

UNIVERSIDAD MIGUEL HERNÁNDEZ DE ELCHE

ESCUELA POLITÉCNICA SUPERIOR DE ELCHE

GRADO EN INGENIERÍA INFORMÁTICA EN
TECNOLOGÍAS DE LA INFORMACIÓN



"DISEÑO E IMPLEMENTACIÓN DE UNA
APLICACIÓN WEB PARA GESTIÓN DE
ADOPCIÓN DE MASCOTAS
GESTIONADAS POR ALBERGUES,
PROTECTORAS Y PARTICULARES"

TRABAJO FIN DE GRADO

Junio - 2023

AUTOR: Alberto Aguilar Domingo
DIRECTOR: Jesús Javier Rodríguez Sala

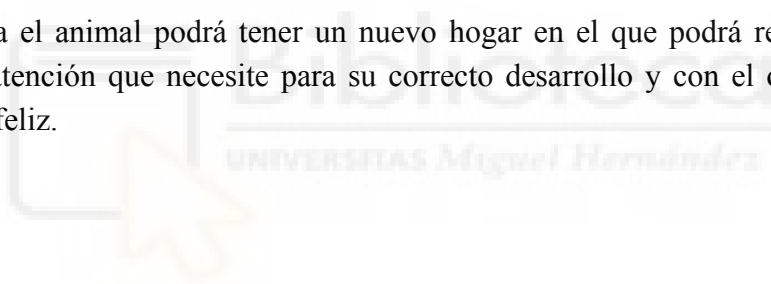
RESUMEN

En el presente documento se hará una descripción del trabajo realizado por el alumno Alberto Aguilar Domingo para su trabajo de final de grado en la carrera de Ingeniería Informática en Tecnologías de la Información de la Universidad Miguel Hernández de Elche.

El proyecto presentado se trata de una aplicación web destinada a facilitar la adopción de animales que se encuentren en búsqueda de un hogar. La aplicación permitirá que tanto usuarios particulares como distintos refugios o protectoras puedan darse de alta y publicar anuncios sobre aquellos animales que se encuentren en una situación vulnerable o en la búsqueda de un hogar.

A través de dichos anuncios, las personas interesadas podrán ponerse en contacto con el propietario del animal para poder formalizar el proceso de adopción del mismo. Dicho proceso se realizará dentro de la propia página para poder dar un entorno seguro al proceso de adopción.

De esta manera el animal podrá tener un nuevo hogar en el que podrá recibir todos los cuidados y la atención que necesite para su correcto desarrollo y con el objetivo de que tenga una vida feliz.



ÍNDICE GENERAL

Capítulo 1. Introducción	10
1.1.- LA ADOPCIÓN DE ANIMALES Y MASCOTAS	10
1.1.1.- Comercialización de animales	10
1.1.2.- Protectoras de animales	11
1.1.3. Asociaciones de animales	12
1.1.4. Gestión de animales a nivel local	12
1.1.5. Legislación actual	13
1.2.- JUSTIFICACIÓN DEL PROYECTO	14
1.3.- OBJETIVOS	14
Capítulo 2. Antecedentes y estado de la cuestión	16
2.1.- ADOPCIÓN DE ANIMALES ACTUALMENTE	16
2.2.- CÓMO ADOPTAR UN ANIMAL ACTUALMENTE	17
2.2.1.- Protectora Baix Vinalopó	17
2.2.2.- Sociedad Protectora para Animales de Torrevieja (SAT)	18
2.2.3.- FAADA	20
2.2.4.- Milanuncios	21
2.2.5.- Resumen	22
2.3.- VALORACIÓN	23
Capítulo 3. Hipótesis de trabajo	24
3.1.- LENGUAJES DE PROGRAMACIÓN	24
3.1.1.- HTML	24
3.1.2.- PHP	25
3.1.3.- JavaScript	26
3.1.4.- CSS	26
3.1.5.- SQL	27
3.2.- METODOLOGÍAS DE DISEÑO E IMPLEMENTACIÓN	27
3.2.1.- Modelo Vista Controlador (MVC)	27
3.2.2.- AJAX	28
3.3.- BASES DE DATOS MySQL	29
3.4.- HERRAMIENTAS DE DESARROLLO	29
3.4.1.- Visual Studio Code	29
3.4.2.- XAMPP	30
3.4.3.- phpMyAdmin	31
3.5.- LIBRERÍAS UTILIZADAS	32
3.5.1.- PHPMailer	32

3.5.2. - Librerías "utilidades.js" y "utilidades.php"	32
3.6.- REPOSITARIOS DE IMÁGENES GRATUITAS	33
Capítulo 4. Metodología y resultados	34
4.1.- PLANIFICACIÓN DEL PROYECTO	34
4.1.1.- Diagrama de Gantt	35
4.2.- CAPTURA DE REQUISITOS	36
4.3.- DISEÑO	40
4.3.1 Diseño de la base de datos	40
4.3.2 Diseño de la interfaz gráfica	42
4.4.- IMPLEMENTACIÓN	44
4.4.1. Resultado final de la aplicación	44
4.4.2. Peculiaridades de la aplicación	46
Capítulo 5. Conclusiones y trabajo futuro	49
5.1.- CONCLUSIONES	49
5.2.- POSIBLES DESARROLLOS FUTUROS	50
Bibliografía	51
Anexo 1 Descripción de los casos de uso	55
Anexo 2. Diseño de la base de datos	71

ÍNDICE DE TABLAS

Tabla 3.1 - Repositorios de imágenes	33
Tabla 4.1 - Requisitos Página Principal	36
Tabla 4.2 - Requisitos Registro de Usuarios	36
Tabla 4.3 - Requisitos del Registro de Animales	36
Tabla 4.4 - Requisitos de la Visualización de Animales	36
Tabla 4.5 - Requisitos del Inicio de sesión	36
Tabla 4.6 - Requisitos de la visualización del perfil de usuario	37
Tabla 4.7 - Requisitos de la comunicación entre usuarios	37
Tabla 4.8 - Requisitos de la detección de errores	37
Tabla 4.9 - Requisitos de la base de datos	37
Tabla 4.10 - Casos de uso del actor Administrador	38
Tabla 4.11 - Casos de uso del actor Usuario no Registrado	38
Tabla 4.12 - Casos de uso del actor Usuario Registrado	39
Tabla 4.13 - Caso de uso N°1	40
Tabla 4.14 - Diseño tabla Animal	42
Tabla A1.1 - CU1: Acceso a la página web	56
Tabla A1.2 - CU2: Registro Particular	57
Tabla A1.3 - CU3: Registro Refugio	58
Tabla A1.4 - CU4: Iniciar Sesión	59
Tabla A1.5 - CU5: Recuperar Contraseña	60
Tabla A1.6 - CU6: Registro de animales	61
Tabla A1.7 - CU7: Ver Perfil	62
Tabla A1.8 - CU8: Eliminar animales	63
Tabla A1.9 - CU9: Registrar Adopción	64
Tabla A1.10 - CU10: Visualización de animales	65
Tabla A1.11 - CU11: Eliminar cuenta	66
Tabla A1.12 - CU12: Crear Chat	67
Tabla A1.13 - CU13: Visualizar Chats	68
Tabla A1.14 - CU14: Cerrar sesión	69
Tabla A1.15 - CU15: Contactar con el administrador	70
Tabla A2.1 - Diseño tabla Animal	72
Tabla A2.2 - Diseño tabla Especie	72
Tabla A2.3 - Diseño tabla Raza	73
Tabla A2.4 - Diseño tabla Comunidad Autónoma	73
Tabla A2.5 - Diseño tabla Provincia	73
Tabla A2.6 - Diseño tabla Municipio	74
Tabla A2.7 - Diseño tabla Usuario	74
Tabla A2.8 - Diseño tabla Particular	75

Tabla A2.9 - Diseño tabla Refugio	75
Tabla A2.10 - Diseño tabla Veterinario	76
Tabla A2.11 - Diseño tabla Fotos	76
Tabla A2.12 - Diseño tabla Chat	77
Tabla A2.13 - Diseño tabla Mensaje	78
Tabla A2.14 - Diseño tabla Adopción	79



ÍNDICE DE FIGURAS

Figura 2.1 - Portal “Protectora Baix Vinalopó”	18
Figura 2.2 - Ejemplo de ficha de un animal en adopción	18
Figura 2.3 - Sociedad Protectora para Animales de Torrevieja	19
Figura 2.4 - Ejemplo de ficha de Animal en adopción en SAT	19
Figura 2.5 - Página Principal de FAADA	20
Figura 2.6 - Animales en adopción en FAADA	20
Figura 2.7 - Ejemplo ficha de adopción en FAADA	21
Figura 2.8 - Ejemplo venta animal en MilAnuncios	22
Figura 3.1.- Logo HTML	25
Figura 3.2 - Logo PHP	25
Figura 3.3 - Logo JavaScript	26
Figura 3.4 - Logo CSS	26
Figura 3.5 - Logo SQL	27
Figura 3.6 - Explicación modelo vista controlador	28
Figura 3.7 - Logo AJAX	28
Figura 3.8 - Logo MySQL	29
Figura 3.9 - Logo vscode	30
Figura 3.10 - Ejemplo código Visual Studio Code	30
Figura 3.11 - Logo XAMPP	31
Figura 3.12 - Logo phpMyAdmin	31
Figura 3.13 - Logo PHPMailer	32
Figura 4.1 - Ciclo de vida incremental	35
Figura 4.2 - Diagrama de Gantt	35
Figura 4.3 - Diagrama casos de uso	39
Figura 4.4 - Diagrama E-R	41
Figura 4.5 - Diagrama Relacional	41
Figura 4.6 - Mockup página principal	43
Figura 4.7 - Mockup página animales	43
Figura 4.8 - Mockup página formularios	43
Figura 4.9 - Mockup página datos usuario	44
Figura 4.10 - Mockup página datos animal	44
Figura 4.11 - Diseño final página principal	45
Figura 4.12 - Diseño final página visualización de animales	45
Figura 4.13 - Diseño final formularios	45
Figura 4.14 - Diseño final perfil usuario	46
Figura 4.15 - Diseño final datos del animal	46
Figura 4.16 - Código subida imágenes al servidor	47
Figura 4.17 - Código modelo subida imágenes al servidor	47



Capítulo 1

Introducción

1.1.- LA ADOPCIÓN DE ANIMALES Y MASCOTAS

Podríamos definir la adopción animal como el proceso de tomar la responsabilidad de un animal que un dueño previamente ha abandonado o dejado a cargo de un refugio de animales [1]. Para hablar de la adopción animal a día de hoy en nuestro país podemos destacar los siguientes puntos:

1.1.1.- Comercialización de animales

La comercialización de animales comprende todas aquellas actividades tanto legales como ilegales que se basan en el comercio o tráfico con animales de compañía o silvestres. La comercialización de animales se trata de un problema a nivel global, ya que en muchos casos estos animales no tienen un fin como animal de compañía sino que son asesinados para su uso en la industria de la moda o en la medicina o para el sector de entretenimiento en zoos o diversos espectáculos.

También en muchos casos estos animales son abandonados en hábitats que no les pertenecen, causando en muchas ocasiones un desequilibrio natural irreparable ya que la fauna y flora locales no están preparadas para estas nuevas especies invasoras que podrían introducir enfermedades desconocidas para las especies del hábitat o convertirse en depredadores sin ningún tipo de competencia.

También es importante hablar de las condiciones de los animales comercializados, ya que en muchas ocasiones se encuentran en situaciones de abandono y requieren atención veterinaria urgente, atentando esta práctica por lo tanto contra el propio bienestar del animal.

Por último debemos indicar que en muchas ocasiones esta práctica fomenta el maltrato animal al ver dichas mascotas como simples objetos comprados que carecen de sentimientos y que se encuentran al servicio de quienes los compran.

1.1.2.- Protectoras de animales

A día de hoy, la forma más segura de poder adoptar una mascota es a través de las distintas protectoras y refugios que operan en nuestro país. Estas protectoras ofrecen un hogar temporal a todos aquellos animales que recogen abandonados de la calle o que hayan sido dejados allí por sus antiguos dueños. Se trata del medio más seguro ya que no solo ofrecen un hogar a estos animales sino también la atención médica que necesiten el tiempo que se encuentren bajo su tutela.

Para poder adoptar un animal de estas protectoras, los requisitos indispensables que debe cumplir la persona que va a adoptar son: ser mayor de edad, tener un domicilio en el que el animal vaya a vivir y, finalmente firmar un contrato de adopción donde la persona se haga responsable del cuidado del animal.

En muchas ocasiones, antes de todo el proceso de adopción se rellena un formulario o se realiza una entrevista oral donde se preguntan aspectos como para qué deseas el animal o si dispones de los medios para el cuidado del mismo. Una vez relleno este formulario un trabajador de la protectora podría incluso visitar tu hogar para cerciorarse de que el animal dispondrá de todos los recursos necesarios y que tus condiciones de vida son compatibles con una mascota. En el caso de recibir el visto bueno, se procederá a firmar un contrato de adopción donde se te hace responsable del animal y se te compromete a su cuidado y bienestar [2].

1.1.3. Asociaciones de animales

También es muy habitual encontrar distintas asociaciones que velan por el bienestar animal y cuyo principal objetivo es denunciar el abuso, el maltrato y el tráfico ilegal de animales. Por lo general, se podrían definir como grupos de personas que se asocian con fines y objetivos comunes para garantizar el bienestar animal, sin fines de lucro, debidamente organizadas y legalmente constituidas [3].

Por lo general suelen trabajar codo con codo junto a los refugios y protectoras de animales y podemos resumir sus funciones en los siguientes puntos:

1. Promover en todas las instancias públicas y privadas la cultura y la protección, atención y buen trato de los animales.
2. Cuidar y velar por la observancia y aplicación de la Ley de Protección y Bienestar Animal y su Reglamento.
3. Informar y concientizar el adecuado trato de los animales.
4. Denunciar las irregularidades en la tenencia de animales.
5. Promover campañas antiabandono, campañas de adopción y campañas de esterilización o castración masiva humanitarias, especialmente en sectores de escasos recursos.

1.1.4. Gestión de animales a nivel local

Todas nuestras ciudades están llenas de animales que o bien han sido abandonados o han nacido en libertad y que acaban formando colonias donde conviven con otros animales de su misma especie.

Los casos más conocidos suelen ser las colonias felinas, donde distintas generaciones de gatos conviven juntos para garantizar su supervivencia. En muchos de estos casos, estas colonias acaban siendo gestionadas por distintas asociaciones que se encargan del cuidado de estos animales pero respetando y preservando su hábitat natural. Dos ejemplos podrían ser la colonia felina que se encuentra en el conocido como parque de tráfico en Elche o incluso la propia colonia felina que se encuentra en la Universidad Miguel Hernández.

Estas colonias también suelen ser gestionadas por los propios ayuntamientos de las localidades en las que se encuentren. En la ciudad de Elche, por ejemplo, podemos encontrar el Plan Municipal de Gestión de Colonias Urbanas Felinas [4], el cual tiene como principal objetivo el bienestar de dichos animales en la ciudad, facilitando la convivencia con los ciudadanos y la prevención de problemas de salud pública, basándose en el control de la población de gatos, sus puntos de alimentación, las condiciones sanitarias y las personas colaboradoras responsables de dichas colonias

Por otro lado, los perros callejeros suelen ser rescatados por protectoras con el objetivo de buscarles un hogar ya que podrían ser una amenaza para el resto de la población mediante la transmisión de enfermedades o algún accidente.

1.1.5. Legislación actual

Entre las medidas que existen para proteger a los animales actualmente, cabe destacar la nueva ley de Bienestar Animal aprobada por el congreso de los diputados el día 9 de febrero de 2023[5]. Esta ley recoge los siguientes puntos:

- Se ha creado un “listado positivo” de aquellos animales que sí serán considerados animales de compañía y se considerará ilegal tener un animal que no pertenezca a esta lista. Sin embargo, al no tener la ley un carácter retroactivo, a todas aquellas personas que ya tuvieran un animal de este tipo antes de la entrada en vigor de la ley se les reconocerá como animal de compañía.
- Se prohíbe que los animales domésticos se encuentren de forma permanente a la intemperie, ya sea en patios o en balcones y terrazas y que queden expuestos a la lluvia y la nieve, o en los cuales no tengan espacios donde resguardarse, para evitar golpes de calor en verano e hipotermia en invierno.
- Se establecen periodos máximos en los que un animal puede estar sin supervisión, siendo el caso de los gatos 3 días y el de los perros 24 horas.
- Quienes deseen tener un perro deberán realizar un cursillo que acredite su capacidad para tener este tipo de animales. Lo que se sabe hasta ahora es que será obligatorio, se realizará por Internet y será gratuito.
- En cuanto a la compra de animales, las tiendas de animales solo podrán vender peces, roedores y pájaros y estos deberán venir de criadores registrados.
- En cuanto a los criadores, estos deberán tener una licencia profesional para ejercer su actividad, mientras que la cría por parte de particulares no registrados será considerada ilegal.
- En cuanto a aquellos animales que estén en estado silvestre o hayan sido abandonados, se prohíbe el sacrificio sin motivos justificados.
- Las colonias de gatos asilvestrados deberán ser esterilizados para impedir que crezcan en número, además, si están en suelo urbano será el propio ayuntamiento quien deberá ocuparse de garantizar su seguridad y salud.

1.2.- JUSTIFICACIÓN DEL PROYECTO

Si a través de Internet intentamos adoptar un animal, podremos observar rápidamente que son pocas las páginas que tienen esta finalidad y aún más son pocas aquellas que ofrecen un entorno seguro para los animales.

Este proyecto nace con el objetivo de intentar solucionar este problema al crear un lugar seguro para la adopción animal a través de la red. Permitiendo así a aquellas personas que no puedan hacerse cargo de sus mascotas encontrarles un hogar donde sepan que van a recibir los cuidados que ellos consideren necesarios e indicar a su nuevo cuidador cuales son los cuidados u otros cuidados especiales que el propio animal necesite.

También cabe destacar que en Internet existen bastantes páginas dedicadas a la compra y venta de animales, sin embargo en nuestra página el proceso de adopción se realizará de forma gratuita intentando así evitar fomentar dicha actividad y todos los problemas derivados a partir de esta.

Por lo tanto, este proyecto no tiene un fin lucrativo sino un fin de ayuda a los animales por lo que no habrá ninguna forma de pago u obtener dinero dentro de la misma.

Este proyecto es más que un trabajo de final de carrera, ya que si la página cubre los requisitos necesarios me gustaría poder publicarla en la red para poder ayudar a todos aquellos animales que lo necesiten.

1.3.- OBJETIVOS

El objetivo principal del proyecto será crear la aplicación web anteriormente mencionada. Dicha aplicación web tendrá que tener los siguientes puntos:

- Página de registro, dividida según se trate de un usuario particular o de un refugio de animales.
- Página de visualización de animales, dividiendo esta página según el tipo de animal.
- Perfil de usuario donde se podrán visualizar los datos del usuario registrado.
- Página de visualización de los datos del animal.

- Chat interno para que los usuarios puedan ponerse en contacto y mantener conversaciones acerca del proceso de adopción.
- Incluir aspectos legales para que el proceso de adopción sea seguro tanto para el nuevo propietario como para el animal.

También se tendrán en cuenta otros aspectos como la usabilidad de la página, el diseño de la misma y su buen funcionamiento.

Desde un punto de vista personal, también se busca aprender el proceso de desarrollo de una aplicación web desde todos los puntos (base de datos, diseño, HTML, back-end, front-end, etc) pudiendo así reforzar todos los aspectos estudiados en la carrera y aprender técnicas de desarrollo web ya existentes como AJAX.

Como objetivo final se busca poder publicar esta página en la web para poder ayudar a todos aquellos animales que lo necesiten y poder darles un nuevo hogar en el que estén seguros y reciban los cuidados que necesiten.



Capítulo 2

Antecedentes y estado de la cuestión

2.1.- ADOPCIÓN DE ANIMALES ACTUALMENTE

Actualmente, la forma más segura de adoptar un animal consiste en ir directamente a una protectora o refugio y tramitar un proceso de adopción directamente con ellos. A pesar de que existen otros métodos, como contactar directamente con un particular, estos muchas veces están sujetos a especulación y al tráfico de animales, sobre todo si estamos hablando de especies exóticas o razas poco comunes. A pesar de que existen nuevas medidas (como la nueva ley de protección animal) que luchan por los derechos de los animales para evitar estas situaciones, es imposible a día de hoy no encontrar algún anuncio de venta de animales en portales tan conocidos como Milanuncios o incluso Wallapop.

El principal problema de esta compra y venta de animales es el hecho de que gran parte de estos animales acaban siendo abandonados, sin haber sido esterilizados previamente y en algunas ocasiones incluso con enfermedades infecciosas. Todo esto debido a que en muchas ocasiones estos animales acaban siendo vistos como juguetes los cuales una vez ya han cumplido su función de entretener acaban siendo tirados a la basura. Según un estudio de la fundación affinity, se calculó que un total de 285.554 animales fueron abandonados en nuestro país en el año 2021, una cifra que supone un 15% de todas las adopciones registradas ese mismo año, como se puede observar se trata de una cifra bastante alta [6].

Otro de los principales problemas de la compra y venta de animales es la existencia de las conocidas como “Puppy Mills”, granjas intensivas de bajo costo donde no se tienen en cuenta la seguridad ni el bienestar animal ya que buscan obtener crías de animales al menor precio posible para poder venderlos. Todos estos aspectos al final no acaban sólo en el abandono y todos los problemas que esto acarrea, sino que en muchas ocasiones viene acompañado de maltrato físico hacia estos animales ocasionándoles lesiones e incluso traumas que los acompañarán el resto de su vida.

Entre todas las soluciones que podemos encontrar, la que busca este proyecto es la de una adopción segura y regulada donde se mire por el bienestar del animal y se asegure que el animal va a recibir un trato y unos cuidados adecuados.

2.2.- CÓMO ADOPTAR UN ANIMAL ACTUALMENTE

A pesar de que no existen muchos portales dedicados a la adopción animal, algunos que podemos encontrar y que servirían de ejemplo para este proyecto sería los siguientes:

2.2.1.- Protectora Baix Vinalopó

Un primer ejemplo a tener en cuenta sería la página de la protectora del Baix Vinalopó [7] (<https://protectorabaix.org/>). Esta protectora opera principalmente por la zona de Alicante y parte de Murcia y tiene como principal objetivo acoger animales abandonados para posteriormente buscarles un hogar.



Figura 2.1 - Portal “Protectora Baix Vinalopó”

También busca personas interesadas en crear nuevos refugios o casas de acogida para distintos animales o gente que mediante sus donaciones pueda costear el costo del mantenimiento o los tratamientos veterinarios de los animales.

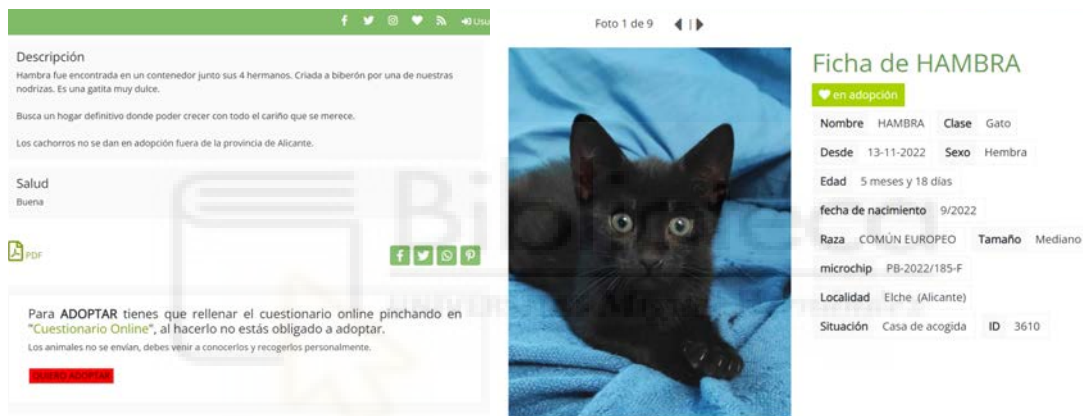


Figura 2.2 - Ejemplo de ficha de un animal en adopción

La figura 2.2 muestra a un animal en adopción, se puede observar que se trata de una ficha bastante completa donde se incluyen bastantes datos y que presenta aspectos importantes como la salud del animal. También podemos encontrar un botón de adopción para poder rellenar un cuestionario con el que la protectora se pondrá en contacto con el interesado.

2.2.2.- Sociedad Protectora para Animales de Torre Vieja (SAT)

La sociedad Protectora para Animales de Torre Vieja [8], también conocida como “SAT”, se trata de una asociación voluntaria cuyo objetivo es rescatar y realojar perros sin hogar o abandonados en la zona de la Vega Baja (<https://satanimalrescue.com/es/>). Del mismo modo que el caso anterior, su principal fuente de ingresos son los donativos y la caridad de las personas, aunque también disponen de distintas tiendas donde parte de los beneficios son usados para el mantenimiento de los animales.



Figura 2.3 - Sociedad Protectora para Animales de Torrevieja

Como en el caso anterior, también podemos encontrar un modelo de ficha con los datos de los distintos animales que se encuentran en adopción (ver figura 2.4). A pesar de no tratarse de una ficha tan completa como en el caso anterior, también incluye lo que serían los aspectos más importantes del animal y distintas fotos del mismo.

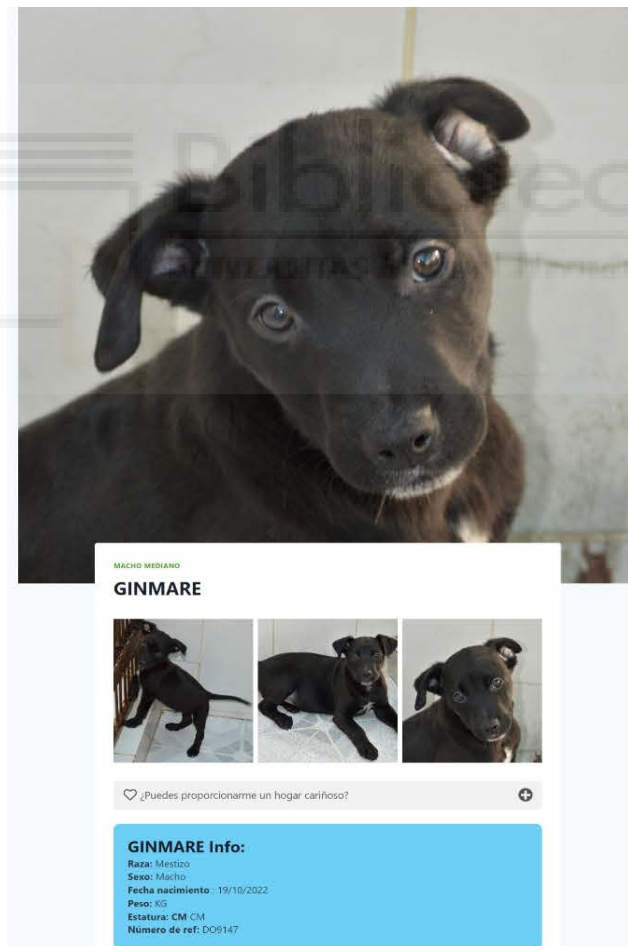


Figura 2.4 - Ejemplo de ficha de Animal en adopción en SAT

2.2.3.- FAADA

La Fundación para el Asesoramiento y Acción en Defensa de los Animales [9], también conocida como FAADA (<https://faada.org/>), es una fundación nacida en el año 2004 cuya principal actividad era la de apadrinar, adoptar y rescatar animales salvajes y de compañía. A partir de la experiencia que obtuvieron en este sector, en el año 2012 decidieron usar todo el conocimiento y recursos adquiridos durante esos 8 años para comenzar a trabajar de forma más estratégica en el ámbito legislativo, educativo y social. Siendo por lo tanto su principal objetivo actualmente promover el respeto animal en los tres ámbitos anteriormente mencionados.



Figura 2.5 - Página Principal de FAADA

Como podemos observar (figura 2.5), en su página disponen de diversos artículos sobre la actualidad del mundo animal así como distintos cursos para aprender cómo actuar en la protección y la defensa de los animales.

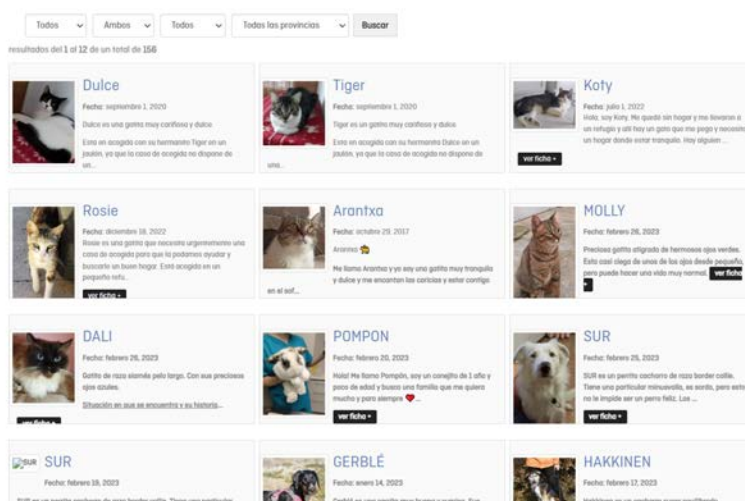


Figura 2.6 - Animales en adopción en FAADA

La web de FAADA también dispone de un apartado donde observar los distintos animales que se encuentran buscando un hogar (figura 2.6), así como fichas con la información individual de cada uno de dichos animales (figura 2.7).

Adopta a Tiger



Tiger es un gatito muy cariñoso y dulce.

Esta en acogida con su hermanita Dulce en jaulón, ya que la casa de acogida no dispone de una habitación para cada gatito. Tiger fue llevado a una perrera con sólo 2 meses, y allí no hubiera sobrevivido, por lo que corrimos para encontrar un lugar seguro para él y su hermanita.

Llevan ya 8 meses viviendo en jaulón y necesitan una familia para tener una vida más digna y en compañía de personas que los quieran. Nos gustaría que se adoptaran juntos por que están muy unidos, pero si no es posible, se darán en adopción por separado.

Importante proteger balcón/terraza, en caso de haberla.

Contacto: M. Carmen, telf. 654 25 69 06

Mari , telf. 650033770 (estrellamari@hotmail.com)

Fecha de alta: septiembre 1, 2020 Tamaño: Mediano Raza: Común europeo (atigrado y blanco) Provincia: Barcelona

Localidad: Barcelona Refugio / Protectora: El Amigo Fiel del Cuervo Núcleo zoológico: ES419030000324 Sexo: Macho

Edad: 3 años y 4 meses Nació: 11 / 2019 Estado del animal: Pendiente de Adoptar web: Faada

FAADA

Fundación FAADA
Rambla Prim, 105-157 Entresuelo 2º
08020 Barcelona

93 624 55 38
Contacto

[Suscríbete al Newsletter](#)

CONOCENOS

- Quiénes somos
- Qué hacemos
- Cómo nos financiamos
- Equipo
- Contacto

INFORMATE

- Comercio
- Entretención
- Convivencia
- Educación

ACTUALIDAD

- Noticias
- Peticiones
- Prensa
- Agenda
- Multimedia

COLABORA

- Actúa
- Hazte socio/a
- Haz un donativo
- Empresas comprometidas
- Iniciativas solidarias
- Prácticas
- Legados
- Tienda

[Política de privacidad](#) [Política de cookies](#) [Aviso legal](#)

Figura 2.7 - Ejemplo ficha de adopción en FAADA

2.2.4.- Milanuncios

Por poner un mal ejemplo, véase el siguiente anuncio (figura 2.8) encontrado en la página de Milanuncios [10]. Como podemos observar, la camada de chihuahuas se encuentra en

venta e incluso son tratados como mercancía al indicar que pueden ser enviados a las Islas o a cualquier parte de España.



The image shows a screenshot of a MilAnuncios advertisement for Chihuahua puppies. The main image is a close-up of a brown and white Chihuahua puppy sitting on grass. Above the image, the text reads 'CAMADA DE CHIHUAHUAS ADORABLES//:' and '500 €'. To the right of the image are buttons for 'Mensaje' and 'Llamar'. Below the image, there is a 'Tienda' section for 'CENTRE CANI' with 'Mensaje' and 'Llamar' buttons. At the bottom, there is a 'Comparte este anuncio' section with social media icons. The description below the image reads: 'Adorables cachorritos chihuahua de pura raza disponibles. Criados en nuestro ambiente familiar, crecen inteligentes y sociables. Ven a visitarlos y compartir con ellos. Cada perrito se entrega con las vacunas correspondientes, microchip implantado, contrato de garantía genética - virica y propiamente desparasitado. Envíos en 24hrs a Baleares, Canarias y a toda la península. Precios desde 500euros: precios variables según características del cachorro. No incluye IVA Contacto - Whatsapp:'. There is also a 'VENDEDOR AFILIADO' badge in the bottom right corner of the image area.

Figura 2.8 - Ejemplo venta animal en MilAnuncios

2.2.5.- Resumen

Como podemos observar en los ejemplos anteriores, existen varias páginas en Internet de protectoras, refugios e incluso asociaciones que permiten la adopción de aquellos animales que se encuentren bajo su tutela. Sin embargo, en contraparte también podemos encontrar bastantes anuncios en portales web donde se permite la compra de animales fomentando todos los problemas que esta actividad acarrea.

También podemos observar que a pesar de que existen distintos portales para la adopción animal, ninguno de ellos está directamente creado para ayudar a particulares que no puedan hacerse cargo de sus mascotas, el cual principalmente es el propósito de este proyecto, ya que incluso en Milanuncios la gran mayoría de anuncios están publicados por personas que tienen granjas y se dedican a la venta animal o por tiendas de mascotas.

2.3.- VALORACIÓN

A pesar de que no existen muchos portales de adopción animal, podemos observar que los que existen actualmente están bastante comprometidos con el bienestar animal. A pesar de todo esto, estas se tratan de páginas que trabajan directamente con su propia protectora o con otras protectoras de la zona a la que pertenecen. No existe ninguna página conocida que se dedique expresamente a permitir a particulares que puedan encontrar un nuevo hogar para sus mascotas, y como hemos podido comprobar las que existen (como Milanuncios o Wallapop) están principalmente destinadas a la compra y venta de mascotas y no a lo que sería la adopción sin ánimo de lucro.

En este apartado es donde quiere entrar este proyecto, ya que no solo quiere ayudar a protectoras sino también a todos esos usuarios que no puedan hacerse cargo de sus mascotas y deseen buscarle un nuevo hogar.



Capítulo 3

Hipótesis de trabajo



3.1.- LENGUAJES DE PROGRAMACIÓN

A continuación, se enumeran todos los lenguajes de programación usados en este proyecto así como una descripción de los mismos y con qué finalidad han sido usados.

3.1.1.- HTML

HTML se trata de un lenguaje de marcado para la elaboración de páginas web. Permite definir una estructura básica y un código para la definición de contenido de una página web y es un estándar a cargo del World Wide Web Consortium (organización dedicada a la estandarización de casi todas las tecnologías ligadas a la web).

HTML es un lenguaje de marcado que permite indicar la estructura de un documento mediante etiquetas y, al tratarse de un estándar, busca ser un lenguaje que permite que cualquier página web escrita en una determinada versión pueda ser interpretada de la misma manera por cualquier navegador web actualizado [11].



Figura 3.1.- Logo HTML

El lenguaje HTML sería por lo tanto utilizado en este proyecto para definir la organización general de la página web y dotarla de una estructura y consistencia.

3.1.2.- PHP

PHP es un lenguaje de programación interpretado del lado del servidor y de uso general que se adapta especialmente al desarrollo web. Creado por el danés-canadiense Rasmus Lerdorf en 1994, el código PHP suele ser procesado en un servidor web por un intérprete PHP implementado como un módulo, un daemon o como un ejecutable de interfaz de entrada común. En un servidor web, el resultado del código PHP interpretado y ejecutado formaría la totalidad o parte de una respuesta PHP [12].



Figura 3.2 - Logo PHP

En este proyecto, el lenguaje PHP será principalmente utilizado para realizar consultas y peticiones a la base de datos donde tendremos registrada la información tanto de usuarios como de los animales.

3.1.3.- JavaScript

JavaScript es un lenguaje de programación interpretado, se define como orientado a objetos, basado en prototipos, imperativo, de tipado débil y dinámico. Su principal uso radica en el lado del cliente, implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuarios y páginas web dinámicas [13].



Figura 3.3 - Logo JavaScript

El lenguaje JavaScript se usa principalmente para dar dinamismo a la web, permitir la carga de elementos sin la necesidad de refrescar la página y hacerla más accesible y fácil de usar.

3.1.4.- CSS

CSS (siglas de Cascading Style Sheets) es un lenguaje de diseño gráfico para definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado. Se usa principalmente para establecer el diseño visual de un documento en la web o para dar un diseño a las interfaces de usuario escritas en lenguaje HTML o XHTML.



Figura 3.4 - Logo CSS

El lenguaje principalmente busca mejorar la accesibilidad de un documento o interfaz, ser más flexible y controlar la especificación de características presentacionales [14]. En este proyecto, se utilizará principalmente para definir el diseño de la página web para que resulte atractiva y accesible a los usuarios de la aplicación.

3.1.5.- SQL

SQL (siglas de Structured Query Language) es un lenguaje de dominio específico, diseñado para administrar y recuperar información de sistemas de gestión de bases de datos relaciones. Una de sus principales características es que permite el uso del álgebra y el cálculo relacional para efectuar consultas con el fin de recuperar información de bases de datos así como para realizar cambios en ellas [15].



Figura 3.5 - Logo SQL

El lenguaje SQL se utiliza en este proyecto para realizar las consultas con la base de datos de la página y para actualizar la misma. Las consultas SQL irán siempre dentro de código PHP ya que este último permite la inclusión de código SQL en algunas de sus librerías.

3.2.- METODOLOGÍAS DE DISEÑO E IMPLEMENTACIÓN

En el siguiente apartado se explican las arquitecturas, patrones y metodologías de diseño e implementación usadas en este proyecto así como también una explicación del uso que se les da en la página web implementada.

3.2.1.- Modelo Vista Controlador (MVC)

El modelo vista controlador consiste en un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario y la lógica de control en tres componentes distintos. A pesar de ser un modelo muy antiguo, ha demostrado su validez a lo largo de los años y se sigue usando en una gran cantidad de aplicaciones y procesos de desarrollo. Como se ha dicho, este estilo de arquitectura separa el código en tres capas distintas:

- Modelo: se encarga de acceder a la capa de almacenamiento de datos, define la funcionalidad del sistema y sus mecanismos de persistencia.

- Vista: conforma la interfaz de usuario y se compone de la información que se envía al cliente y los mecanismos de interacción con este.
- El Controlador: actúa como intermediario entre el Modelo y la Vista, gestionando el flujo de información entre ellos y las transformaciones para adaptar los datos a las necesidades de cada uno.

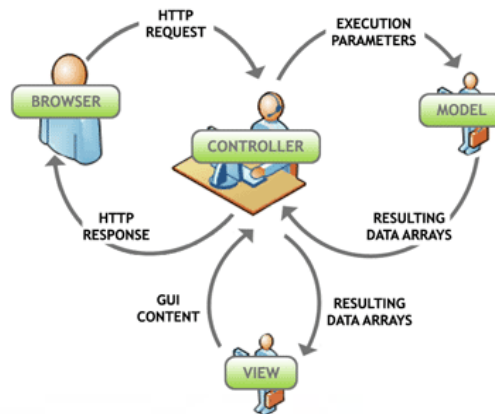


Figura 3.6 - Explicación modelo vista controlador (fuente: <https://www.troopsf.com/mvc/>)

El modelo vista controlador se utiliza en este proyecto para asegurar un correcto flujo de datos entre los distintos elementos de la aplicación web y asegurar así la separación de la lógica de datos de la interacción del usuario [16].

3.2.2.- AJAX

AJAX (acrónimo de Asynchronous JavaScript and XML) se trata de una tecnología JavaScript que permite hacer consultas por HTTP de manera asíncrona contra el servidor, es decir, permite acceder a datos existentes en el servidor sin necesidad de recargar la página completamente. Estas consultas contra el servidor se realizan por medio de JavaScript y los datos se procesan mediante este mismo lenguaje, permitiendo actualizar el contenido de la página justamente donde sea preciso. [17]



Figura 3.7 - Logo AJAX

Todo esto hace que se mejore la experiencia de usuario ya que permite la carga y descarga dinámica de contenido de una web sin la necesidad de refrescar la misma. AJAX es utilizado en este proyecto con ese fin, crear una página dinámica que cargue su contenido de forma transparente para el usuario mejorando así su experiencia con la misma.

3.3.- BASES DE DATOS MySQL

MySQL es un sistema de gestión de bases de datos relacionales y está considerado como la base de datos de código abierto más popular del mundo. Utiliza un modelo cliente-servidor donde el cliente realiza distintas solicitudes escribiendo instrucciones SQL específicas en MySQL y la aplicación del servidor responderá con la información solicitada y esta aparecerá frente a los clientes. [18]

Se caracteriza por ser de código abierto, lo cual significa que el usuario es libre de usarlo y modificarlo. Haciendo por lo tanto que el código fuente se adapte mejor a las necesidades de una aplicación concreta. Su alto rendimiento y flexibilidad han convertido a MySQL en un estándar que la industria ha estado usando durante muchos años. En este proyecto se usa la base de datos MySQL para almacenar todos los datos de la aplicación desarrollada. [19]



Figura 3.8 - Logo MySQL

3.4.- HERRAMIENTAS DE DESARROLLO

A continuación se indicarán todas las herramientas usadas para el desarrollo de esta aplicación web.

3.4.1.- Visual Studio Code

Visual Studio Code es un editor de código fuente desarrollado por Microsoft que incluye soporte para la depuración, control integrado de Git, resaltado de sintaxis, finalización inteligente de código y refactorización de código.

- M: La letra M hace referencia a la incorporación del sistema de gestión de bases de datos conocido como MySQL o MariaDB.
- P: XAMPP utiliza el lenguaje de programación PHP.
- P: Perl es otro lenguaje de programación que proporciona XAMPP, enfocado en la administración del sistema y programación de red.

XAMPP se utiliza, por lo tanto, para poder gestionar la base de datos de la aplicación web desarrollada y como servidor web local. [21]



Figura 3.11 - Logo XAMPP

3.4.3.- phpMyAdmin

phpMyAdmin es una herramienta de software libre desarrollada en PHP, destinada a manejar la administración de MySQL en la Web. phpMyAdmin es compatible y permite una gran cantidad de operaciones tanto en MySQL como MariaDB.

Dispone de una interfaz de usuario que permite realizar las operaciones más frecuentes cómo crear tablas, relaciones, permisos, etc. También permite ejecutar directamente cualquier instrucción SQL a través de su interfaz gráfica.

phpMyAdmin dispone de una gran cantidad de documentación, que abarca desde páginas web hasta libros escritos por sus propios desarrolladores. phpMyAdmin se utiliza por lo tanto en este proyecto para poder gestionar la base de datos de la aplicación web. [22]



Figura 3.12 - Logo phpMyAdmin

3.5.- LIBRERÍAS UTILIZADAS

A continuación se indicarán las librerías utilizadas para el desarrollo de esta aplicación web.

3.5.1.- PHPMailer

PHPMailer es una librería de código abierto escrita en PHP que permite el envío de correos electrónicos de forma segura. Fue desarrollada por Brent R. Matzelle, Andy Prevost, Jim Jagielski y Marcus Bointon, originalmente lanzada en el año 2001, siendo a día de hoy una de las librerías más utilizadas para el envío de correos electrónicos en aplicaciones web basadas en PHP. [23]

En este proyecto la librería ha sido utilizada para el envío de correos electrónicos de confirmación a la hora del registro (para comprobar que el correo existe realmente) y la eliminación de la cuenta.

Se puede encontrar tanto el enlace de descarga como un pequeño tutorial de la propia librería en el siguiente enlace: <https://github.com/PHPMailer/PHPMailer> [24]



Figura 3.13 - Logo PHPMailer

3.5.2. - Librerías "utilidades.js" y "utilidades.php"

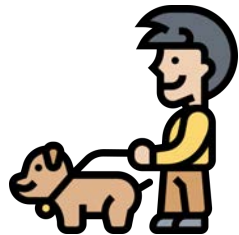


Las librerías "utilidades.js" y "utilidades.php" son dos librerías ligeras que trabajan conjuntamente para la comunicación cliente/servidor en aplicaciones web. Dichas librerías contienen funciones para crear aplicaciones siguiendo el patrón MVC, acelerando el desarrollo de aplicaciones en JavaScript y PHP.

Además, cabe destacar que dichas librerías también contienen todas las funciones necesarias para que el motor AJAX de la página web funcione correctamente. [25]

3.6.- REPOSITORIOS DE IMÁGENES GRATUITAS

A lo largo de la aplicación web se han usado distintas imágenes obtenidas de repositorios gratuitos y de libre uso. En la tabla 2.1 se muestra el origen de todas las fuentes visuales utilizadas dentro de la aplicación web.

Tabla 3.1 - Repositorios de imágenes

Repositorio	Descripción	Ejemplo
Flaticon [26]	De esta página se han obtenido tanto el icono de enviar mensaje, como los iconos de selección de registro y el icono del favicon.	
Pixabay [27]	De esta página se ha obtenido la imagen de la sección “Adopta” que se encuentra en la página principal.	
Emojipedia [28]	De esta página se han obtenido los emojis usados en la página de visualización de animales.	

Capítulo 4

Metodología y resultados

4.1.- PLANIFICACIÓN DEL PROYECTO

La división de un proyecto en fases permite disminuir la incertidumbre y dividir el trabajo entre los encargados del desarrollo del proyecto. Para conseguir esta división del proyecto se ha aplicado un ciclo de vida iterativo, más concretamente, el ciclo de vida incremental. En primer lugar, un ciclo de vida iterativo busca dividir el cronograma del proyecto en pequeñas fases relativamente independientes de la anterior. Estas fases reciben el nombre de iteraciones o ciclos del proyecto, y a pesar de que cada iteración es una parte de un todo, tienen su propio objetivo particular.

Es decir, cada fase del ciclo iterativo tiene un objetivo acotado y busca obtener parte del producto final en pequeños bloques. [29]

Se opta por los ciclos de vida iterativos cuando es necesario gestionar objetivos poco definidos o de una alta complejidad o cuando la entrega parcial del producto es clave para el éxito. Este tipo de ciclo de vida permite al equipo del proyecto incorporar la retroalimentación e ir incrementando la experiencia del equipo durante el proyecto[30].

El ciclo de vida incremental es una particularización del anterior, mediante el cual cada ciclo que se realiza va obteniendo una porción del proyecto, servicio o resultado completo. A cada porción generada en una iteración se le denomina incremento. Es decir, vamos produciendo porciones del resultado del proyecto que están acabadas al 100% e iteramos hasta tener todas las porciones [31].

Son cuatro las fases que componen el modelo incremental: análisis, diseño código y pruebas. Se pueden observar en la siguiente imagen:

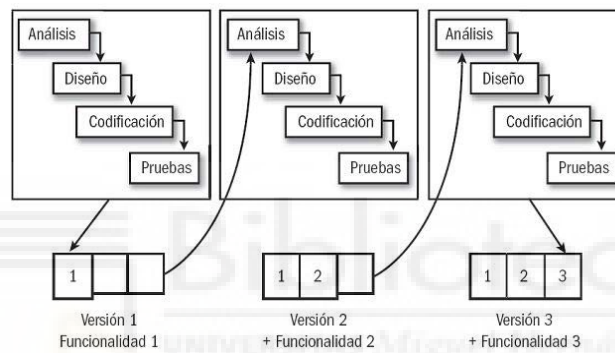


Figura 4.1 - Ciclo de vida incremental (fuente:

<https://abzlocal.mx/top-33-imagen-modelo-de-ciclo-de-vida-incremental/>)

4.1.1.- Diagrama de Gantt

A continuación, se incluirá el diagrama de Gantt que resume el desarrollo del proyecto, así como sus etapas y el tiempo de realización de cada una de ellas.

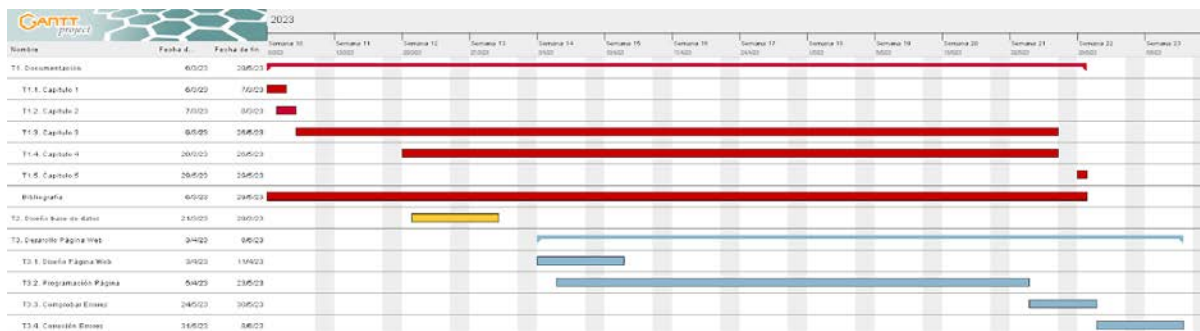


Figura 4.2 - Diagrama de Gantt

4.2.- CAPTURA DE REQUISITOS

Mediante las siguientes tablas se explicarán todos los requisitos funcionales que deberá tener la aplicación web desarrollada.

Tabla 4.1 - Requisitos Página Principal

RF-1	Página Principal
Descripción	La aplicación consta de una página web desarrollada con las tecnologías HTML5, JavaScript, PHP y basada en un motor AJAX.

Tabla 4.2 - Requisitos Registro de Usuarios

RF-2	Registro de Usuarios
Descripción	La aplicación permitirá el registro tanto de usuarios particulares como de protectoras y refugios de animales.

Tabla 4.3 - Requisitos del Registro de Animales

RF-3	Registro de Animales
Descripción	Una vez que el usuario se haya registrado, este podrá registrar los distintos animales de los que no pueda hacerse cargo. De esta forma, se publicará un anuncio con los datos del animal mediante el cual los distintos usuarios podrán ponerse en contacto con su actual dueño.

Tabla 4.4 - Requisitos de la Visualización de Animales

RF-4	Visualización de animales
Descripción	En la página podrán verse las características de cada uno de los animales que hayan sido registrados en la plataforma.

Tabla 4.5 - Requisitos del Inicio de sesión

RF-5	Inicio de sesión
Descripción	Los usuarios podrán iniciar sesión dentro de la página web una vez se hayan registrado y así podrán acceder a distintas acciones que si no estuvieran registrados no podrían.

Tabla 4.6 - Requisitos de la visualización del perfil de usuario

RF-6	Visualización perfil de usuario
Descripción	El usuario podrá ver toda la información asociada a su perfil. Dicha información será la que se ha indicado a la hora del registro. También podrán gestionar todos los animales que tengan registrados en la aplicación.

Tabla 4.7 - Requisitos de la comunicación entre usuarios

RF-7	Comunicación entre usuarios
Descripción	Los usuarios podrán comunicarse entre ellos para poder formalizar el proceso de adopción de la mascota.

Tabla 4.8 - Requisitos de la detección de errores

RF-8	Detección de errores
Descripción	La página dispondrá de mecanismos para la detección de errores durante el proceso de registro y para comprobar que se indican los datos de forma correcta.

Tabla 4.9 - Requisitos de la base de datos

RF-9	Mantenimiento de la base de datos
Descripción	La base de datos que utiliza esta aplicación está basada en MySQL. Se deberá mantener y actualizar con frecuencia la base de datos para asegurar que no ocurra ningún tipo de error.

La aplicación va a contar con tres roles o tipos de usuario:

- Administrador
- Usuario registrado
- Usuario no registrado

A continuación, en las tablas 4.10, 4.11 y 4.12, se expondrá el nombre de cada actor (tipo de usuario), junto a una descripción del mismo y los casos de uso con los que está relacionado.

Tabla 4.10 - Casos de uso del actor Administrador

Actor	Administrador
Descripción	El administrador es aquella persona que se encarga del mantenimiento de la página, de asegurar el correcto funcionamiento de esta y de actualizar y mejorar la misma.
Caso de uso relacionados	<p>Cu 1. Acceso a la página web</p> <p>Cu 2. Registro particular</p> <p>Cu 3. Registro refugio</p> <p>Cu 4. Iniciar Sesión</p> <p>Cu 5. Recuperar contraseña</p> <p>Cu 6. Registro de animales</p> <p>Cu 7. Ver Perfil</p> <p>Cu 8. Eliminar Animales</p> <p>Cu 9. Registrar Adopción</p> <p>Cu 10. Visualización de animales</p> <p>Cu 11. Eliminar Cuenta</p> <p>Cu 12. Crear Chat</p> <p>Cu 13. Visualizar Chats</p> <p>Cu 14. Cerrar Sesión</p>

Tabla 4.11 - Casos de uso del actor Usuario no Registrado

Actor	Usuario no registrado
Descripción	El usuario no registrado es todo aquel usuario que acceda a la página web pero que no haya formalizado su proceso de registrado. Por lo tanto, tendrá funciones limitadas dentro de la página.
Caso de uso relacionados	<p>Cu 1. Acceso a la página web</p> <p>Cu 2. Registro particular</p> <p>Cu 3. Registro refugio</p> <p>Cu 10. Visualización de animales</p> <p>Cu 15. Contactar con el administrador</p>

Tabla 4.12 - Casos de uso del actor Usuario Registrado

Actor	Usuario registrado
Descripción	Usuario que se ha dado de alta dentro de la página y disfruta de todas las funciones de dentro de la página.
Caso de uso relacionados	<p>Cu 1. Acceso a la página web</p> <p>Cu 4. Iniciar sesión</p> <p>Cu 5. Recuperar contraseña</p> <p>Cu 6. Registro de animales</p> <p>Cu 7. Ver perfil</p> <p>Cu 8. Eliminar animales</p> <p>Cu 9. Registrar adopción</p> <p>Cu 10. Visualización de animales</p> <p>Cu 11. Eliminar cuenta</p> <p>Cu 12. Crear chat</p> <p>Cu 13. Visualizar chats</p> <p>Cu 14. Cerrar sesión</p> <p>Cu 15. Contactar con el administrador</p>

El diagrama de los casos de uso para los usuarios de la aplicación quedaría como se indica en la figura 4.3.

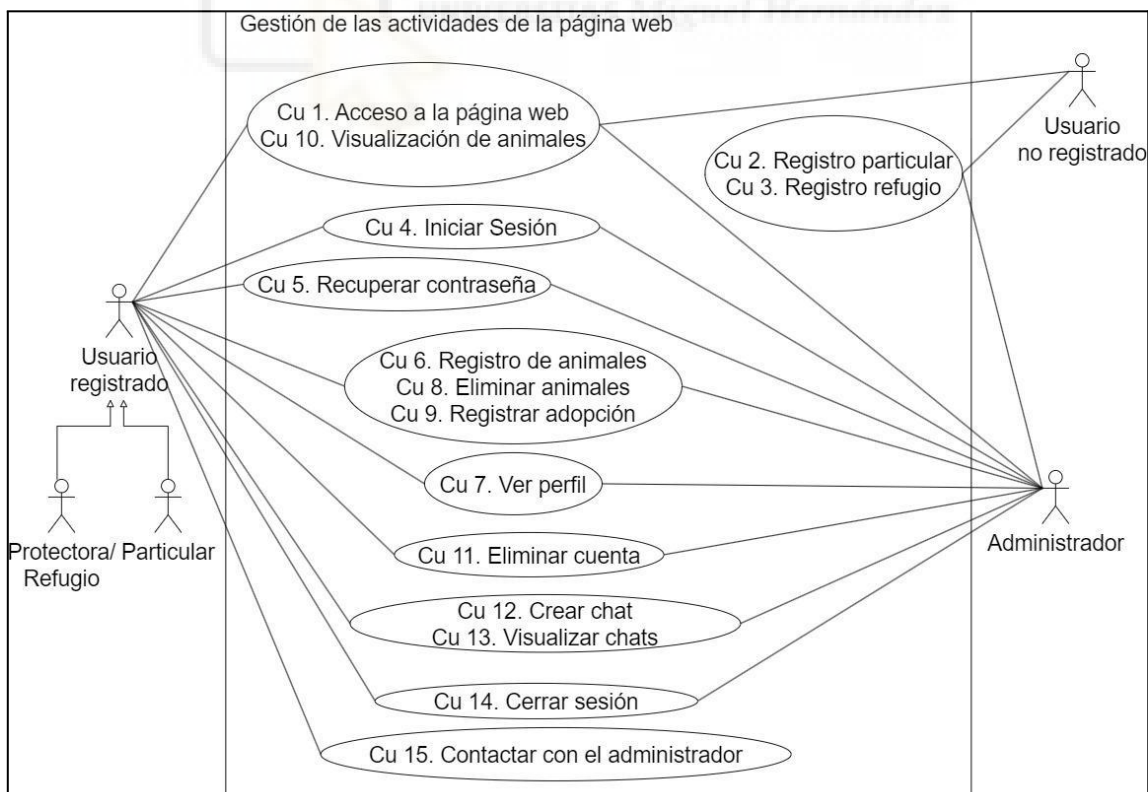


Figura 4.3 - Diagrama casos de uso

La tabla 4.13, muestra una descripción detallada del funcionamiento del primero de los casos de uso. Los casos de uso restantes se pueden encontrar en el anexo número 1 de esta documentación.

Tabla 4.13 - Caso de uso N°1

Cu 1. Acceso a la página web	
Actores que intervienen	Usuarios registrados, no registrados y administrador.
Descripción	Una persona desea acceder a la aplicación web para poder visualizar todos los contenidos que esta ofrezca.
Dependencias	No
Precondición	Disponer de un dispositivo inteligente y una conexión a internet para poder acceder a la página.
Secuencia normal	1. Acceder a la aplicación. 2. Visualizar el contenido.
Post Condición	
Excepciones	No
Rendimiento	
Frecuencia	Alta
Importancia	Alta
Urgencia	Alta
Estabilidad	
Comentarios	

4.3.- DISEÑO

4.3.1 Diseño de la base de datos

La base de datos de la plataforma ha sido implementada mediante el sistema de gestión de base de datos MySQL y con el apoyo de la herramienta phpMyAdmin (herramienta escrita en PHP que permite el manejo de MySQL a través de un navegador web).

El diagrama E-R de la base de datos de la aplicación es el que se muestra en la figura 4.4, de igual modo, en la figura 4.5 se presenta el diseño relacional de la misma base de datos.

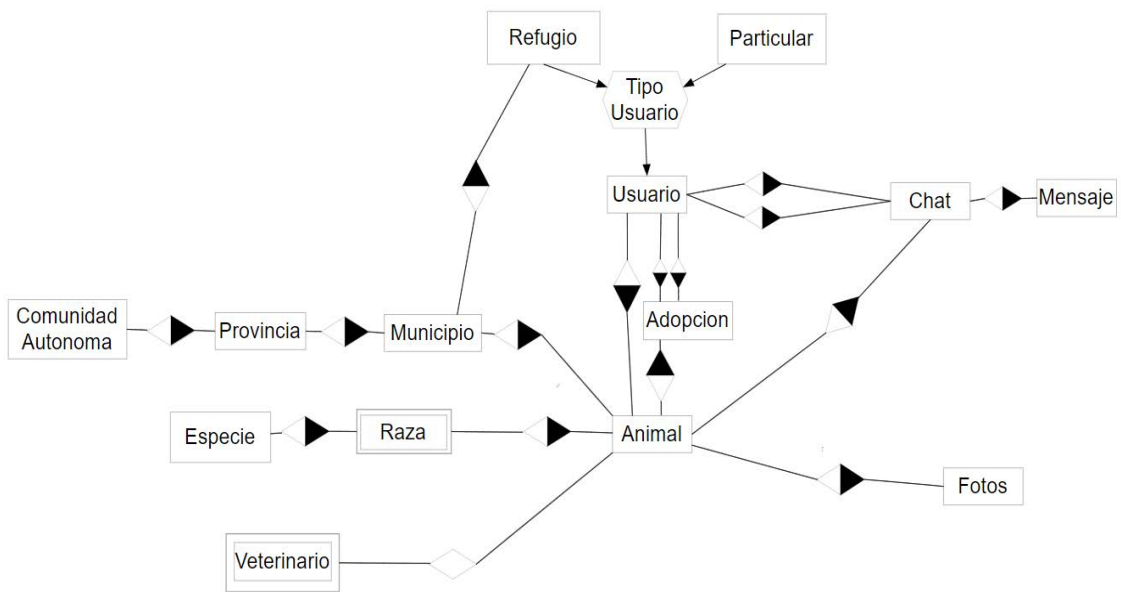


Figura 4.4 - Diagrama E-R

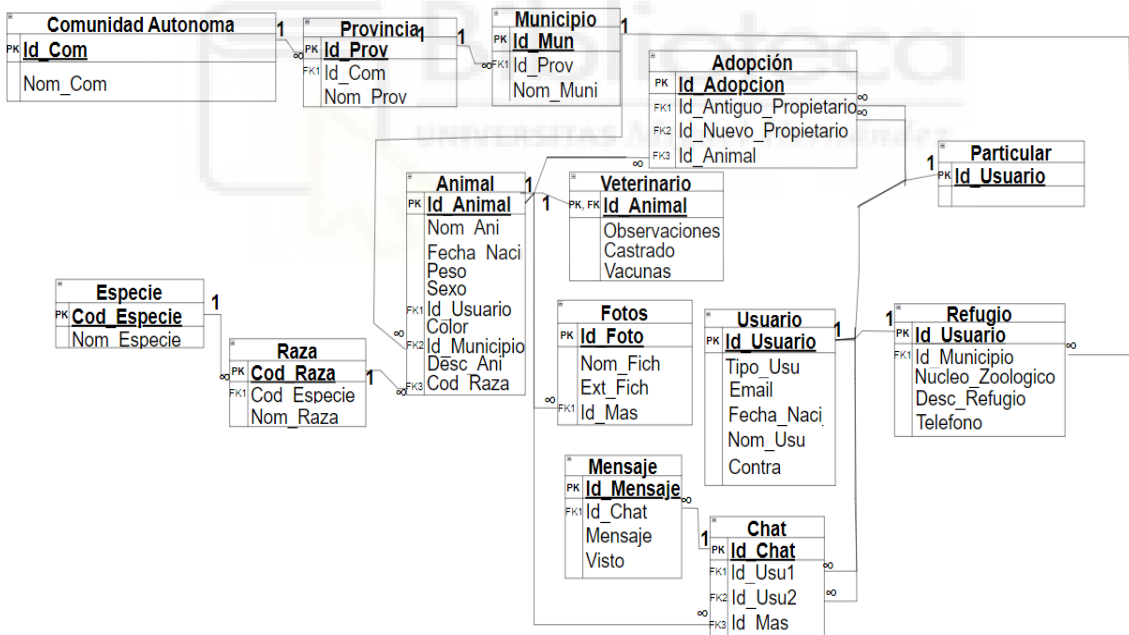


Figura 4.5 - Diagrama Relacional

A continuación se muestra una descripción y el diseño detallado de la tabla “Animal” de la base de datos (ver tabla 4.14). En el anexo 2 se encuentra la descripción y diseño completo de todas las tablas de la aplicación.

Animal: Tabla donde almacenaremos la información de los animales que hayan sido registrados en nuestro servidor. Compuesta por los campos:

Tabla 4.14 - Diseño tabla Animal

Tabla Animal	
Id_Animal	Clave Principal, de tipo entero
Nom_Ani	Nombre del animal, varchar de 15 de longitud.
Fecha_Nacimiento	Fecha de nacimiento de la mascota, tipo fecha.
Sexo	Sexo de la mascota, varchar de longitud 1 con dos posibles valores (H = Hembra, M= Macho).
Color	Color de la mascota, varchar de longitud 30.
Peso	Peso de la mascota, de tipo entero.
Id_Usuario	Identificador del usuario que ha registrado dicha mascota, clave ajena procedente de la tabla Usuario.
Id_Municipio	Identificador del municipio de donde procede dicha mascota, clave ajena procedente de la tabla Municipio.
Cod_Raza	Identificador de la raza del animal, se trata de una clave ajena procedente de la tabla Raza.
Desc_Ani	Descripción del animal aportada por el usuario que lo registra, se trata de un campo de tipo text.
Animal (Id_Animal, Nom_Ani, Fecha_Nacimiento, Sexo, Color, Peso, Id_Municipio, Cod_Raza, Desc_Ani) C.P: Id_Animal C.Ajena: Id_Municipio -> Municipio C.Ajena: Id_Usuario -> Usuario C.Ajena: Cod_Raza -> Raza	

4.3.2 Diseño de la interfaz gráfica

A continuación, se expondrán los distintos croquis y mockups que se han tenido en cuenta para el diseño de la interfaz gráfica de la web.

Siendo el mockup de la página principal el siguiente:

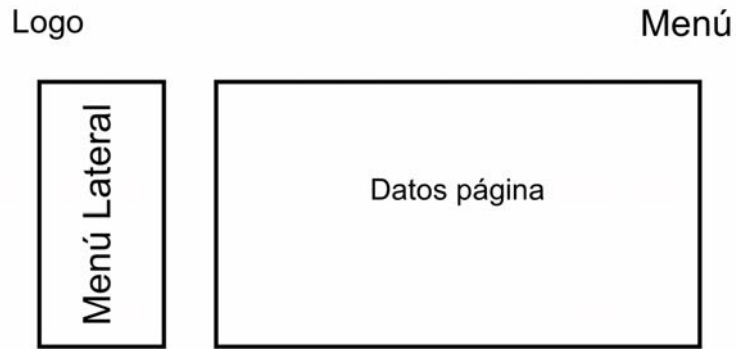


Figura 4.6 - Mockup página principal

El mockup de la página donde se mostrarán los animales el siguiente:

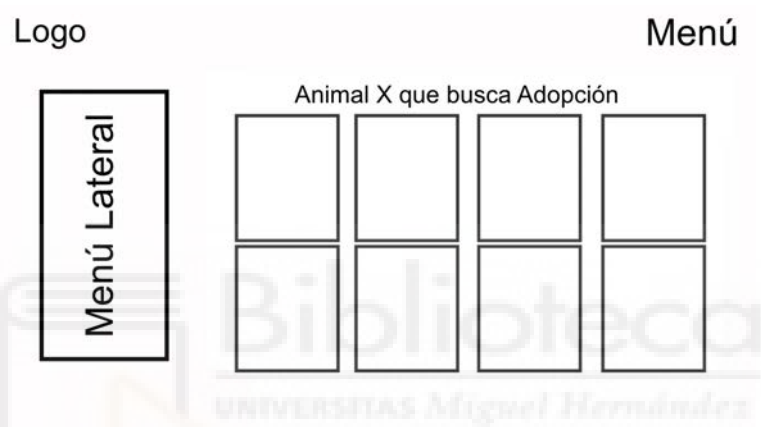


Figura 4.7 - Mockup página animales

El mockup de las páginas donde hayan formularios sería el siguiente:

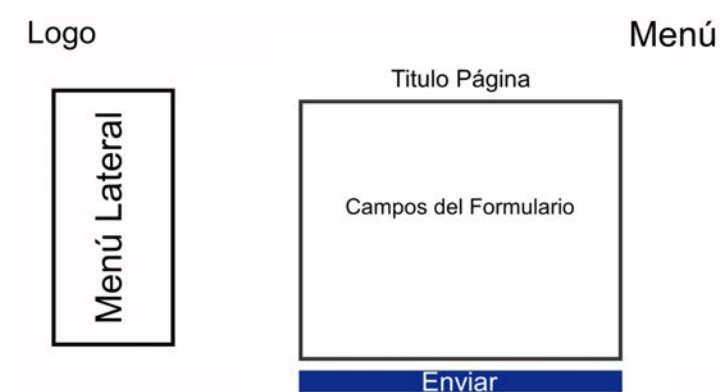


Figura 4.8 - Mockup página formularios

La página donde se mostrarán los datos de los usuarios tendrá el siguiente diseño:

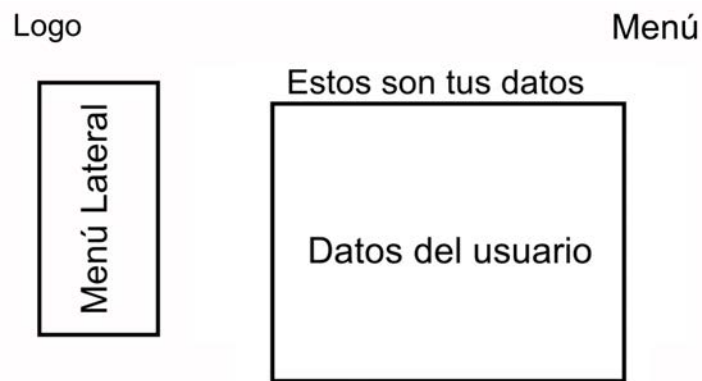


Figura 4.9 - Mockup página datos usuario

Finalmente, las pantallas donde se mostrarán los datos de los animales presentarán la siguiente estructura:



Figura 4.10 - Mockup página datos animal

4.4.- IMPLEMENTACIÓN

4.4.1. Resultado final de la aplicación

A continuación, se expondrán capturas de cómo ha sido el resultado final del diseño de la aplicación a partir de los distintos mockups mostrados en el apartado anterior.

En primer lugar, mostraremos el resultado final de la página principal de la aplicación:

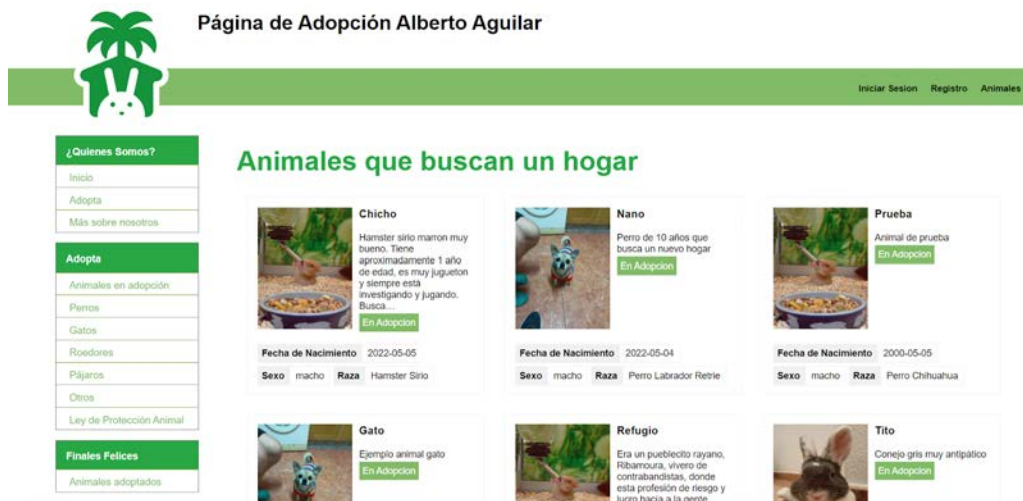


Figura 4.11 - Diseño final página principal

El diseño final de la página de visualización de animales sería el siguiente:



Figura 4.12 - Diseño final página visualización de animales

El diseño final de los distintos formularios que encontramos en la página sería el siguiente:



Figura 4.13 - Diseño final formularios

Los datos de los usuarios serán mostrados de la siguiente forma en la aplicación final:



Figura 4.14 - Diseño final perfil usuario

Y por último, el diseño final de la página donde se muestran los datos del animal sería la siguiente:



Figura 4.15 - Diseño final datos del animal

4.4.2. Peculiaridades de la aplicación

En este subapartado se van a describir ciertas particularidades que se han tenido que resolver para algunas de las funcionalidades de la aplicación.

Al tratarse de una página basada en la metodología AJAX, ha sido necesario aplicar el siguiente código en la parte del controlador de la página del registro del animal para poder subir varias imágenes al servidor.

```

if(ComprobarCampos()==1)
{
    //Codigo necesario para subir fotos a la carpeta del servidor
    var form = document.forms.namedItem("registro_animal");

    var oOutput = document.getElementById("login_msg");
    var oData = new FormData(form);

    oData.append("CustomField", "This is some extra data");

    var oReq = new XMLHttpRequest();

    oReq.open("POST", "08_RegistrarAnimal_m.php", true);

    oReq.onload = function(oEvent)
    {
        if (oReq.status == 200)
        {
            oOutput.innerHTML = oReq.responseText; //El responseText es el "echo" que devuelve el servicio
            //console.log("Respuesta servidor:"+oReq.responseText);
        }
        else
        {
            oOutput.innerHTML = "Error <br/>" + oReq.responseText;
        }
    };
    oReq.send(oData);
}

```

Figura 4.16 - Código subida imágenes al servidor

Este código se encarga de asignar (una vez clicado el botón de “Registrar al Animal”), conjuntos de pares clave-valor de los datos rellenos en el formulario para poder enviar dichos conjuntos mediante XMLHttpRequest al modelo, en el cual hay que implementar el siguiente código:

```

for($i=0 ; $i<$n ; $i++)
{
    $nombre=basename($_FILES['fichero'][$i]['name']);

    //Primero debemos coger la extensión del archivo para poder almacenarlo
    //Se divide en espacios, tabulaciones y cambios de línea
    $token = strtok($nombre, "."); // Primer token
    while($token != false)
    {
        $ext_fich=$token; //Variable que contendra la extensión del archivo
        // En los tokens subsecuentes no se incluye el string $cadena
        $token = strtok(".");
    }
    $ext_fich="." . $ext_fich;//Le añadimos el punto a la extensión del fichero

    //Insertamos la imagen en la tabla fotos
    $SQL="INSERT INTO fotos (Ext_Fich,Id_Mas) values ('$ext_fich','$id_animal_registrado)";
    $res = $mysqli->query($SQL); //Lanzamos la query del SQL y obtenemos una fila con todos los datos que devuelve la consulta
    $id_foto_subir=$mysqli->insert_id; //Guardamos el id de la foto que acabamos de registrar

    //Obtenemos el nombre final de la foto
    $nom_final_foto=$id_foto_subir.$ext_fich;

    //Guardamos el nombre en la base de datos
    $SQL="UPDATE fotos SET Nom_Fich='$nom_final_foto'WHERE Id_Foto='$id_foto_subir'";
    $res = $mysqli->query($SQL);

    $fichero_subido = $dir_subida . $nom_final_foto;
    if (move_uploaded_file($_FILES['fichero'][$i]['tmp_name'], $fichero_subido))
    {
    }
    else
    {
        echo "Fallo al subir la imagen: ".$i;
        break;
    }
}

```

Figura 4.17 - Código modelo subida imágenes al servidor

En dicho segmento de código se recorre mediante un bucle la variable `$_FILES` (que es la variable en la cual se almacenan todos los datos de los ficheros que han sido enviados en el formulario), obtendremos la extensión del fichero subido, calcularemos el nombre final de la foto (será la suma de su ID y su extensión) y a continuación se almacenará en una carpeta predefinida del servidor. Cabe destacar que previamente se han ido almacenando en la base de datos de MySQL todos los datos de la imagen.

Para poder enviar correos de confirmación al usuario o correos al administrador, ha sido necesario implementar la librería PHPMailer en nuestro servidor. Dicha librería nos permite enviar correos a distintos usuarios utilizando un dirección de correo electrónico predefinida. El código implementado en la parte del modelo sería el siguiente:

```
require 'PHPMailer/Exception.php';
require 'PHPMailer/PHPMailer.php';
require 'PHPMailer/SMTP.php';

//Create an instance; passing `true` enables exceptions
$mail = new PHPMailer(true);

//Creamos una variable de sesion con el código y el correo para poder comprobarlo mas tarde
$_SESSION["codigo_recu"]=$_POST["codigo"];
$_SESSION["correo_usu"]=$_POST["correo_usu"];

try {
    //Server settings
    $mail->SMTPDebug = 0; //Enable verbose debug output
    $mail->isSMTP(); //Send using SMTP
    $mail->Host = 'smtp.gmail.com'; //Set the SMTP server to send through //Configuramos
    $mail->SMTPAuth = true; //Enable SMTP authentication
    $mail->Username = 'albertoadopcion@gmail.com'; //SMTP username //Direccion de correo electronico que
    $mail->Password = 'cvzggstchzsqginc'; //SMTP password //Contraseña del correo anterior (gen
    $mail->SMTPSecure = 'tls'; //Enable implicit TLS encryption
    $mail->Port = 587; //TCP port to connect to; use 587 if you have set `SM

    //Recipients
    $mail->setFrom('albertoadopcion@gmail.com', 'Alberto'); //Indicamos el correo que envia y el nombre del usuar
    $mail->addAddress($_POST["correo_usu"]); //Indicamos el correo que recibe y su nombre (opciona

    //Content
    $mail->isHTML(true); //Set email format to HTML
    $mail->Subject = 'Completa tu registro';
    $mail->Body = 'El código a introducir para completar su registro es: '.$_POST["codigo"]."<br><br>Si no estas tr

    $mail->send();
    echo 'El mensaje se envio correctamente';
} catch (Exception $e) {
    echo "Hubo un error al enviar un mensaje. Error: {$mail->ErrorInfo}";
}
break;
```

Figura 4.18 - Código PHPMailer

La propia librería permite enviar mensajes de forma muy sencilla rellenando los distintos campos de propiedades que presenta. Incluso permite incrustar código HTML dentro del propio mensaje para poder darle forma y diseño. Sin embargo, como se tratan de simples mensajes con códigos o texto se ha optado por un mensaje sencillo simplemente con texto.

Capítulo 5

Conclusiones y trabajo futuro

5.1.- CONCLUSIONES

La web desarrollada cumple los principales objetivos que se tenían previstos y que han sido documentados en el capítulo 1 de esta memoria.

En general, estoy bastante contento con el trabajo realizado ya que la página ha sido dotada de todas aquellas funciones necesarias para que pueda ser funcional y usable, permitiendo así una buena experiencia de usuario a todas aquellas personas que en un futuro lleguen a usar este portal para poder adoptar una mascota.

Estoy bastante satisfecho de haber empleado la metodología AJAX ya que esto dota a la web de un gran dinamismo con muchas posibilidades tanto de desarrollo como de implementación.

En conclusión, el resultado final de la web tanto a nivel de diseño como de funcionalidad es satisfactorio y espero que el trabajo realizado pueda ayudar en un futuro tanto a mascotas como a sus dueños.

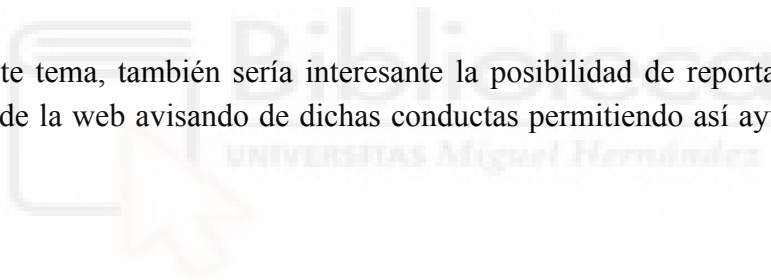
5.2.- POSIBLES DESARROLLOS FUTUROS

Para que la aplicación web cumpla su función sería necesario que en un futuro fuese publicada en Internet y que cualquier usuario pudiese utilizarla siempre que quisiese con el objetivo de ayudar a la adopción animal.

Para poder evitar el tráfico de animales, sería necesario dotar a la página web de medidas legales que permitieran evitar estas actividades, así como condenarlas o denunciarlas en el caso de que sea necesario.

Un filtro de palabras dentro del chat interno entre usuarios también sería buena idea para evitar la situación anterior, ya que al evitar que ciertos usuarios hablen sobre ciertos temas se ayudaría a evitar estas actividades.

En cuanto a este tema, también sería interesante la posibilidad de reportar usuarios a la administración de la web avisando de dichas conductas permitiendo así ayudar a combatir este problema.



Bibliografía

- [1] Adopción en animales - Wikipedia
https://es.wikipedia.org/wiki/Adopci%C3%B3n_en_animales
Fecha de consulta: 06/03/2023
- [2] Cómo adoptar cachorros de una protectora de animales
<https://www.zooplus.es/magazine/perros/adoptar-un-perro/como-adoptar-cachorros-de-una-protectora-de-animales>
Fecha de consulta: 06/03/2023
- [3] ¿Asociaciones Protectoras de Animales?
<https://www.maga.gob.gt/sitios/uba/asociaciones-protectoras-de-animales/#:~:text=Las%20Asociaciones%20Protectoras%20de%20Animales,debidamente%20organizadas%20y%20legalmente%20constituidas.>
Fecha de consulta: 06/03/2023
- [4] Plan Municipal de Gestión de Colonias Urbanas de Elche
<https://www.elche.es/wp-content/uploads/2021/05/PLAN-MUNICIPAL-DE-GESTION-DE-COLONIAS-URBANAS-FELINAS-DE-ELCHE.pdf>
Fecha de consulta: 06/03/2023

- [5] Código de Protección y Bienestar Animal
https://www.boe.es/biblioteca_juridica/codigos/codigo.php?id=204&modo=2¬a=0
Fecha de consulta: 07/03/2023
- [6] Las cifras del abandono de perros y gatos aún lejos de descender
<https://www.fundacion-affinity.org/perros-gatos-y-personas/busco-un-animal-de-compania/las-cifras-del-abandono-de-perros-y-gatos-aun>
Fecha de consulta: 07/03/2023
- [7] Protectora del Baix Vinalopó
<https://protectorabaix.org/>
Fecha de consulta: 07/03/2023
- [8] Sociedad protectora para animales de Torre Vieja
<https://satanimalrescue.com/es/>
Fecha de consulta: 07/03/2023
- [9] Fundación para el Asesoramiento y Acción en Defensa de los Animales
<https://faada.org/>
Fecha de consulta: 08/03/2023
- [10] Anuncio de Milanuncios
<https://www.milanuncios.com/otros-animales/camada-de-chihuahuas-adorables-478585386.htm>
Fecha de consulta: 08/03/2023
- [11] HTML - Wikipedia
<https://es.wikipedia.org/wiki/HTML>
Fecha de consulta: 09/03/2023
- [12] PHP- Wikipedia
<https://es.wikipedia.org/wiki/PHP>
Fecha de consulta: 09/03/2023
- [13] JavaScript- Wikipedia
<https://es.wikipedia.org/wiki/JavaScript>
Fecha de consulta: 09/03/2023
- [14] CSS- Wikipedia
<https://es.wikipedia.org/wiki/CSS>
Fecha de consulta: 09/03/2023

- [15] SQL- Wikipedia
<https://es.wikipedia.org/wiki/SQL>
Fecha de consulta: 09/03/2023
- [16] Modelo Vista Controlador (MVC)
<https://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista-controlador-mvc.html>
Fecha de consulta: 10/03/2023
- [17] Ajax
<https://desarrolloweb.com/home/ajax>
Fecha de consulta: 10/03/2023
- [18] MySQL
<https://es.wikipedia.org/wiki/MySQL>
Fecha de consulta: 10/03/2023
- [19] ¿Qué es MySQL? Explicación detallada para principiantes
<https://www.hostinger.es/tutoriales/que-es-mysql>
Fecha de consulta: 10/03/2023
- [20] Visual Studio Code - Wikipedia
https://es.wikipedia.org/wiki/Visual_Studio_Code
Fecha de consulta: 11/03/2023
- [21] Conoce qué es Xampp y por qué deberías usarlo en tus proyectos
<https://www.dongee.com/tutoriales/que-es-xampp/#:~:text=Como%20te%20mencionamos%20antes%2C%20Xampp,facilitan%20la%20experiencia%20al%20desarrollador.>
Fecha de consulta: 11/03/2023
- [22] Bringing MySQL to the web
<https://www.phpmyadmin.net/>
Fecha de consulta: 12/03/2023
- [23] PHPMailer
<https://en.wikipedia.org/wiki/PHPMailer>
Fecha de consulta: 12/03/2023

- [24] GitHub - PHPMailer
<https://github.com/PHPMailer/PHPMailer>
Fecha de consulta: 13/03/2023
- [25] Apuntes de la asignatura de Desarrollo de Servicios Web
Jesús Javier Rodríguez Sala
Fecha de consulta: 24/05/2023
- [26] Flaticon
<https://www.flaticon.com/>
Fecha de consulta: 26/05/2023
- [27] Pixabay
<https://pixabay.com/es/>
Fecha de consulta: 26/05/2023.
- [28] Emojipedia
<https://emojipedia.org/es/>
Fecha de consulta: 26/05/2023
- [29] Ciclo de vida iterativo: qué es, sus ventajas y desventajas
<https://blog.comparasoftware.com/ciclo-de-vida-iterativo-que-es-y-cuales-son-sus-ventajas/>
Fecha de consulta: 20/03/2023
- [30] Ciclo de vida clásico, iterativo y ágil
<https://www.itmplatform.com/es/blog/ciclos-de-vida-clasico-iterativo-y-agil/#:~:text=Los%20ciclos%20de%20vida%20iterativos,parte%20del%20equipo%20del%20proyecto.>
Fecha de consulta: 20/03/2023
- [31] Ciclos de vida iterativo e incremental, ¿Qué son?
[https://www.cursodireccionproyectos.com/2018/07/ciclos-de-vida-iterativo-e-incremental-que-son/#:~:text=disminuye%20la%20incertidumbre\).-,Ciclo%20de%20vida%20incremental.,iteraci%C3%B3n%20se%20le%20denomina%20incremento.](https://www.cursodireccionproyectos.com/2018/07/ciclos-de-vida-iterativo-e-incremental-que-son/#:~:text=disminuye%20la%20incertidumbre).-,Ciclo%20de%20vida%20incremental.,iteraci%C3%B3n%20se%20le%20denomina%20incremento.)
Fecha de consulta: 20/03/2023

Anexo 1

Descripción de los casos de uso

En el presente anexo se describen detalladamente los casos de uso de la aplicación.

Tabla A1.1 - CU1: Acceso a la página web

Cu 1. Acceso a la página web	
Actores	Usuarios registrados, no registrados y administrador.
Descripción	Una persona desea acceder a la aplicación web para poder visualizar todos los contenidos que esta ofrezca.
Dependencias	No
Precondición	Disponer de un dispositivo inteligente y una conexión a internet para poder acceder a la página.
Secuencia normal	1. Acceder a la aplicación. 2. Visualizar el contenido.
Post Condición	
Excepciones	No
Rendimiento	
Frecuencia	Alta
Importancia	Alta
Urgencia	Alta
Estabilidad	
Comentarios	

Tabla A1.2 - CU2: Registro Particular

Cu 2. Registro particular	
Actores	Usuarios no registrados que no sean ni protectoras o refugios.
Descripción	Una persona no registrada en la página desea hacerlo.
Dependencias	No
Precondición	La persona no debe estar registrada previamente y debe de ser mayor de 18 años.
Secuencia normal	<ol style="list-style-type: none"> 1. Acceder a la aplicación. 2. Acceder a la sección de registro de la aplicación. 3. Seleccionar el tipo de usuario del que se trata. <ol style="list-style-type: none"> 3.1. Seleccionar registro como particular. 4. Introducir los datos necesarios para el registro. <ol style="list-style-type: none"> 4.1. Introducir nombre y apellidos 4.2. Introducir correo electrónico. 4.3. Comprobar correo electrónico. 4.4. Introducir contraseña. 4.5. Volver a introducir la contraseña. 4.6. Comprobar si la contraseña es válida y coincide. 4.7. Introducir fecha de nacimiento. 4.8. Comprobar que el usuario es mayor de 18 años. 4.9. Comprobar si se han rellenado todos los campos. 5. Introduce el código de confirmación de registro que se le ha enviado por correo. 6. Registro finalizado.
Post Condición	Ahora el usuario debe cumplir con toda la política de la página en cuanto a sus datos e interacción con la comunidad.
Excepciones	<ol style="list-style-type: none"> 1. Si el correo electrónico no es válido, se vuelve al paso 4.2. 2. Si la contraseña no es válida o no coinciden se vuelve al paso 4.4. 3. Si el usuario no es mayor de 18 años, se vuelve al paso 4.7. 4. Si no se han rellenado todos los campos, se vuelve al paso 5. Si no se confirma el correo mediante el correo de confirmación, no se continuará con el registro y se volverá al paso 3.
Rendimiento	
Frecuencia	Alta
Importancia	Alta
Urgencia	Alta

Tabla A1.3 - CU3: Registro Refugio

Cu 3. Registro Refugio	
Actores	Refugio o protectora no registrada.
Descripción	Un refugio o protectora no registrada desea hacerlo.
Dependencias	No
Precondición	El refugio o protectora no debe estar registrada previamente.
Secuencia normal	<ol style="list-style-type: none"> 1. Acceder a la aplicación. 2. Acceder a la sección de registro de la aplicación. 3. Seleccionar el tipo de usuario del que se trata. <ol style="list-style-type: none"> 3.1. Seleccionar registro como protectora. 4. Introducir los datos necesarios para el registro. <ol style="list-style-type: none"> 4.1. Introducir nombre de la protectora 4.2. Introducir correo electrónico. 4.3. Comprobar correo electrónico. 4.4. Introducir contraseña. 4.5. Volver a introducir la contraseña. 4.6. Comprobar si la contraseña es válida y coincide. 4.7. Introducir fecha de fundación. 4.8. Comprobar fecha de fundación. 4.9. Introducir número de teléfono. 4.10. Comprobar número de teléfono. 4.11. Introducir número de núcleo zoológico. 4.12. Introducir descripción de la protectora. 4.13. Indicar ubicación de la protectora. 4.14. Comprobar si se han rellenado todos los campos. 5. Introduce el código de confirmación enviado por correo. 6. Registro finalizado.
Post Condición	Ahora el usuario debe cumplir con toda la política de la página en cuanto a sus datos e interacción con la comunidad.
Excepciones	<ol style="list-style-type: none"> 1. Si el correo electrónico no es válido, se vuelve al paso 4.2. 2. Si la contraseña no es válida o no coinciden se vuelve al paso 4.4. 3. Si se introduce una fecha que no es válida, se vuelve al paso 4.7. 4. Si no se introduce un teléfono válido, se vuelve al paso 4.9. 5. Si no se han rellenado todos los campos, se vuelve al paso 4.1. 6. Si no se confirma el correo mediante el correo de confirmación, no se continuará con el registro y se volverá al paso 3.
Rendimiento	
Frecuencia	Alta
Importancia	Alta
Urgencia	Alta

Tabla A1.4 - CU4: Iniciar Sesión

Cu 4. Iniciar Sesión	
Actores que intervienen	Usuarios registrados y administrador.
Descripción	Un usuario registrado, un refugio o el administrador desean iniciar sesión en la página web.
Dependencias	No
Precondición	El usuario debe de estar registrado previamente.
Secuencia normal	<ol style="list-style-type: none"> 1. Acceder a la página web. 2. Acceder a la sección de “Iniciar Sesión” 3. Introduce los datos indicados <ol style="list-style-type: none"> 3.1. Introduce el correo 3.2. Introduce la contraseña 3.3. Se comprueban los datos introducidos 4. Hacer clic en el botón de iniciar sesión 5. Sesión Iniciada
Post Condición	El usuario puede acceder a todos los apartados dentro de la página web.
Excepciones	<ol style="list-style-type: none"> 1. Si falta algún campo por rellenar, se volverá al paso 3. 2. Si el usuario o la contraseña no coinciden con ningún usuario registrado, se le indicará al usuario y se volverá al paso 3.
Rendimiento	
Frecuencia	Alta
Importancia	Alta
Urgencia	Alta
Estabilidad	
Comentarios	

Tabla A1.5 - CU5: Recuperar Contraseña

Cu 5. Recuperar Contraseña	
Actores que intervienen	Usuarios registrados y administrador.
Descripción	Un usuario registrado (ya sea particular o protectora) ha olvidado su contraseña y desea recuperarla.
Dependencias	No.
Precondición	El usuario debe de estar registrado previamente.
Secuencia normal	<ol style="list-style-type: none"> 1. Acceder a la página web. 2. Acceder a la sección de “Iniciar Sesión” 3. Hacer clic en el apartado “¿Perdiste tu contraseña?”. 4. Introducir un correo electrónico 5. Comprobar correo electrónico. 6. Introducir el código que se ha enviado a la dirección indicada. 7. Comprobar el código introducido. 8. Introducir la nueva contraseña. 9. Volver a introducir la contraseña. 10. Comprobar si es una contraseña válida y si coinciden. 11. Contraseña cambiada.
Post Condición	El usuario pasará a tener una nueva contraseña y podrá acceder a su cuenta.
Excepciones	<ol style="list-style-type: none"> 1. Si no se introduce un correo registrado, se cancela el proceso y se vuelve al paso 2. 2. Si no se introduce un código válido, se cancela el proceso y se vuelve al paso 2. 3. Si la contraseña introducida no es válida o no coinciden, se vuelve al paso 8.
Rendimiento	
Frecuencia	Baja
Importancia	Baja
Urgencia	Baja
Estabilidad	
Comentarios	

Tabla A1.6 - CU6: Registro de animales

Cu 6. Registro de animales	
Actores que intervienen	Usuarios registrados y administradores.
Descripción	Un usuario registrado desea registrar un animal en adopción dentro de la página.
Dependencias	No
Precondición	El animal no debe de estar registrado previamente.
Secuencia normal	<ol style="list-style-type: none"> 1. Acceder a la página web. 2. Iniciar sesión. 3. Hacer clic en “Registrar Animal” 4. Introducir los datos necesarios para el registro del animal <ol style="list-style-type: none"> 4.1. Introducir el nombre del animal. 4.2. Introducir fecha de nacimiento del animal. 4.3. Comprobar fecha de nacimiento. 4.4. Introducir el sexo del animal. 4.5. Introducir la especie, la raza, el color y el peso. 4.6. Introducir una descripción del animal. 4.7. Introducir sus datos veterinarios. 4.8. Introducir la ubicación donde se encuentra el animal. 4.9. Subir una imagen de la mascota. 4.10. Comprobar que se haya subido una imagen. 4.11. Comprobar si se han rellenado todos los campos. 5. Hacer clic en “Registrar Animal” 6. Registro del animal finalizado
Post Condición	El animal quedará registrado dentro de la aplicación web y se podrán observar sus datos dentro de la misma.
Excepciones	<ol style="list-style-type: none"> 1. Si no se introduce una fecha de nacimiento válida, se vuelve al paso 4.2. 2. Si no se ha subido al menos una imagen de la mascota, se vuelve al paso 4.10. 3. Si no se han rellenado todos los campos, se vuelve al paso 4.1.
Rendimiento	
Frecuencia	Alta
Importancia	Alta
Urgencia	Alta
Estabilidad	
Comentarios	

Tabla A1.7 - CU7: Ver Perfil

Cu 7. Ver Perfil	
Actores que intervienen	Usuarios registrados y administrador.
Descripción	Un usuario registrado o el administrador desea ver sus datos de perfil.
Dependencias	No
Precondición	El usuario debe de estar registrado previamente.
Secuencia normal	1. Acceder a la página web. 2. Acceder a la sección de “Perfil” desde la barra de navegación. 3. Ver los datos del perfil
Post Condición	
Excepciones	
Rendimiento	
Frecuencia	Alta
Importancia	Alta
Urgencia	Alta
Estabilidad	
Comentarios	

Tabla A1.8 - CU8: Eliminar animales

Cu 8. Eliminar animales	
Actores que intervienen	Usuarios registrados y administradores.
Descripción	Un usuario registrado o un administrador desea eliminar un animal en adopción.
Dependencias	No
Precondición	El animal debe de estar registrado por dicho usuario previamente.
Secuencia normal	<ol style="list-style-type: none"> 1. Acceder a la página web. 2. Iniciar sesión. 3. Hacer clic en “Perfil” 4. Observamos los animales que están registrados por dicho usuario. 5. En la sección de “Opciones” hacemos clic en eliminar. 6. En la ventana que se nos mostrará seleccionamos la opción Sí. 5. El animal y todos sus datos habrán sido eliminados.
Post Condición	El animal quedará registrado dentro de la aplicación web y se podrán observar sus datos dentro de la misma.
Excepciones	<ol style="list-style-type: none"> 1. Si el usuario no tiene ningún animal registrado, se le informará. 2. Si el animal ya está adoptado no se podrá eliminar. 3. Si seleccionamos “No” en la ventana de confirmación no se eliminará el animal y se volverá al paso 4.
Rendimiento	
Frecuencia	Baja
Importancia	Media
Urgencia	Media
Estabilidad	
Comentarios	

Tabla A1.9 - CU9: Registrar Adopción

Cu 9. Registrar Adopción	
Actores que intervienen	Usuarios registrados y administradores.
Descripción	Un usuario registrado o un administrador desea registrar la adopción de un animal.
Dependencias	No
Precondición	El animal debe de estar registrado por dicho usuario previamente y se debe haber mantenido alguna conversación sobre dicho animal con algún usuario.
Secuencia normal	<ol style="list-style-type: none"> 1. Acceder a la página web. 2. Iniciar sesión. 3. Hacer clic en “Perfil” 4. Observamos los animales que están registrados por dicho usuario. 5. En la sección de “Opciones” hacemos clic en “Adopción”. 6. En la ventana que se nos mostrará seleccionamos el usuario que va a adoptar el animal. 5. Registramos la adopción. 6. El animal pasará a ser adoptado.
Post Condición	El animal quedará registrado dentro de la aplicación web como adoptado y sus datos no se podrán eliminar.
Excepciones	<ol style="list-style-type: none"> 1. Si el usuario no tiene ningún animal registrado, se le informará. 2. Si el animal ya está adoptado no se podrá modificar. 3. Si seleccionamos “Cerrar” en la ventana de registro de adopción, se cancelará el proceso y se volverá al paso 4.
Rendimiento	
Frecuencia	Media
Importancia	Alta
Urgencia	Alta
Estabilidad	
Comentarios	

Tabla A1.10 - CU10: Visualización de animales

Cu 10. Visualización de animales	
Actores que intervienen	Usuarios registrados, no registrados y administrador.
Descripción	Un usuario registrado o usuarios no registrados desean ver los datos de los animales que hay registrados en la aplicación.
Dependencias	Para poder visualizar animales registrados en la aplicación, deben de haber animales registrados con anterioridad.
Precondición	
Secuencia normal	<ol style="list-style-type: none"> 1. Acceder a la página web. 2. Acceder a la sección “Animales”. 3. Seleccionamos el tipo de animal que queremos ver. 4. Hacemos clic en la ficha del animal que queramos ver. 5. Observamos los datos del animal.
Post Condición	
Excepciones	
Rendimiento	
Frecuencia	Alta
Importancia	Alta
Urgencia	Alta
Estabilidad	
Comentarios	

Tabla A1.11 - CU11: Eliminar cuenta

Cu 11. Eliminar Cuenta	
Actores que intervienen	Usuarios registrados y administrador.
Descripción	Un usuario registrado dentro de la aplicación desea eliminar su cuenta.
Dependencias	No
Precondición	La persona debe de estar registrada dentro de la aplicación web con anterioridad.
Secuencia normal	<ol style="list-style-type: none"> 1. Acceder a la página web. 2. Iniciar sesión. 3. Acceder a la sección de “Perfil” desde la barra de navegación. 4. Hacemos clic en el botón “Eliminar Perfil”. 5. Indicamos que sí queremos borrar nuestra cuenta en la ventana que se nos mostrará. 6. Introducimos el código de confirmación que se nos habrá enviado por correo electrónico. 7. Comprobar si el código introducido es correcto. 8. Usuario eliminado.
Post Condición	Todos los datos del usuario dentro de la aplicación se habrán eliminado.
Excepciones	Si el código introducido no es correcto, se cancelará el proceso y se volverá al paso 6.
Rendimiento	
Frecuencia	Baja
Importancia	Media
Urgencia	Media
Estabilidad	
Comentarios	

Tabla A1.12 - CU12: Crear Chat

Cu 12. Crear Chat	
Actores que intervienen	Usuarios registrados y administrador.
Descripción	Un usuario registrado dentro de la aplicación desea contactar con otro usuario que haya publicado un anuncio sobre su mascota.
Dependencias	No
Precondición	La persona debe de estar registrada dentro de la aplicación web con anterioridad.
Secuencia normal	<ol style="list-style-type: none"> 1. Acceder a la página web. 2. Iniciar sesión. 3. Acceder a la ficha de un animal desde la página de inicio o desde el apartado de visualización de animales. 4. Hacemos clic en el botón “Contacta con el dueño”. 5. Comprobamos que no se trate del mismo usuario que ha registrado el animal o que ya exista un chat entre ambos usuarios. 6. Se crea el chat o se redirige al mismo en el caso de que ya existiera.
Post Condición	Los usuarios pueden entablar una conversación acerca del animal en cuestión para formalizar el proceso de adopción.
Excepciones	Si el usuario que clica el botón es el mismo que ha registrado el animal, se avisará al usuario y no se creará el chat.
Rendimiento	
Frecuencia	Alta
Importancia	Alta
Urgencia	Alta
Estabilidad	
Comentarios	

Tabla A1.13 - CU13: Visualizar Chats

Cu 13. Visualizar Chats	
Actores que intervienen	Usuarios registrados y administrador.
Descripción	Un usuario registrado dentro de la aplicación desea visualizar todas las conversaciones que ha tenido hasta ahora.
Dependencias	No
Precondición	La persona debe de estar registrada dentro de la aplicación web y haber mantenido conversaciones con algún usuario.
Secuencia normal	<ol style="list-style-type: none"> 1. Acceder a la página web. 2. Iniciar sesión. 3. Acceder a la sección de “Mensajes” desde la barra de navegación. 4. Seleccionar el chat que queremos ver. 5. Observar el chat.
Post Condición	Una vez seleccionado el chat, el usuario podrá enviar mensajes o leer la conversación mantenida con el otro usuario.
Excepciones	En el caso de que el usuario no haya mantenido ninguna conversación se le indicará que no existe ningún chat.
Rendimiento	
Frecuencia	Alta
Importancia	Alta
Urgencia	Alta
Estabilidad	
Comentarios	

Tabla A1.14 - CU14: Cerrar sesión

Cu 14. Cerrar sesión	
Actores que intervienen	Usuarios registrados y administrador.
Descripción	Un usuario registrado y que haya iniciado sesión desea cerrar sesión.
Dependencias	No
Precondición	La persona debe de estar registrada dentro de la aplicación web y haberse logueado.
Secuencia normal	<ol style="list-style-type: none"> 1. Acceder a la página web. 2. Iniciar sesión. 3. Hacer clic en el botón “Cerrar Sesión” de la barra de navegación. 4. Sesión cerrada
Post Condición	Una vez cerrada la sesión, el usuario volverá a la página inicial y no tendrá acceso a las mismas opciones que si estuviera logueado.
Excepciones	En el caso de que el usuario no se loguee, nunca le saldrá el botón de cerrar sesión.
Rendimiento	
Frecuencia	Alta
Importancia	Alta
Urgencia	Alta
Estabilidad	
Comentarios	

Tabla A1.15 - CU15: Contactar con el administrador

Cu 15. Contactar con el administrador	
Actores que intervienen	Usuarios registrados y no registrados.
Descripción	Un usuario desea contactar con el administrador de la página para resolver dudas, informar de algún error o realizar distintas gestiones.
Dependencias	No
Precondición	Ninguna
Secuencia normal	<ol style="list-style-type: none"> 1. Acceder a la página web. 2. Localizar la sección “Contactar” que se encuentra en el pie de la página. 3. Rellenar el formulario de contacto. <ol style="list-style-type: none"> 3.1. Introducir el nombre de la persona que contacta. 3.2. Introducir correo electrónico. 3.3. Comprobar correo electrónico 3.4. Introducir el aspecto o duda que se desea tratar. 3.5. Comprobar que se han rellenado todos los campos. 4. Se envía el formulario. 5. Esperar contestación del administrador.
Post Condición	El usuario esperará la contestación por parte del administrador a través de correo electrónico.
Excepciones	<ol style="list-style-type: none"> 1. Si el correo introducido no es correcto, se vuelve al paso 3.2. 2. Si no se han rellenado todos los campos se vuelve al paso 3.
Rendimiento	
Frecuencia	Baja
Importancia	Alta
Urgencia	Media
Estabilidad	
Comentarios	

Anexo 2

Diseño de la base de datos

En el presente anexo se describe el detalle de las tablas de la base de datos.

Animal: Tabla donde almacenaremos la información de los animales que hayan sido registrados en nuestro servidor. Compuesta por los campos:

Tabla A2.1 - Diseño tabla Animal

Tabla Animal	
Id_Animal	Clave Principal, de tipo entero
Nom_Ani	Nombre del animal, varchar de 15 de longitud.
Fecha_Nacimiento	Fecha de nacimiento de la mascota, tipo fecha.
Sexo	Sexo de la mascota, varchar de longitud 1 con dos posibles valores (H = Hembra, M= Macho).
Color	Color de la mascota, varchar de longitud 30.
Peso	Peso de la mascota, de tipo entero.
Id_Usuario	Identificador del usuario que ha registrado dicha mascota, clave ajena procedente de la tabla Usuario.
Id_Municipio	Identificador del municipio de donde procede dicha mascota, clave ajena procedente de la tabla Municipio.
Cod_Raza	Identificador de la raza del animal, se trata de una clave ajena procedente de la tabla Raza.
Desc_Ani	Descripción del animal aportada por el usuario que lo registra, se trata de un campo de tipo text.
Animal (Id_Animal, Nom_Ani, Fecha_Nacimiento, Sexo, Color, Peso, Id_Municipio, Cod_Raza, Desc_Ani) C.P: Id_Animal C.Ajena: Id_Municipio -> Municipio C.Ajena: Id_Usuario -> Usuario C.Ajena: Cod_Raza -> Raza	

Especie: Tabla donde almacenaremos la información de las distintas especies de animales que existen. Compuesta por los campos:

Tabla A2.2 - Diseño tabla Especie

Tabla Especie	
Cod_Especie	Clave Principal, de tipo entero
Nom_Especie	Nombre del animal, varchar de 15 de longitud.
Especie (Cod_Especie, Nom_Especie) C.P: Cod_Especie	

Raza: Tabla donde almacenaremos la información de las distintas razas de animales que podemos tener. Compuesta por los campos:

Tabla A2.3 - Diseño tabla Raza

Tabla Raza	
Cod_Raza	Clave Principal, de tipo entero
Nom_Raza	Nombre del animal, varchar de 15 de longitud.
Cod_Especie	Clave ajena, código de la especie procedente de la relación de existencia entre esta tabla y la de especie.
Raza (Cod_Raza, Nom_Raza, Cod_Especie) C.P: Cod_Raza C.Ajena: Cod_Especie -> Especie VNN	

Comunidad Autónoma: Tabla donde almacenaremos toda la información de las comunidades autónomas españolas. Compuesta por los campos:

Tabla A2.4 - Diseño tabla Comunidad Autónoma

Tabla Comunidad Autónoma	
Id_Com	Clave Principal, de tipo entero
Nom_Com	Nombre de la comunidad autónoma, varchar de 50 de longitud.
Comunidad Autónoma (Id_Com, Nom_Com) C.P: Id_Com	

Provincia: Tabla donde almacenaremos toda la información acerca de las provincias españolas. Compuesta por los campos:

Tabla A2.5 - Diseño tabla Provincia

Tabla Provincia	
Id_Prov	Clave Principal, de tipo entero
Nom_Prov	Nombre de la provincia, varchar de 30 de longitud.
Id_Com	Identificador de la comunidad autónoma donde se halle dicha provincia, se trata de una clave ajena procedente de la tabla “Comunidad Autónoma”
Provincia (Id_Prov, Nom_Prov, Id_Com) C.P: Id_Prov C.Ajena: Id_Com -> Comunidad Autónoma	

Municipio: Tabla donde almacenaremos toda la información acerca de los municipios españoles. Compuesta por los campos:

Tabla A2.6 - Diseño tabla Municipio

Tabla Municipio	
Id_Mun	Clave Principal, de tipo entero
Nom_Muni	Nombre del municipio, varchar de 30 de longitud.
Id_Prov	Identificador de la provincia donde se halle dicha localidad, se trata de una clave ajena procedente de la tabla "Provincia"
Localidad (Id_Mun, Nom_Muni, Id_Prov) C.P: Id_Mun C.Ajena: Id_Prov -> Provincia	

Usuario: Tabla donde almacenaremos toda la información acerca de los usuarios que usen la aplicación. Compuesta por

Tabla A2.7 - Diseño tabla Usuario

Tabla Usuario	
Id_Usuario	Clave Principal, de tipo entero
Tipo_Usu	Indicará el tipo de usuario del que se trata, tendrá un 1 cuando sea particular y un 0 cuando sea refugio
Nom_Usuario	Nombre del Usuario, campo de tipo varchar de 50 de longitud
Email	Cuenta de correo asociada al usuario, campo de tipo varchar de 50 de longitud
Fech_Naci_Usu	Fecha de nacimiento del usuario o de la creación del refugio, campo de tipo date.
Contra	Contraseña del usuario, campo de tipo varchar de 50 de longitud
Usuario (Id_Usuario Tipo_Usu, Nom_Usuario, Email, Fech_Naci_Usu) C.P: Id_Usuario	

Particular: Tabla donde almacenaremos toda la información acerca de los usuarios particulares que usen la aplicación. Compuesta por los campos:

Tabla A2.8 - Diseño tabla Particular

Tabla Particular	
Id_Usuario	Clave Principal, de tipo entero. Se trata también de una clave ajena procedente de la especialización de la tabla usuario.
Particular (Id_Usuario) C.P: Id_Usuario C. Ajena: Id_Usuario -> Usuario VNN	

Refugio: Tabla donde almacenaremos toda la información acerca de los refugios que usen la aplicación. Compuesta por los campos:

Tabla A2.9 - Diseño tabla Refugio

Tabla Refugio	
Id_Usuario	Clave Principal, de tipo entero. Se trata también de una clave ajena procedente de la especialización de la tabla usuario.
Id_Municipio	Identificador del municipio donde se ubica dicho refugio, clave ajena procedente de la tabla municipio
Telefono	Teléfono del refugio, se trata de un campo de tipo entero
Desc_Refu	Descripción del refugio, campo de tipo text
Nucleo_Zoolo gico	Campo de tipo text que recoge el número de núcleo zoológico del refugio.
Refugio (Id_Usuario, Id_Municipio, Telefono, Desc_Refu, Núcleo_Zoologico) C.P: Id_Usuario C. Ajena: Id_Municipio -> Municipio VNN	

Veterinario: Tabla donde almacenaremos toda la información veterinaria de los animales registrados en la aplicación. Compuesta por

Tabla A2.10 - Diseño tabla Veterinario

Tabla Veterinario	
Id_Animal	Clave Principal, de tipo entero. Se trata de una clave ajena procedente de la relación de existencia entre la tabla veterinario y animal
Observaciones	Identificador de la localidad de donde se ubica dicho refugio, clave ajena procedente de la tabla localidad. Se trata de un campo de tipo text
Castrado	Tendrá el valor 1 si la mascota está castrada y 0 si no lo está.
Vacunas	Campo de tipo text con las vacunas administradas al animal.
Veterinario (Id_Animal, Observaciones, Castrado, Vacunas) C.P: Id_Animal C. Ajena: Id_Animal -> Animal VNN	

Fotos: Tabla donde almacenaremos toda la información acerca de las fotos de los animales registrados en la aplicación. Compuesta por los campos:

Tabla A2.11 - Diseño tabla Fotos

Tabla Fotos	
Id_Foto	Clave Principal, de tipo entero. Identificador de la foto dentro de la base de datos
Nom_Fich	Nombre de la imagen que se encuentra almacenada. Se trata de un campo de tipo varchar de 50 de longitud.
Id_Mas	Clave ajena procedente de la tabla mascota. Se trata del identificador de la mascota a la cual está asociada la imagen.
Ext_Fich	Extension de la imagen que se almacenará en la base de datos
Fotos (Id_Foto, Nom_Arch, Id_Mas, Ext_Fich) C.P: Id_Foto C. Ajena: Id_Mas -> Animal	

Chat: Tabla donde almacenaremos toda la información acerca de los chats que mantienen los usuarios registrados en la aplicación. Compuesta por los campos:

Tabla A2.12 - Diseño tabla Chat

Tabla Chat	
Id_Chat	Clave Principal, de tipo entero. Identificador del chat dentro de la base de datos
Id_Usu1	Clave ajena procedente de la tabla Usuario. Se trata del identificador del usuario propietario de la mascota.
Id_Usu2	Clave ajena procedente de la tabla Usuario. Se trata del identificador del usuario que está interesado en la adopción de la mascota.
Id_Mas	Clave ajena procedente de la tabla Animal. Se trata del identificador del animal sobre el que los usuarios anteriores están manteniendo una conversación.
Chat (Id_Chat, Id_Usu1, Id_Usu2, Id_Mas) C.P: Id_Chat C. Ajena: Id_Usu1 -> Usuario C. Ajena: Id_Usu2 -> Usuario C. Ajena: Id_Mas -> Animal	

Mensaje: Tabla donde almacenaremos todos los mensajes de los distintos chats que los usuarios mantengan. Compuesta por los campos:

Tabla A2.13 - Diseño tabla Mensaje

Tabla Mensaje	
Id_Mensaje	Clave Principal, de tipo entero. Identificador del mensaje dentro de la base de datos
Id_Chat	Clave ajena procedente de la tabla Chat. Se trata del identificador del chat al que pertenece el mensaje.
Mensaje	Campo que recoge el mensaje escrito por el usuario. Se trata de un campo de tipo text.
Visto	Campo de tipo entero que recoge si un mensaje ha sido leído por el usuario receptor o no. Contendrá el valor 1 en el caso de que lo haya leído y el valor 0 en el caso de que no.
Fecha_Hora	Campo de tipo timestamp que recoge la fecha y la hora a la que se envió el mensaje.
Autor	Nombre del usuario que envía el mensaje, se trata de un campo de tipo varchar de 50 caracteres de longitud.
Mensaje (Id_Mensaje, Id_Chat, Mensaje, Visto, Fecha_Hora, Autor) C.P: Id_Mensaje C. Ajena: Id_Chat -> Chat VNN	

Adopción: Tabla donde almacenaremos todos los datos relativos a un proceso de adopción dentro de la aplicación. Compuesta por los campos:

Tabla A2.14 - Diseño tabla Adopción

Tabla Adopción	
Id_Adopcion	Clave Principal, de tipo entero. Identificador del proceso de adopción dentro de la base de datos.
Id_Antiguo_Propietario	Clave ajena procedente de la tabla Usuario. Se trata del identificador del usuario al que pertenecía la mascota.
Id_Nuevo_Propietario	Clave ajena procedente de la tabla Usuario. Se trata del identificador del usuario que ha adoptado la mascota.
Id_Animal	Clave ajena procedente de la tabla Animal. Se trata del identificador de la mascota que ha sido adoptada.
Fecha_Adopción	Campo de tipo timestamp que recoge la fecha y la hora a la que se realizó la adopción.
Adopción (Id_Adopción, Id_Antiguo_Propietario, Id_Nuevo_Propietario, Id_Animal, Fecha_Adopción) C.P: Id_Adopcion C. Ajena: Id_Antiguo_Propietario -> Usuario C. Ajena: Id_Nuevo_Propietario -> Usuario C. Ajena: Id_Animal -> Animal	