad: 2.000€.
- Gastos legales y administrativos: 700€.
- Imprevistos y margen operativo: 1.000€.

Total estimado: 52.000€ anuales.

Para hacer frente a estos costes, Sheltra contempla distintas vías de **financiación**:

- Donaciones particulares, tanto puntuales como recurrentes.
- Subvenciones públicas destinadas a entidades sin ánimo de lucro del ámbito animalista o digital.
- Programas de ayudas al emprendimiento social o digital (ayuntamientos, diputaciones o entidades europeas).
- Convenios con protectoras y asociaciones que deseen usar la plataforma como canal de difusión.
- Posibilidad de establecer colaboraciones con empresas éticas que quieran patrocinar o apoyar la iniciativa.

8.3. Estudio de riesgos

En el proceso de preparar el inicio de este proyecto, han surgido varios riesgos que conviene tener en cuenta. Algunos de ellos podrían influir tanto en el lanzamiento como en el funcionamiento diario de la plataforma. Por eso, se han pensado posibles soluciones o medidas para prevenirlos y actuar a tiempo si se presentan. Aquí explico los principales:

Uno de los riesgos más evidentes en las primeras fases del proyecto es la

falta de usuarios activos, tanto personas que publiquen animales en adopción como adoptantes. Esto puede deberse a un desconocimiento de la plataforma o a la desconfianza hacia un servicio nuevo. Para solucionarlo, se contempla una estrategia de difusión a través de redes sociales, colaboraciones con refugios y campañas de concienciación que destaque el carácter responsable, ético y gratuito del servicio.

Otro riesgo relevante es el **mal uso de la plataforma** por parte de algunos usuarios, como la publicación de animales falso, fraudulento o inadecuado. Para evitar este tipo de situaciones, en la plataforma se revisan y controlan tanto las cuentas de los usuarios como los anuncios que publican, así como un protocolo de moderación que permita detectar y eliminar rápidamente cualquier uso inapropiado.

También se ha considerado el posible **incumplimiento de la normativa legal** en materia de adopciones de animales, especialmente cuando se trata de acuerdos entre particulares. Para minimizar este riesgo, se ha trabajado en la redacción de unos términos y condiciones claros, alineados con la legislación vigente, y se cuenta con asesoramiento legal externo que garantice la seguridad de todas las partes implicadas.

Desde el punto de vista técnico, pueden surgir **problemas relacionados con la infraestructura**, como caídas del servidor, errores en el sistema o vulnerabilidades de seguridad. Para prevenirlas, se ha optado por un entorno estable de desarrollo y alojamiento, realizando copias de seguridad frecuentes y aplicando medidas básicas de protección de datos, además de pruebas de funcionamiento.

Otro riesgo importante es la **falta de financiación** a medio y largo plazo, especialmente tratándose de una entidad sin ánimo de lucro. El mantenimiento del servicio, la ampliación de funcionalidades y la contratación de personal pueden verse limitados si no se cuenta con recursos económicos suficientes. Como solución, Sheltra apuesta por una combinación de fuentes de ingresos, incluyendo donaciones directas de usuarios, convenios de colaboración con entidades del mismo ámbito y la solicitud de subvenciones.

Adicionalmente, existe el riesgo de **falta de confianza por parte de los usuarios**, sobre todo en cuanto a la protección de sus datos personales y la seriedad de los procesos de adopción. Para responder a esta preocupación, la plataforma incluye políticas de privacidad transparentes y una comunicación clara que transmita profesionalidad y compromiso ético.

9. Calidad

9.1. Pruebas

Prueba unitaria:

En la carpeta test creo un archivo llamado UserTest.php donde haré los test necesarios para los métodos del Usuario, en este caso, haré la prueba unitaria para el método addUser() de UserController.php.

Lo que hago ahora es un test que compruebe que el método addUser() de la clase UserController() funcione correctamente. Primero creo una instancia del

controlador y luego llamo al método addUser(), pasando los datos del usuario de prueba como parámetros. Este método crea una instancia del modelo User, establece sus propiedades y llama al método addUser() del modelo. Al mismo tiempo, con el assertTrue() compruebo que el método addUser() devuelva true, lo que significa que la creación de usuario fue exitosa:

```
tests > UserTest.php > UserTest
1  <?php
2
3  use PHPUnit\Event\Code\Test;
4  use PHPUnit\Framework\TestCase;
5  require_once __DIR__ . '/../app/controller/UserController.php';
6
7  class UserTest extends TestCase {
8      public function testCreateUser() {
9          $controller = new UserController();
10         $response = $controller->addUser("Juan", "juan@example.com",
11             "12345678", "12345678L", "123456789", "Calle Juan, 123");
12         $this->assertTrue($response);
13     }
14 }
```

Resultado del test:

```
PS C:\xampp\htdocs\sheltra> vendor\bin\phpunit tests
PHPUnit 11.5.21 by Sebastian Bergmann and contributors.

Runtime:      PHP 8.2.12
.
.
.
Time: 00:00.075, Memory: 8.00 MB
OK (1 test, 1 assertion)
```

Prueba con Mocks:

Creamos el método testLoginSuccess() que será el test mock que comprobará que el método login() de la clase userController funcione correctamente. Primero creo un Mock de la clase User y le digo que compruebe que el método login() siempre devuelva true.

Luego creo una instancia del UserController() y llamo al método login() del controlador pasándole los parámetros de prueba y el mock de User. En el método login() de userController tuve que añadir un if para comprobar si se le pasa o no el parámetro \$user, si se le pasa es que se está ejecutando el test y se le están pasando los datos de prueba, si no, el método funciona igual que antes, comprobando al mismo tiempo con assertTrue() si el usuario existe.

Prueba mock:

```

public function testLoginSuccess() {
    $email = 'vega@gmail.com';
    $password = '12345678';

    $userMock = $this->createMock(User::class);

    $userMock->expects($this->once())->method('setEmail')->with($email);
    $userMock->expects($this->once())->method('setPassword')->with($password);
    $userMock->expects($this->once())->method('checkUsuario')->willReturn(true);
    $userMock->expects($this->once())->method('getId')->willReturn(2);
    $userMock->expects($this->once())->method('getUsername')->willReturn('Vega');

    $controller = new UserController($userMock);

    $result = $controller->login($email, $password, $userMock);

    $this->assertIsArray($result);
    $this->assertEquals(2, $result['id']);
    $this->assertEquals('Vega', $result['username']);
}

```

Resultado del test:

```

PS C:\xampp\htdocs\Sheltra> vendor\bin\phpunit tests
PHPUnit 11.5.21 by Sebastian Bergmann and contributors.

Runtime:      PHP 8.2.12

..
2 / 2 (100%)

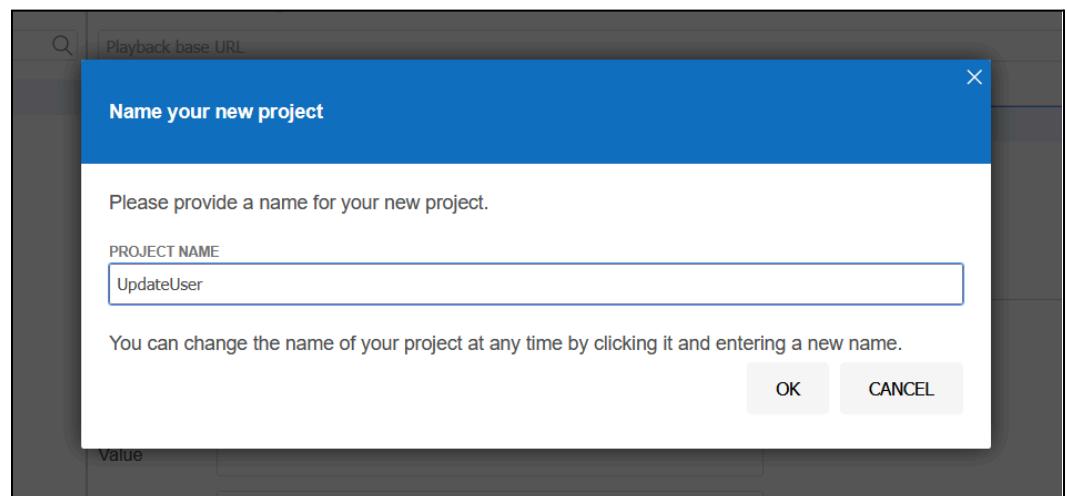
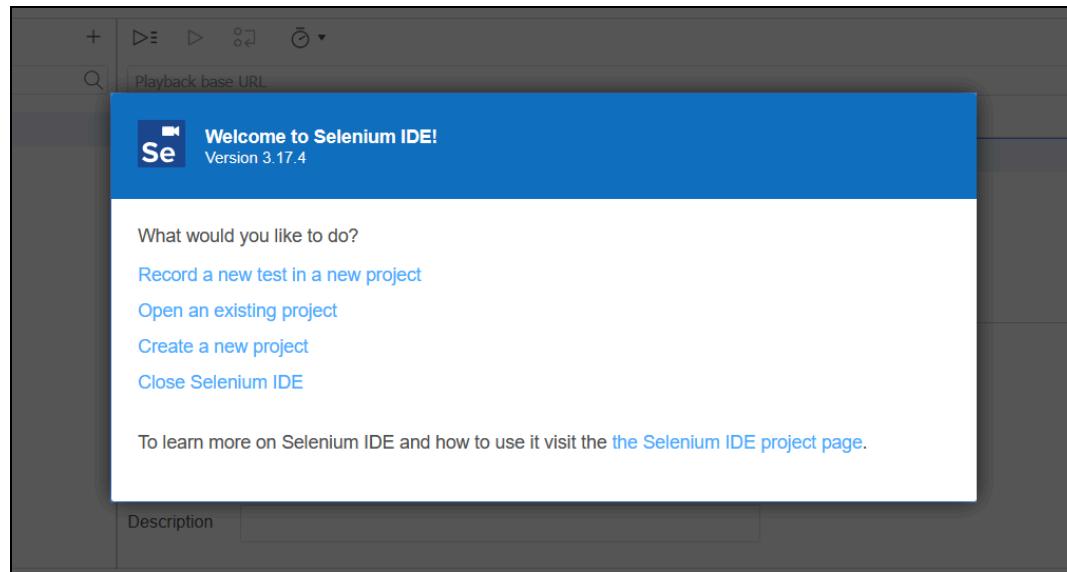
Time: 00:00.088, Memory: 8.00 MB

OK (2 tests, 7 assertions)

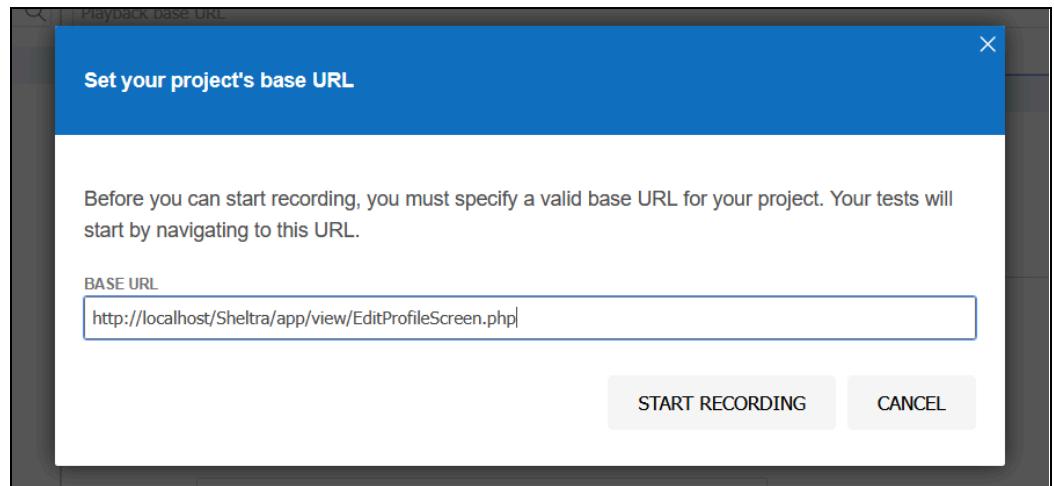
```

Prueba Funcional:

Para hacer las pruebas funcionales primero tuve que instalar la extensión de Selenium IDE. Cuando ya lo tuve instalado, tenemos que dar a "Create a new project", y después le tenemos que poner un nombre, en este caso UpdateUser, que va a ser el método que voy a hacer la prueba:



Cuando ya tenemos el proyecto creado y le damos a "Start recording" en la esquina superior derecha, pregunta la URL. Cogí la URL de la pantalla de editar perfil de Sheltra:



Antes de empezar con la prueba, compruebo que valores tiene el usuario de Juan (que va a ser el que use) en la base de datos:

orrar	9	Ana	ana@gmail.com	\$2y\$10\$D3RQHaWyK5qRMxN4QQEOmX2PS4ySiyv6lMPri12vC...	44444444M	444444444	Calle Ana, 22
orrar	12	Julia	julia@gmail.com	\$2y\$10\$LNhz7ke.KBMB/MfHGPsP.UMG0M2kLGpO4X79oSnhph...	55555555U	555555555	Calle julia, 55
orrar	13	Beth	beth@gmail.com	\$2y\$10\$mZfoxmvzEupBdKkFocJmyuMsakeM/V6q3y2Q1J02X/...	22222222X	222222222	Calle Beth, 22
orrar	14	Juan	juan@example.com	\$2y\$10\$hoEBbmDxZLJE23DaGejaO/JCl.zxKjhzznHVBLBP/...	12345678L	123456789	Calle Juan, 123

Ahora como le di a "Start recording" puedo empezar a llenar los campos con los datos nuevos de Juan:

DNI/NIF	
11111111A	
Teléfono	Dirección
111 11 11 11	Calle ejemplo, 123
 ● Selenium IDE is recording...	
Guardar	

Selenium debería haber guardado todos los pasos que hice al llenar los campos al editar el perfil de usuario:

File  Edit  View  Insert  Tools  Help 

http://localhost

Command	Target	Value
3	click	name=username
4	type	name=username
5	click	name=email
6	type	name=email
7	click	name=password
8	type	name=password
9	click	name=confirm_password
10	type	name=confirm_password
11	click	name=dni
12	type	name=dni
13	click	name=phone
14	type	name=phone
15	click	name=address
16	type	name=address
17	click	css=.save-button

Le damos a "Stop recording" y luego a "Run current test" para comprobar que todo va bien

	Command	Target	Value
3	✓ click	name=username	
4	✓ type	name=username	Lupe
5	✓ click	name=email	
6	✓ type	name=email	lupe@gmail.com
7	✓ click	name=password	
8	✓ type	name=password	12345678
9	✓ click	name=confirm_password	
10	✓ type	name=confirm_password	12345678
11	✓ click	name=dni	
12	✓ type	name=dni	98765432M
13	✓ click	name=phone	
14	✓ type	name=phone	987 65 43 21
15	✓ click	name=address	
16	✓ type	name=address	Calle Lupe, 321
17	✓ click	css= save-button	

Después de comprobar que todo ha ido bien voy a provocar un fallo para ver si tambien funciona, en este caso cambiaré el valor de confirm_password para que no coincida con la contraseña:

Command	type	//	✖
Target	name=confirm_password	✖	🔍
Value	11		
Description			

En el caso de nuestra web, el test no dará Error porque el código está hecho para que en vez de lanzar un error si el usuario rellena algo mal, salte un alert avisando (obviamente sin actualizarlo en la base de datos):

	4	✓ type	name=username	Lupe
	5	✓ click	name=email	
	6	✓ type	name=email	lupe@gmail.com
	7	✓ click	name=password	
	8	✓ type	name=password	12345678
	9	✓ click	name=confirm_password	
	10	✓ type	name=confirm_password	11
	11	✓ click	name=dni	
	12	✓ type	name=dni	98765432M
	13	✓ click	name=phone	
	14	✓ type	name=phone	087 65 43 21

10. Conclusiones y mejoras futuras

Sheltra cumple con los objetivos planteados al inicio del proyecto, permitiendo a los usuarios ver animales en adopción, enviar cuestionarios, colaborar mediante donaciones y gestionar su cuenta de forma sencilla.

Mejoras futuras que podrían implementarse:

- Permitir subir más imágenes por animal.
- Incluir filtros avanzados en la búsqueda (por raza, tamaño, edad...).
- Integración de pasarelas de pago reales para donaciones.
- Implementar un sistema de comentarios o valoraciones para refugios.
- Historial de adopciones.
- Sistema de seguimiento post-adopción.
- Versión móvil App

10.1. Conclusiones

El desarrollo de Sheltra ha permitido aplicar de forma práctica conocimientos de diseño web, bases de datos, arquitectura MVC y gestión de proyectos. Se ha diseñado una plataforma funcional, con una interfaz amigable y un enfoque claro en la adopción responsable de animales.

El trabajo ha fomentado la planificación, el diseño centrado en el usuario y la mejora continua a través de pruebas y correcciones. A pesar de las limitaciones de tiempo y alcance, se ha conseguido construir una base sólida que puede ser ampliada y mantenida en el futuro.

11. Bibliografía

11.1. Bibliografía

- Infopropiedadintelectual.es. (s.f.). *Regulaciones para la adopción de animales en territorio español*. Recuperado de:
<https://infopropiedadintelectual.es/regulaciones-para-la-adopcion-de-animales-en-territorio-espanol/>
- NoticiasTrabajo. (2023). *Requisitos para adoptar un perro o gato según la nueva Ley de Bienestar Animal*. Recuperado de:
<https://noticiastrabajo.huffingtonpost.es/sociedad/requisitos-adoptar-perro-o-gato-nueva-ley-bienestar-animal/>
- Legalondo. (s.f.). *¿Cuáles son los requisitos legales para adoptar una mascota en España?* Recuperado de:
<https://www.legalondo.com/cuales-son-los-requisitos-legales-para-adoptar-una-mascota-en-espana/>