

CENTRO INTEGRADO DE FORMACIÓN  
PROFESIONAL N°1 DE CEUTA

GRADO SUPERIOR DE DESARROLLO DE  
APLICACIONES MULTIPLATAFORMA

PROYECTO FINAL DE GRADO

AdoptaTuPet

Jesús Roca Obispo

CURSO 2024/2025

# RESUMEN

AdoptaTuPet es una app para móviles Android que muestra perros y gatos en adopción de varias protectoras. Permite buscarlos con filtros, ver fotos, leer su historia y hablar en un foro. También incluye un mapa con refugios y avisa cuando hay nuevos animales.

## **Palabras clave**

Adopción, animales, android, foro, aplicación móvil.

# ABSTRACT

AdoptaTuPet is an Android mobile app that shows dogs and cats available for adoption from different shelters. It lets users search with filters, see photos, read their stories, and chat in a forum. It also includes a map of shelters and alerts for new animals.

## **Keywords**

Adoption, animals, Android, forum, mobile application.

# ÍNDICE

<b>1. INTRODUCCIÓN</b>	<b>4</b>
<b>2. JUSTIFICACIÓN DEL PROYECTO</b>	<b>4</b>
<b>3. OBJETIVOS</b>	<b>5</b>
3.1 Objetivo general:	5
3.2 Objetivos específicos:	6
<b>4. ALTERNATIVAS</b>	<b>6</b>
4.1 Miwuki Pet Shelter	6
4.2 AdoptaMe	7
4.3 Amazdog	7
4.4 Redes Sociales (Facebook, Instagram, Tiktok...)	8
4.5 Petfinder	8
<b>5. ANÁLISIS DAFO</b>	<b>9</b>
<b>6. ANÁLISIS</b>	<b>9</b>
6.1 Requisitos Funcionales	9
6.2 Requisitos No Funcionales	10
6.3 Casos de Uso	11
<b>7. PLANIFICACIÓN</b>	<b>12</b>
7.1 Planificación del Proyecto	12
7.2 Fases de desarrollo	13
7.3 Diagrama de GANTT	14
7.4 Tiempo de inversión	14
<b>8. DISEÑO</b>	<b>16</b>
8.1 Arquitectura y metodología de desarrollo	16
8.2 Diagrama de Casos de Uso	17
8.3 Diagrama Entidad - Relación	17
8.4 Diagrama de Clases	18
8.5 Prototipo	19
8.6 Tipografía y estilo	24
8.7 Herramientas utilizadas en el proyecto	24
<b>9. DESARROLLO</b>	<b>26</b>
9.1 Aspectos relevantes del desarrollo	26
9.2 Metodologías	27
9.3 Técnicas	27
9.4 Tecnologías	28
<b>10. CONCLUSIONES</b>	<b>29</b>
<b>BIBLIOGRAFÍA</b>	<b>30</b>
<b>ANEXO I: GLOSARIO</b>	<b>32</b>

# 1.INTRODUCCIÓN

El abandono de mascotas, sobre todo perros y gatos, es un problema muy común en muchas ciudades y pueblos. Cada año, muchos animales son abandonados o se pierden (Fundación Affinity, s.f.), y los refugios y protectoras no tienen suficiente espacio ni recursos para cuidarlos bien (Aires Machado, 2025). Esto afecta mucho a los animales, pero también a las personas que trabajan en estos centros y a la comunidad en general. Además, muchas protectoras no tienen suficiente dinero ni visibilidad para conseguir que los animales sean adoptados.

Por eso, la tecnología, especialmente las aplicaciones para móviles, puede ayudar a mejorar esta situación (López, 2024). De esta idea nace AdoptaTuPet, una aplicación móvil que hace más fácil adoptar mascotas. El objetivo de esta app es ofrecer un espacio sencillo y agradable para que cualquier persona pueda ver qué animales están buscando hogar, tener información completa sobre ellos y ponerse en contacto con las protectoras rápidamente.

AdoptaTuPet no es solo una app para ver animales, también quiere enseñar a las personas la importancia de adoptar en vez de comprar mascotas. Además, busca que la gente sea más responsable con sus animales, informe sobre la esterilización y ayude a sentir más empatía hacia mascotas que tienen más dificultades para ser adoptadas, como animales mayores o con necesidades especiales.

La aplicación también tiene herramientas útiles para que los administradores y voluntarios puedan gestionar fácilmente la información de los animales, actualizarla constantemente y seguir el proceso de adopción sin problemas. Cuenta también con mapas interactivos para encontrar rápidamente las protectoras, formularios fáciles para adoptar mascotas y notificaciones que mantienen informados a los usuarios en todo momento.

## 2.JUSTIFICACIÓN DEL PROYECTO

Este proyecto existe porque hay muchos animales abandonados que necesitan una familia (Fundación Affinity, s.f.). Aunque hay muchas protectoras, refugios y perreras que los recogen y cuidan, la mayoría trabaja por su cuenta (FAADA, s.f.). Cada una tiene su página web o redes sociales, pero no están conectadas entre ellas. Esto hace que las personas que quieren adoptar no siempre encuentren fácilmente a los animales, o ni siquiera sepan que hay uno cerca que necesita ayuda (Sánchez, 2025).

Por eso se ha creado una aplicación para el móvil que se llama AdoptaTuPet. La idea es juntar en un solo sitio a todas estas asociaciones para que cualquier persona pueda ver desde el móvil todos los animales que están buscando un hogar, sin tener que buscar en muchas webs distintas o ir de refugio en refugio.

Dentro de la app se pueden ver fotos de los animales y datos como su edad, tamaño, raza y dónde están. También hay un botón para hablar directamente con la protectora que lo cuida, por si alguien quiere saber más o empezar el proceso de adopción.

Además, AdoptaTuPet ayuda a las protectoras a trabajar juntas. Todas pueden subir sus animales a la app, compartir información y organizarse mejor. Así, los animales tienen más posibilidades de ser adoptados y las asociaciones lo tienen más fácil para colaborar.

La app también tiene un foro donde cualquier persona puede escribir. Se puede usar para avisar si alguien ha visto un animal perdido, dar consejos, pedir ayuda o compartir ideas. Es un espacio abierto para que la gente y las protectoras puedan comunicarse entre sí.

También se pueden usar filtros para buscar animales según el tipo, la edad, el tamaño o la ciudad. Incluso se puede activar la ubicación del móvil para ver si hay animales cerca que necesitan adopción.

AdoptaTuPet es fácil de usar, funciona desde el móvil y no hace falta saber mucho de tecnología. El objetivo es que adoptar sea algo sencillo y que más animales puedan encontrar una familia.

## 3.OBJETIVOS

Antes de comenzar con el desarrollo de la aplicación AdoptaTuPet, era importante tener claro qué se quería conseguir. Este proyecto nace con la intención de ofrecer una solución sencilla y accesible que facilite el proceso de adopción de animales. Para ello, se planteó un objetivo general muy concreto, acompañado de una serie de objetivos específicos que ayudan a definir mejor cómo se pretende alcanzar esa meta. A continuación, se detallan estos objetivos.

### 3.1 Objetivo general:

Hacer que adoptar perros y gatos sea más fácil usando una aplicación para el móvil que junte en un solo sitio a todas las asociaciones y ayude a las personas a encontrar animales que buscan una familia.

## 3.2 Objetivos específicos:

- Reunir en la app a protectoras, refugios, perreras y gateras para que más gente vea a los animales que están en adopción.
- Dejar que cada persona busque animales según lo que quiera: por edad, tamaño, raza o ciudad.
- Tener un foro donde se puedan compartir mensajes, avisos, consejos o experiencias sobre animales.
- Hacer más fácil que las personas hablen con las asociaciones para adoptar.
- Ayudar a que haya menos abandono de animales, haciendo que adoptar sea algo rápido y seguro.
- Dar más oportunidades a los animales que casi nadie adopta, como los que ya son mayores.
- Animar a la gente a adoptar en vez de comprar animales, y explicar por qué eso es importante.

## 4.ALTERNATIVAS

Antes de diseñar AdoptaTuPet, se buscó qué herramientas ya usan las protectoras en España para dar a conocer a los animales que tienen en adopción. Se revisaron cuatro aplicaciones móviles: Miwuki Pet Shelter (<https://petshelter.miwuki.com/>), AdoptaMe (<https://adopta-me.com/>), Amazdog (<https://amazdog.com/>) y la internacional Petfinder (<https://www.petfinder.com/>), y el uso que hacen muchas protectoras de las redes sociales más populares (Facebook, Instagram y TikTok). De cada opción se anotaron las funciones que ofrece, sus ventajas y sus desventajas. Con esta información se puede ver qué necesidades siguen sin cubrirse y cómo una nueva app puede mejorarlas.

### 4.1 Miwuki Pet Shelter

Miwuki Pet Shelter (<https://petshelter.miwuki.com/>) es una plataforma que permite a los usuarios encontrar animales en adopción, como perros, gatos, conejos, hurones, entre otros. La aplicación ofrece filtros de búsqueda por especie, edad, tamaño y ubicación, facilitando el contacto directo con las protectoras.

#### **Ventajas:**

- Amplia base de datos de animales en adopción.

- Filtros de búsqueda detallados.
- Contacto directo con las protectoras.

**Desventajas:**

- No incluye funcionalidades de foro o comunidad para interacción entre usuarios.
- La interfaz puede resultar compleja para usuarios con poca experiencia tecnológica.

## 4.2 AdoptaMe

AdoptaMe (<https://adopta-me.com/>) es una aplicación española que conecta a usuarios con animales en adopción. Desde su lanzamiento, ha registrado más de 75.000 descargas y cuenta con la colaboración de múltiples protectoras en todo el país.

**Ventajas:**

- Enfoque local y nacional, facilitando la adopción en distintas regiones.
- Interfaz sencilla y fácil de usar.
- Actualizaciones frecuentes con nuevos animales disponibles.

**Desventajas:**

- Disponible únicamente para dispositivos Android.
- No ofrece una sección de foro o comunidad para compartir experiencias o consejos.

## 4.3 Amazdog

Amazdog (<https://amazdog.com/>) es una plataforma que proporciona información sobre refugios y asociaciones cercanas, permitiendo a los usuarios ver animales disponibles para adopción y ofrecer ayuda.

**Ventajas:**

- Información detallada sobre refugios y asociaciones locales.
- Facilita la conexión entre usuarios y organizaciones.

**Desventajas:**

- No cuenta con una base de datos extensa de animales en adopción.
- Carece de funcionalidades interactivas como foros o comunidades.

## 4.4 Redes Sociales (Facebook, Instagram, Tiktok...)

Muchas protectoras y refugios utilizan redes sociales como Facebook, Instagram, Tiktok... para publicar animales en adopción, compartir historias y eventos relacionados.

### Ventajas:

- Amplia difusión y alcance.
- Interacción directa con los usuarios.
- Actualizaciones frecuentes y en tiempo real.

### Desventajas:

- Información dispersa y no centralizada.
- Dificultad para buscar animales específicos.
- Posibilidad de fraudes o información no verificada (Peters, 2025).

## 4.5 Petfinder

Petfinder (<https://www.petfinder.com/>) es una aplicación internacional que permite buscar animales en adopción en miles de refugios y grupos de rescate. Ofrece filtros por ubicación, raza, edad, tamaño y sexo.

### Ventajas:

- Amplia base de datos a nivel internacional.
- Filtros de búsqueda avanzados.

### Desventajas:

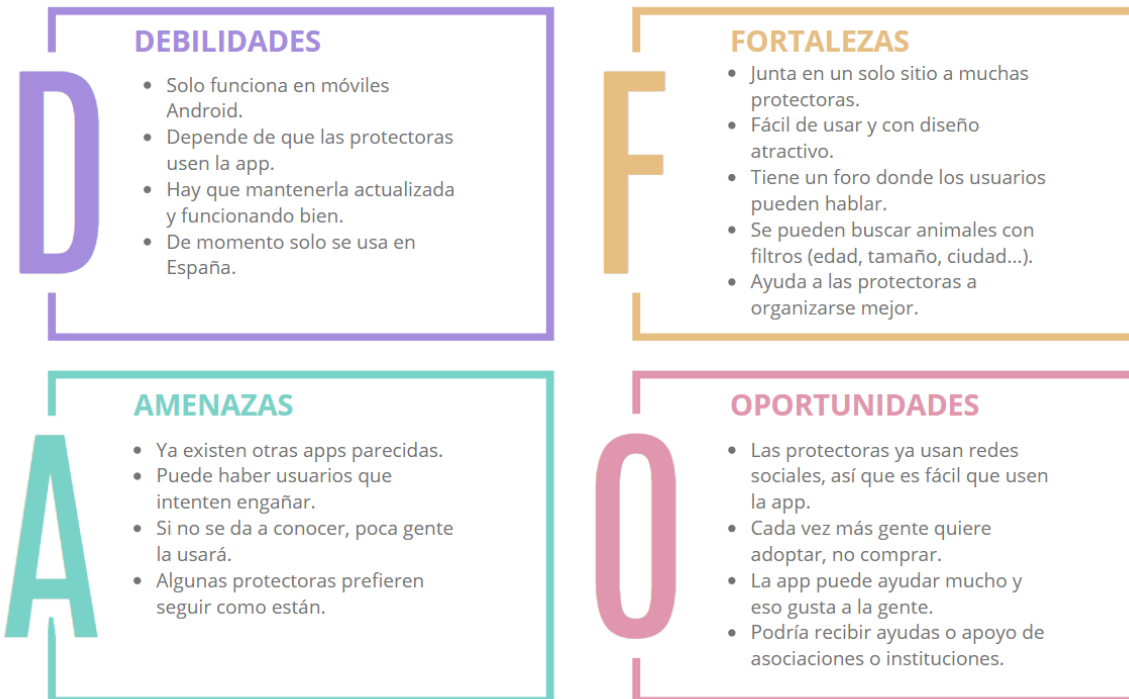
- Enfocada principalmente en el mercado estadounidense, con menor presencia en España.
- Interfaz en inglés, lo que puede dificultar su uso para algunos usuarios hispanohablantes.

Una de las cosas que diferencia a AdoptaTuPet de otras aplicaciones es que tiene un **foro dentro de la propia app**. En este espacio, cualquier persona puede escribir mensajes para dar consejos, avisar si ha visto un animal perdido, pedir ayuda o compartir su experiencia con la adopción. Las otras aplicaciones que existen en España, como Miwuki, AdoptaMe o Amazdog, **no tienen esta función de foro**. Solo permiten ver animales o contactar con las protectoras, pero no tienen un lugar para que los usuarios hablen entre ellos. Por eso, el foro de AdoptaTuPet es algo que la



hace única y que puede ayudar mucho a crear una comunidad más unida en torno a la adopción de animales.

## 5. ANÁLISIS DAFO



## 6. ANÁLISIS

### 6.1 Requisitos Funcionales

#### **RF01 - Registro de usuarios**

La aplicación permitirá a los usuarios crear una cuenta mediante un formulario de registro.

#### **RF02 - Inicio de sesión**

Los usuarios podrán acceder a su cuenta introduciendo sus credenciales previamente registradas.

#### **RF03 - Publicación de animales en adopción**

Los usuarios autorizados podrán subir animales a la plataforma, especificando: edad, raza, sexo, localidad, tamaño, descripción y una fotografía.

**RF04 - Visualización de animales disponibles**

Los usuarios registrados podrán consultar el listado de animales en adopción, accediendo a su información detallada.

**RF05 - Proceso de adopción**

Los usuarios registrados podrán iniciar el proceso de adopción seleccionando un animal y enviando una solicitud a la protectora responsable.

**RF06 - Filtro de búsqueda**

El sistema incluirá filtros por especie, raza, tamaño y localidad para facilitar la búsqueda de animales.

**RF07 - Foro de la comunidad**

Los usuarios registrados podrán participar en un foro donde compartir experiencias, dudas, eventos y otras informaciones relacionadas con los animales.

**RF08 - Información de contacto**

En cada publicación se mostrará la información de contacto y ubicación de la persona o protectora responsable del animal.

**RF09 - Edición de perfil**

Los usuarios podrán modificar su nombre, correo electrónico y foto de perfil desde su cuenta.

**RF10 - Gestión de publicaciones**

Los administradores podrán editar o eliminar publicaciones incorrectas o no autorizadas.

## 6.2 Requisitos No Funcionales

**RNF01 - Seguridad**

El sistema garantizará la protección de las contraseñas mediante cifrado y asegurará la confidencialidad de los datos personales.

**RNF02 - Rendimiento**

La aplicación ofrecerá una experiencia fluida, con tiempos de carga reducidos y una navegación ágil.

**RNF03 - Compatibilidad**

La aplicación será compatible con dispositivos Android desde la versión 8.0 (Oreo) en adelante.

**RNF04 - Escalabilidad**

La arquitectura de la base de datos estará preparada para soportar un aumento significativo en el número de usuarios y publicaciones.

**RNF05 - Diseño atractivo**

La interfaz contará con un diseño visual claro, atractivo y accesible para todo tipo de usuarios.

**RNF06 - Mantenimiento**

La estructura del código permitirá su fácil mantenimiento y escalado mediante el uso de buenas prácticas de organización.

**RNF07 - Disponibilidad**

El sistema estará diseñado para ofrecer alta disponibilidad y tolerancia a fallos, mostrando mensajes de error cuando sea necesario.

## 6.3 Casos de Uso

**CU01 - Registrarse**

Caso de uso principal.

**CU02 - Iniciar sesión**

Caso de uso principal.

**CU03 - Visualizar contenido**

Caso de uso principal.

**CU03.1 - Foro**

Extensión desde "Visualizar contenido".

**CU03.2 - Anuncios**

Extensión desde "Visualizar contenido".

**CU03.3 - Información de la página**

Extensión desde "Visualizar contenido".

**CU03.4 - Animales en adopción**

Extensión desde "Visualizar contenido".

**CU04 - Añadir contenido**

Caso de uso principal.

**CU04.1 - Post en Foro**

Extensión desde "Añadir contenido".

**CU04.2 - Subir anuncios**

Extensión desde "Añadir contenido".

**CU04.3 - Subir adopción**

Extensión desde "Añadir contenido".

**CU05 - Cerrar sesión**

Caso de uso principal.

## 7. PLANIFICACIÓN

Antes de comenzar con el desarrollo de AdoptaTuPet, se estableció una planificación detallada que guiara cada paso del proyecto. Esta organización previa fue clave para asegurar una ejecución ordenada, permitiendo distribuir las tareas en fases, asignar tiempos realistas y mantener el control del avance. A continuación, se describen los distintos momentos del proceso, desde la definición inicial hasta la entrega final.

### 7.1 Planificación del Proyecto

Para crear AdoptaTuPet seguimos una serie de pasos ordenados. Así avanzamos sin perdernos y siempre supimos qué tocaba después.

#### **1. Definir el problema y los requisitos**

Primero aclaramos cuál era la dificultad: adoptar animales no resulta tan sencillo como debería. Miramos otras aplicaciones parecidas, vimos lo que funcionaba y lo que se podía mejorar, y anotamos todo lo que nuestra app debía ofrecer.

#### **2. Diseñar la apariencia**

Hicimos bocetos rápidos de las pantallas, elegimos colores y tipografías y redactamos una pequeña guía de estilo. Esto nos ayudó a mantener la misma imagen en toda la aplicación.

#### **3. Desarrollar las funciones básicas**

Abrimos el proyecto en Android Studio y construimos la base. Empezamos por lo esencial (registro, inicio de sesión y listado de animales) y, cuando eso funcionó, añadimos el foro, los filtros y los formularios de contacto. Probamos cada parte en cuanto la terminamos.

#### **4. Conectar servicios externos**

Integramos Google Maps para mostrar las protectoras en un mapa y configuramos el sistema para guardar imágenes y datos de forma rápida y segura.

#### **5. Probar la aplicación**

Revisamos cada módulo por separado y luego usamos la app como lo haría cualquier persona. Pedimos a compañeros que la probaran para detectar fallos y recopilar sugerencias.

## **6. Preparar la entrega**

Dejamos lista una versión que podría subirse a Google Play, escribimos la documentación técnica y creamos una guía de usuario sencilla.

## **7. Planificar mejoras futuras**

Finalmente apuntamos ideas para próximas versiones y preparamos la estructura para que sea fácil añadir funciones nuevas o corregir errores más adelante.

# **7.2 Fases de desarrollo**

A la hora de realizar la aplicación AdoptaTuPet, el trabajo se ha dividido en varias fases.

### **Fase 1: Pensar qué se va a realizar.**

Lo primero fue tener claro de qué iba la app y qué queríamos que hiciera. Se miraron otras apps parecidas para ver qué tenían, y se apuntó lo que gustaba y lo que no tanto. También se hizo una lista con todo lo que debería tener la app, como por ejemplo ver animales, un foro, o la adopción.

### **Fase 2: Diseño**

Después se hicieron dibujos de cómo serían las pantallas (los típicos bocetos con cajas). Se eligieron los colores principales, el tipo de letra y cómo moverse por la app. También se hizo una especie de guía con todo esto para que luego el diseño se mantuviera igual en todas las pantallas.

### **Fase 3: Empezar a programar**

Con todo pensado, se arrancó el proyecto en Android Studio. Primero se hicieron las cosas más importantes: ver animales, iniciar sesión, registrarse. Luego se fueron metiendo las demás partes como el foro, el formulario para adoptar o el mapa. Cada vez que se terminaba una parte, se probaba antes de seguir.

### **Fase 4: Unirlo todo**

Aquí ya se fue juntando todo lo hecho, como Google Maps, base de datos, fotos... Se revisó que todo estuviera conectado bien y que no diera errores.

### **Fase 5: Probar**

Ya con la app funcionando, se probó tanto en el emulador como en un móvil de verdad. Se miró que al pulsar botones o escribir cosas todo respondiera bien.

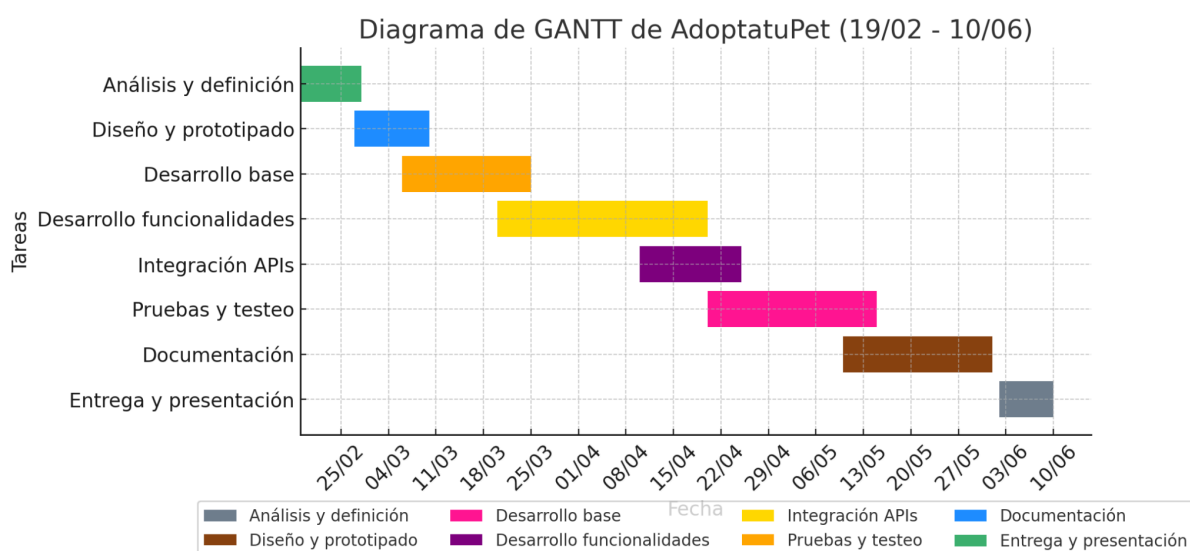
También se pidió a un par de personas que la probaran y dijeran si entendían cómo usarla.

### Fase 6: Entregar

Por último, se preparó todo para la entrega del proyecto. Se escribió la memoria, una pequeña guía explicando cómo va la app y se hicieron capturas para enseñar el resultado. Si en algún momento se quiere subir a Google Play, ya está lista para eso también.

## 7.3 Diagrama de GANTT

A continuación se muestra el diagrama de GANTT del proyecto AdoptaTuPet, donde se representa visualmente la planificación temporal de cada una de las fases del desarrollo. Este cronograma permitió organizar las tareas de forma clara y controlar el progreso del proyecto desde el inicio hasta su entrega final.



## 7.4 Tiempo de inversión

Durante el desarrollo de AdoptaTuPet, el equipo organizó el trabajo en fases claramente definidas, que se desarrollaron entre el 19 de febrero y el 10 de junio de 2025. A continuación, se detalla brevemente cada una:

### Fase 1. Análisis y definición

**Duración:** 10 días (19 al 28 de febrero)

En esta fase se definió la idea del proyecto, se investigaron apps similares y se identificaron las necesidades de los usuarios clave (adoptantes, protectoras,

visitantes). A partir de esta información se elaboraron los primeros requisitos y un esquema funcional inicial.

## **Fase 2. Diseño y prototipado**

**Duración:** 12 días (27 de febrero al 10 de marzo)

Se diseñaron los primeros bocetos en Figma, definiendo la estética general de la app. Se creó una guía de estilo visual y se generó un prototipo navegable que sirvió como referencia para el desarrollo posterior.

## **Fase 3. Desarrollo base**

**Duración:** 20 días (6 al 25 de marzo)

Se configuró el proyecto técnico y se implementaron las funcionalidades básicas: autenticación, navegación y visualización de animales. Se estableció la estructura base de la app y la conexión inicial con Firebase.

## **Fase 4. Desarrollo de funcionalidades**

**Duración:** 31 días (20 de marzo al 20 de abril)

Se implementaron funcionalidades avanzadas como la gestión de publicaciones, la edición de perfiles y el buscador. También se reforzó la estructura de datos y se añadieron validaciones.

## **Fase 5. Integración de APIs**

**Duración:** 15 días (10 al 25 de abril)

Se integraron servicios externos como Google Maps y se conectó la app a una API REST. Esta fase también incluyó la carga de imágenes y otros elementos interactivos.

## **Fase 6. Pruebas y testeo**

**Duración:** 25 días (20 de abril al 15 de mayo)

Se realizaron pruebas funcionales en varios dispositivos, resolviendo errores detectados por el equipo y usuarios externos. Se evaluó la experiencia de usuario y se aplicaron mejoras.

## **Fase 7. Documentación**

**Duración:** 22 días (10 de mayo al 1 de junio)

Se documentó el proyecto en detalle, incluyendo análisis, capturas, funcionamiento de la app y arquitectura técnica. También se preparó todo para la entrega final.

## Fase 8. Entrega y presentación

**Duración:** 9 días (2 al 10 de junio)

Se realizó la entrega final del proyecto y se preparó la presentación. Esta etapa incluyó revisiones finales y ensayos para exponer el trabajo de forma clara y profesional.

# 8. DISEÑO

Al diseñar AdoptaTuPet se siguió un camino sencillo: primero se eligió una base técnica clara (MVC y un servidor que guarda los datos mientras la app muestra las pantallas); luego se hicieron dibujos y diagramas para ver qué puede hacer cada usuario y cómo se guardará la información; después se creó un boceto rápido de las pantallas para pulir la navegación y los colores; se decidió usar la letra Fredoka y una mezcla de naranja y azules suaves para que todo sea amable; y por último, con herramientas como Android Studio, Retrofit, Firebase, Google Maps y GitHub, se fue construyendo, probando y publicando la app paso a paso.

## 8.1 Arquitectura y metodología de desarrollo

Para que el desarrollo de la aplicación AdoptaTuPet fuese más claro, organizado y fácil de mantener, se ha utilizado el modelo MVC, que significa Modelo - Vista - Controlador. Esta forma de trabajar ayuda a separar cada parte del proyecto: una parte se encarga de los datos, otra de lo que ve el usuario y otra de las acciones. Gracias a esto, si en el futuro hay que cambiar algo (por ejemplo, el diseño de una pantalla o la forma de guardar los datos), solo hay que modificar una parte y no todo el proyecto (Microsoft, 2024).

El **Modelo** es donde se guardan los datos: los animales, los usuarios, los mensajes del foro, etc.

La **Vista** es lo que el usuario ve en la pantalla del móvil. Está hecha con archivos XML, donde se colocan los botones, textos e imágenes.

El **Controlador** es el que recibe lo que el usuario hace (por ejemplo, pulsar "Adoptar") y decide qué debe pasar después: si tiene que buscar datos, guardarlos o mostrar algo nuevo en pantalla.

Además de usar el modelo MVC, también se ha trabajado con una **arquitectura cliente-servidor**. Esto quiere decir que la app (el cliente) no guarda todos los datos por sí sola, sino que se conecta a un servidor para conseguir o enviar información. Por ejemplo, cuando el usuario busca un animal, la app le pide esos datos al servidor (IBM Developer, 2021).

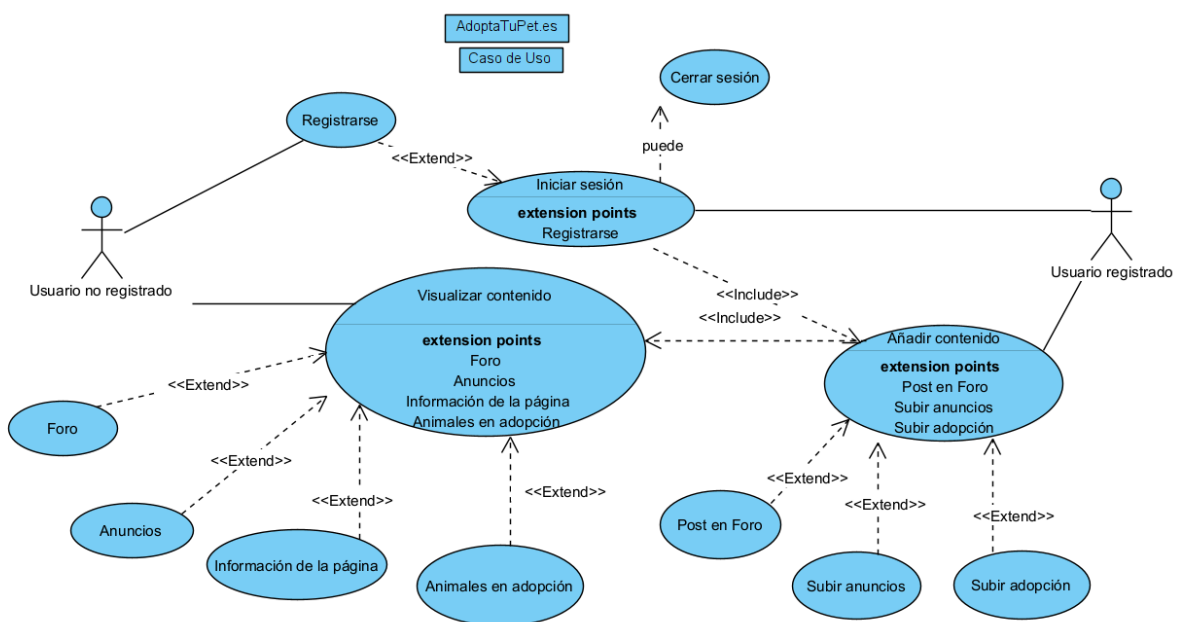


Para hacer esto, se ha usado **Retrofit**, que permite enviar mensajes desde la app al servidor usando internet. El servidor está hecho en **PHP** y se conecta a una base de datos en **MySQL**. Así, cuando la app pide algo (como el listado de animales), el servidor busca los datos y se los manda a la app para que se muestren al usuario.

Durante todo el desarrollo se ha seguido una **metodología iterativa**, donde cada parte se programaba, se probaba y se corregía antes de seguir con la siguiente. También se ha utilizado la técnica de **programación modular**, dividiendo el proyecto en bloques pequeños (como pantallas, funciones o controladores) para facilitar la organización y el mantenimiento del código.

## 8.2 Diagrama de Casos de Uso

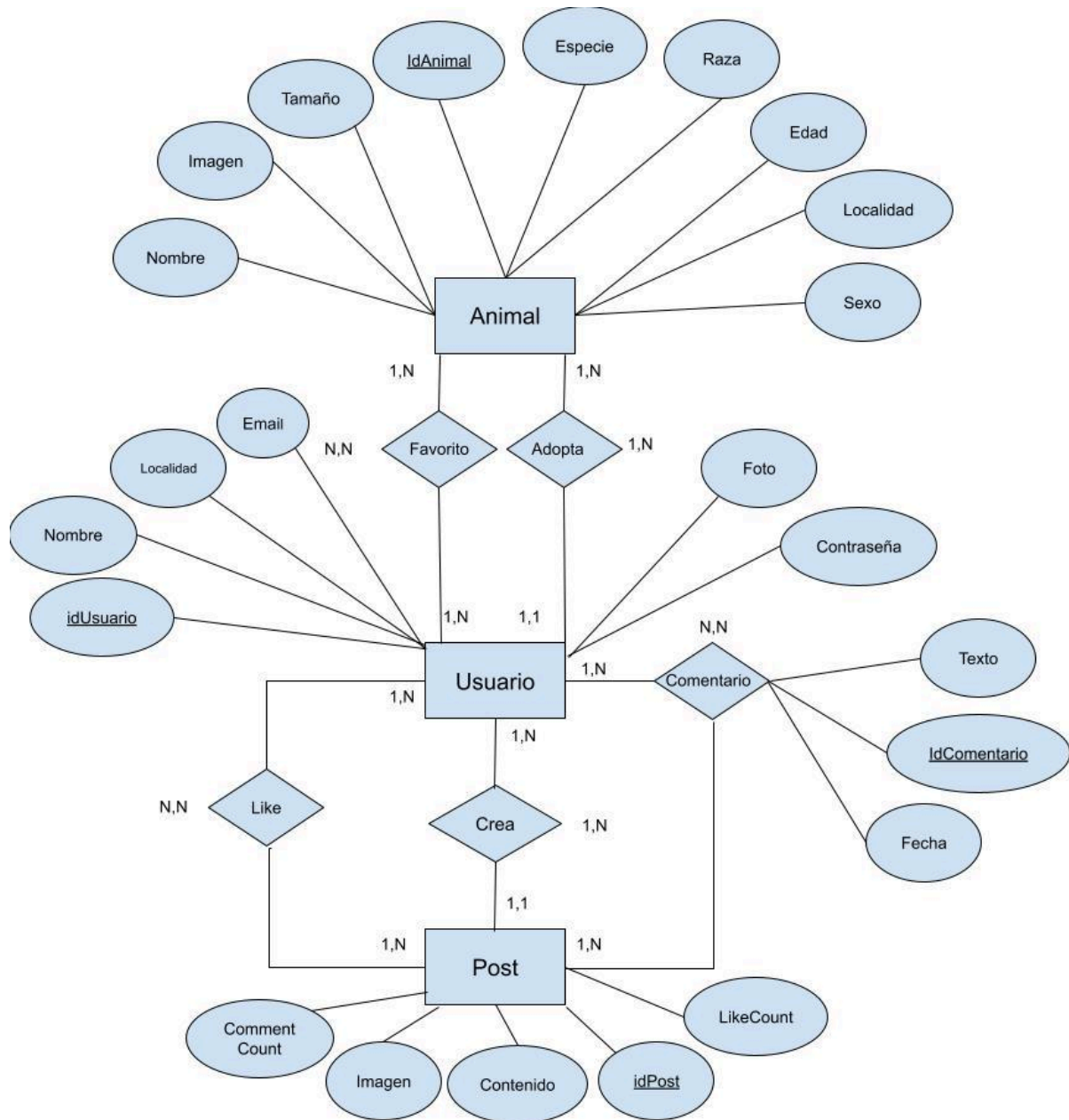
El diagrama de casos de uso sirve para mostrar de forma sencilla qué acciones pueden realizar los usuarios dentro de la aplicación. Cada caso de uso representa una funcionalidad importante del sistema, como registrarse, iniciar sesión, ver animales o publicar en el foro. Además, se indica qué tipo de usuario puede realizar cada acción, como un visitante, un usuario registrado o una protectora. Este diagrama ha sido muy útil durante la fase de análisis para tener claro qué debe hacer la app y qué necesidades debe cubrir desde el punto de vista del usuario.



## 8.3 Diagrama Entidad - Relación

Antes de crear la base de datos del proyecto AdoptaTuPet, fue necesario pensar bien qué información se iba a guardar y cómo se iba a organizar. Para ello, se ha realizado un diagrama entidad-relación (ER), que permite representar de forma visual las tablas principales, sus atributos y las relaciones entre ellas. Este paso es

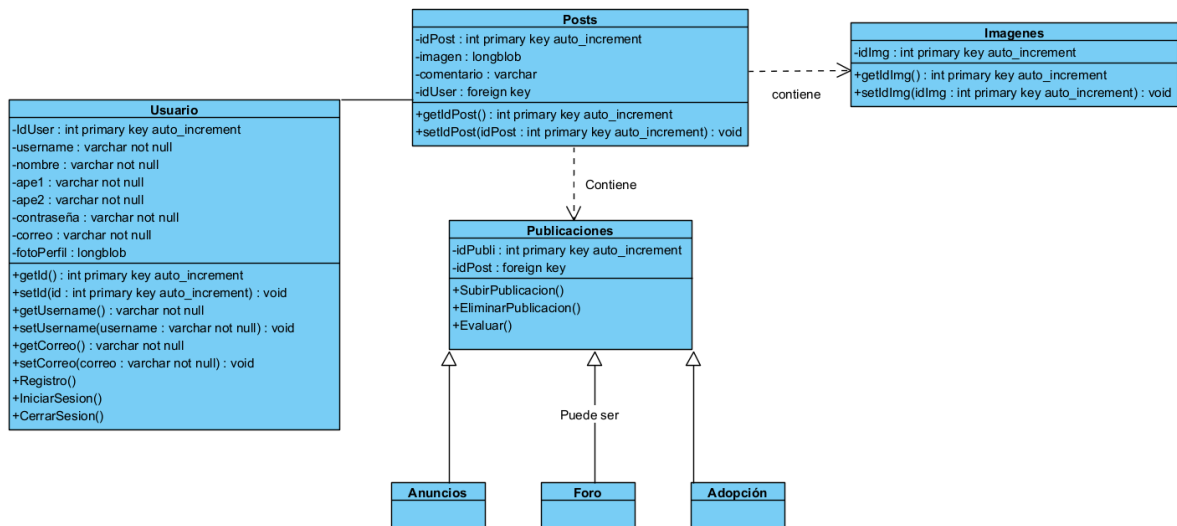
fundamental para asegurarse de que la estructura de la base de datos sea clara, funcional y fácil de mantener. Gracias a este diagrama, fue más sencillo diseñar las consultas y preparar el backend para gestionar correctamente todos los datos de la aplicación.



## 8.4 Diagrama de Clases

El diagrama de clases se ha utilizado para representar cómo se organizan y relacionan las clases dentro del código de la aplicación. Este tipo de diagrama es muy útil para planificar la estructura interna del programa, ya que muestra las clases principales, sus atributos, métodos y las relaciones que hay entre ellas (como

herencias o asociaciones). En el caso de AdoptaTuPet, este diagrama ha servido para tener una visión general del modelo de datos en Android y cómo se conectan con la lógica de la app, facilitando tanto el desarrollo como el mantenimiento del proyecto.



## 8.5 Prototipo

Antes de empezar a programar la aplicación AdoptaTuPet, se diseñó un prototipo visual que sirviera como guía para todo el desarrollo. Este prototipo muestra cómo serían las pantallas principales de la app, qué elementos tendría cada una (botones, menús, textos, etc.) y cómo se movería el usuario entre ellas. Se realizó con herramientas de diseño como Canva, y ayudó a tener una idea clara del aspecto y funcionamiento de la app antes de escribir código. Gracias a este paso, fue más fácil organizar el trabajo, detectar errores de diseño a tiempo y asegurar una buena experiencia para el usuario final.

### Pantalla de Inicio (Home)

#### Antes (Diseño inicial)

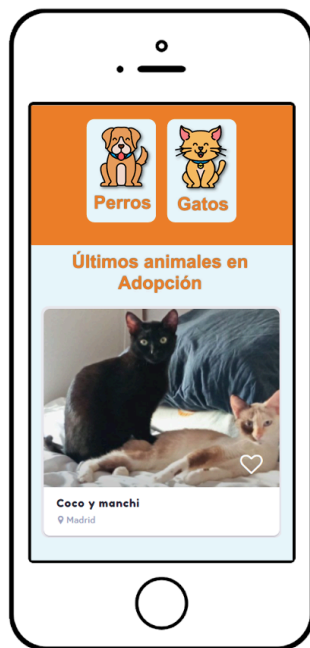
La aplicación tenía un diseño más conceptual, tipo maqueta:

- En la parte superior se mostraba un título grande (“AdoptaTuPet.es”) con fondo celeste, simulando una web.
- El centro de la pantalla estaba ocupado por una imagen estática de un gato y mensajes como “Adopta · Difunde · Involúcrate”.



- Abajo había un bloque naranja con un mensaje general (“Todos los animales necesitan un hogar”).
- La navegación inferior tenía iconos sencillos y texto (“Inicio”, “Adopta”, etc.), sobre fondo azul claro.

Este diseño funcionaba bien como prototipo para mostrar la idea general, pero no incluía contenido real ni interacción con los usuarios.



## Pantalla de Adoptar

### Diseño anterior de la sección "Adopta"

- Sección dividida por especie: un bloque naranja con botones ilustrados para “Perros” y “Gatos”.
- Título centrado: “Últimos animales en Adopción”.
- Tarjeta de animal simple: imagen rectangular, nombre del animal debajo, ubicación, y un icono de favorito (corazón) en la esquina.

La estructura mostraba animales pero de forma visualmente más compacta y con menos detalle textual.

## Pantalla de Contacto

El apartado de contacto se ubicaba en la parte inferior de la vista, acompañado de una tarjeta con una imagen de animales en adopción. Justo debajo:

- Logo del proyecto y nombre (“AdoptaTuPet.es”) en azul claro.
- Sección con iconos de redes sociales (Facebook, Twitter, Instagram).
- Dirección postal escrita en texto plano.
- Correo electrónico de contacto.



- Aviso legal y política de privacidad resumidos en una sola línea.

Toda la información se presentaba en bloque, con enfoque informativo estático.

## Filtro

### Antes:

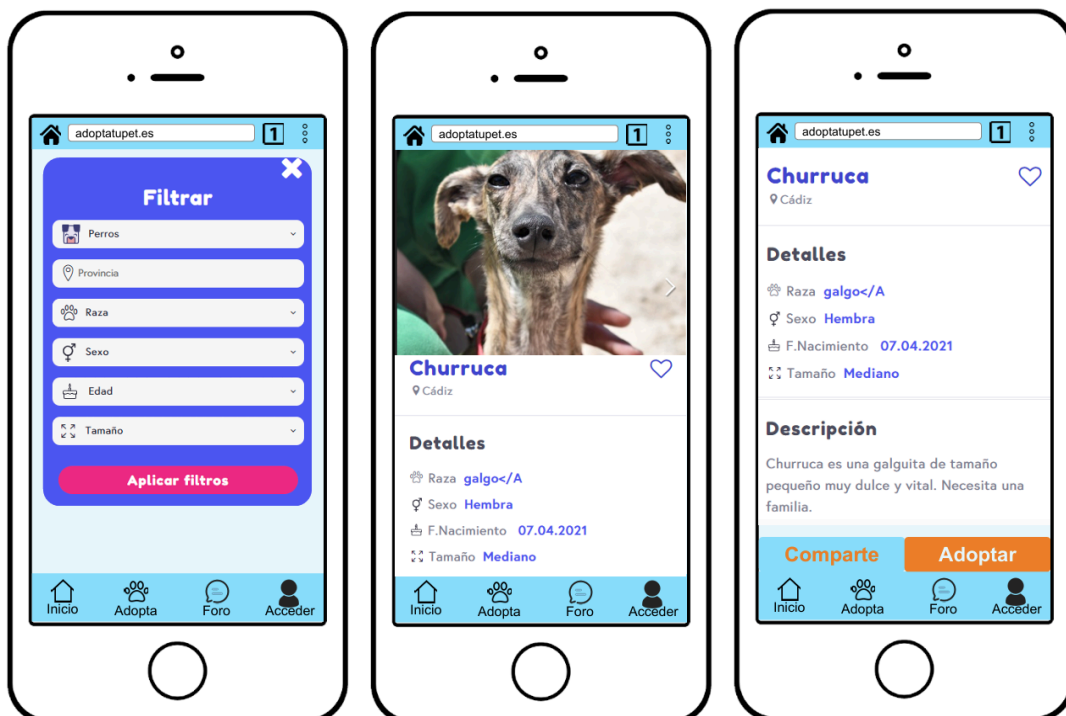
El filtro se mostraba como una pantalla completa con fondo azul y elementos grandes. Permitía seleccionar especie, provincia, raza, sexo, edad y tamaño, todos con iconos y menús desplegables. Al final había un botón llamativo para aplicar filtros.

## Perfil del animal

### Antes:

El perfil ocupaba toda la pantalla y mostraba:

- Foto grande en la parte superior.
- Nombre del animal y ubicación.
- Detalles básicos (raza, sexo, fecha de nacimiento, tamaño).
- Descripción en un bloque separado.
- Botones al final para compartir o adoptar.



Todo esto está hecho para que la experiencia sea cómoda y clara para cualquier persona, sin líos, y que lo más importante (el animal y su información) sea lo que más destaca en la pantalla.

## **Registro e Inicio de Sesión**

Desde el menú inferior de la app, en la pestaña "Acceder", se llega a una pantalla que permite al usuario identificarse, ya sea registrándose si es nuevo o iniciando sesión si ya tiene cuenta.

### **Registro**

La pantalla de registro y de inicio de sesión eran formularios contenidos dentro de una tarjeta centrada en el medio de la pantalla. Visualmente incluían:

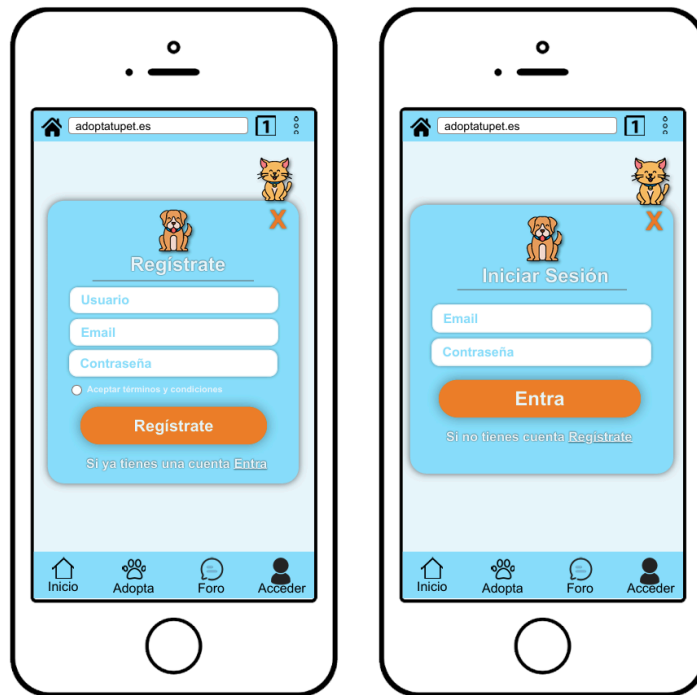
- Iconos de perro y gato como elementos decorativos.
- Campos de texto sencillos: nombre, email y contraseña.
- Casilla para aceptar términos y condiciones.
- Botón naranja con texto "Regístrate".
- Enlace alternativo debajo para cambiar entre registro y login.
- Parte inferior con barra de navegación visible.

El diseño se mantenía dentro de un espacio acotado, separado del resto de la interfaz.

### **Inicio de Sesión**

La pantalla de inicio de sesión es parecida, pero más simple: solo pide correo y contraseña. Cuando se pulsa el botón naranja "Entra", la app comprueba si esos datos están en la base de datos y, si son correctos, lleva al usuario al apartado principal.

Si el usuario aún no está registrado, puede hacerlo desde el enlace inferior que dice **"Si no tienes cuenta, Regístrate"**.



## Foro

El foro presentaba una estructura simple con publicaciones en forma de tarjetas:

- Nombre de usuario, avatar y mensaje en la parte superior.
- Imagen opcional justo debajo del texto.
- Reacciones visibles en forma de contador: comentarios, me gusta, compartir.
- Fondo celeste con bordes marcados entre publicaciones.
- Botón flotante lateral para alternar modo noche.

Las interacciones eran visuales, pero la estructura era más plana y comprimida.



## Tema oscuro

Un detalle nuevo que se ve en esta pantalla es el icono de la luna a la izquierda, que indica la posibilidad de activar un modo oscuro o nocturno. Esto puede ser útil para leer por la noche o en sitios con poca luz, aunque en esta pantalla se ve aún en diseño claro.

## Función

La idea principal del foro es conectar a la gente. Aquí no solo se habla de adopciones, sino de cualquier cosa relacionada con animales: desde emergencias, como un perro que se traga una almendra, hasta reflexiones más personales. Es un sitio donde los usuarios pueden desahogarse, pedir ayuda o dar consejos.

Este espacio del foro le da un toque más cercano y comunitario a la app, y está pensado para que todos puedan participar de forma sencilla y rápida, tanto con texto como con imágenes. También ayuda a mantener el interés por la app más allá de la adopción, ya que fomenta el día a día entre personas que quieren lo mejor para sus animales.

## 8.6 Tipografía y estilo

Para que la app se vea cercana y moderna, se ha utilizado la fuente Fredoka, una letra redondeada, clara y amigable que encaja bien con una app de adopciones.

Los colores principales definidos en la guía de estilo son:

- **colorPrimary: #ED7F05** → un naranja cálido que da fuerza y personalidad.
- **colorPrimaryVariant: #E7F6FC** → azul muy claro, usado como fondo.
- **colorSecondary: #87DDFF** → un azul más llamativo para botones y elementos destacados.

Además, se han usado esquinas redondeadas, márgenes suaves y botones con formas homogéneas para que toda la interfaz tenga coherencia y sea cómoda de usar.

## 8.7 Herramientas utilizadas en el proyecto

Durante el desarrollo de la aplicación se ha utilizado Android Studio como entorno principal de programación (Android Developers, s.f.). Para la gestión de usuarios y almacenamiento en la nube se integró Firebase (Firebase, s.f.), mientras que la comunicación entre la app y el servidor se realizó mediante Retrofit (Square, s.f.).



La app incluye un mapa con ubicaciones de protectoras, implementado con la API de Google Maps (Google Maps Platform, s.f.) y una interfaz coherente gracias a los componentes de Material Design (Material Design, s.f.).

En el backend se utilizó PHPMyAdmin para gestionar la base de datos MySQL (phpMyAdmin, s.f.), y la plataforma Render para alojar el servidor y hacerlo accesible desde la app (Render, s.f.).

Para controlar el historial del código se usaron Git (Git, s.f.) y GitHub como repositorio en la nube (GitHub, s.f.).

## Dominio

Render.com es una plataforma web que ofrece servicios para desplegar aplicaciones, APIs, bases de datos y sitios web estáticos en la nube. Permite a desarrolladores subir sus proyectos y hacer que estén disponibles públicamente sin necesidad de gestionar servidores manualmente.

### ¿Qué se puede hacer con Render?

Render permite trabajar con distintos tipos de proyectos:

- **Web Services:** para subir APIs o backends hechos en lenguajes como Node.js, Python, Ruby, Go, etc.
- **Static Sites:** para páginas web simples (HTML, CSS, JS) como portafolios, landing pages o documentación.
- **Background Workers:** para ejecutar tareas en segundo plano.
- **Cron Jobs:** para ejecutar scripts automáticamente en horarios definidos.
- **Bases de datos:** Render puede desplegar bases PostgreSQL directamente desde su plataforma.

## Dominios

- Render te ofrece un subdominio gratuito terminado en [.onrender.com](https://render.com), que puedes usar para acceder a tu app inmediatamente.
  - Mi dominio: [adoptatupet.onrender.com](https://adoptatupet.onrender.com)
- Permite usar HTTPS automáticamente (Let's Encrypt), sin configuraciones complicadas.

## Planes y precios

Render tiene tanto opciones gratuitas como de pago:

Tipo de servicio	Gratuito disponible	Pago a partir de
Static Sites	Sí	Gratis, con 100 GB/mes
Web Services	Sí (Free Tier)	Desde \$7/mes por instancia
Cron Jobs	Sí (limitado)	Desde \$1/mes por trabajo
Background Workers	No en plan gratis	Desde \$7/mes
PostgreSQL Database	No gratuito	Desde \$7/mes por base

- El plan gratuito de Web Services entra en "suspensión" si no hay uso activo, para ahorrar recursos.
- Puedes escalar los servicios si necesitas más potencia o disponibilidad continua.

## 9. DESARROLLO

### 9.1 Aspectos relevantes del desarrollo

El proyecto AdoptaTuPet se ha dividido en tres partes principales: la app que usa el usuario en el móvil, el servidor que gestiona las peticiones, y la base de datos que guarda toda la información.

En la app se diseñaron diferentes pantallas que permiten ver animales en adopción, publicar mensajes en el foro o iniciar sesión. Todo está organizado para que el usuario pueda moverse fácilmente entre las distintas secciones. El código de la app está dividido en archivos según lo que hacen: algunos se encargan de mostrar los datos en pantalla, otros de gestionar lo que hace el usuario (como pulsar un botón o rellenar un formulario), y otros se ocupan de enviar y recibir datos del servidor.

El servidor se encarga de recibir las peticiones de la app. Por ejemplo, cuando alguien quiere ver animales disponibles, el código del servidor busca esa información en la base de datos y la prepara para enviarla. Cada acción que puede hacer la app (como añadir un animal o iniciar sesión) tiene un archivo PHP específico. Esto ayuda a que el código esté ordenado y sea fácil de mantener.

La base de datos guarda la información en tablas, como si fueran hojas de cálculo. Cada fila tiene sus datos (por ejemplo, el nombre y la edad de un animal) y su identificador. El código que conecta con la base de datos está en archivos que permiten buscar, añadir o modificar estos datos cuando sea necesario.

## 9.2 Metodologías

Para que el desarrollo fuera más claro y organizado, cada parte del sistema se trabajó por separado. Por un lado, se preparó la parte que ve el usuario. Por otro, se creó el servidor que gestiona las peticiones. Y por último, se diseñó la base de datos. Aunque cada parte se desarrolló por separado, el código está hecho para que todo funcione junto, sin problemas.

En la app, el código se dividió en carpetas. Hay una para el diseño de las pantallas, otra para mostrar listas como los animales o los comentarios, y otra para enviar datos al servidor. Cada pantalla tiene su propio archivo de código y otro para el diseño visual. Esto hace que el trabajo esté bien separado y sea fácil hacer cambios solo donde hace falta.

En el servidor, el código también se organizó en partes. Hay archivos que reciben los datos que envía la app, archivos que validan esos datos y otros que se encargan de consultar o actualizar la base de datos. Así, si se quiere modificar cómo se guardan los mensajes del foro, por ejemplo, solo hay que cambiar una parte del código, sin tocar lo demás.

## 9.3 Técnicas

Cuando el usuario hace algo en la app, como ver animales o escribir un mensaje, el código de la app envía una petición al servidor. El código del servidor entiende lo que se está pidiendo, accede a la base de datos y responde con los datos que necesita la app. Todo este proceso ocurre en segundos y sin que el usuario lo note.

El código de la app está preparado para transformar automáticamente los datos que llegan del servidor en objetos que se pueden usar directamente. Por ejemplo, cuando llega la información de un animal, el código la convierte en un objeto "Animal" que se puede mostrar fácilmente en pantalla.

También hay partes del código que guardan información en el móvil, como el nombre del usuario o si ha iniciado sesión. Así, cuando se vuelve a abrir la app, no hace falta poner los datos otra vez.

En el servidor, el código está pensado para proteger los datos y evitar errores. Cada función del código tiene una tarea concreta, como registrar un usuario o devolver

una lista de animales. El código revisa que los datos estén bien antes de guardarlos o mostrarlos.

## 9.4 Tecnologías

Las tecnologías utilizadas en el desarrollo de la aplicación AdoptaTuPet se agrupan según su función en el sistema:

### Lenguajes de programación

- **Java**

Lenguaje de programación orientado a objetos utilizado principalmente en el desarrollo de aplicaciones Android. Permite organizar el código en clases y métodos, facilitando su mantenimiento y reutilización (Android Developers, s. f.).

- **Kotlin**

Lenguaje moderno desarrollado por JetBrains, totalmente compatible con Java y recomendado oficialmente por Google para aplicaciones Android. Su sintaxis es más simple y segura frente a errores comunes como los valores nulos (JetBrains, s. f.).

- **PHP**

Lenguaje interpretado que se ejecuta en servidores web. En este proyecto se ha usado para procesar las peticiones de la app y gestionar las operaciones con la base de datos (PHP, s. f.).

### Lenguajes de marcado y datos

- **XML (eXtensible Markup Language)**

Lenguaje utilizado en Android para definir la interfaz de usuario, estructurando elementos como botones, textos o imágenes mediante etiquetas jerárquicas (Android Developers, s. f.).

- **JSON (JavaScript Object Notation)**

Formato de datos liviano y fácil de leer. Se emplea para enviar y recibir información estructurada entre la app y el servidor, como fichas de animales o mensajes (JSON, s. f.).

## Tecnologías de mapas

- **Google Maps API**

Tecnología desarrollada por Google que permite mostrar mapas interactivos en aplicaciones. En este proyecto se utiliza para localizar visualmente las protectoras cercanas desde la app (Google Maps Platform, s. f.).

## 10. CONCLUSIONES

Al finalizar el proyecto, se comprueba que la aplicación ya reúne en un mismo lugar los perros y gatos de varias protectoras, ofrece filtros de búsqueda sencillos, muestra fichas completas con fotos, integra un foro y presenta un mapa de refugios; además, las protectoras gestionan sus animales desde la propia app. Todos estos objetivos formaban parte del plan inicial y se han cubierto dentro del calendario previsto, que iba desde el análisis a comienzos de mayo hasta la preparación para su entrega a finales de mes.

La integración de Google Maps y el sistema de registro avanzaron incluso antes de la fecha marcada, mientras que las pruebas con usuarios y el pulido del diseño necesitaron algunos días extra sobre lo indicado en el diagrama de Gantt. Quedaron pendientes la versión para iOS y un módulo de estadísticas internas sobre adopciones, ambos previstos ya como trabajos de una fase posterior.

A corto plazo se propone añadir notificaciones push para avisar de nuevos animales, un chat directo entre adoptante y protectora, la posibilidad de incluir otras especies y un modo oscuro completo. Estas mejoras pueden planificarse en hitos breves para que el proyecto siga creciendo sin complicar su mantenimiento.

Desde un punto de vista personal, el desarrollo ha permitido recorrer todas las etapas de una aplicación móvil real, desde la idea inicial hasta las pruebas. La base creada es estable y deja la aplicación lista para ampliarse cuando sea necesario, manteniendo el objetivo principal de facilitar que más animales encuentren un hogar.

# BIBLIOGRAFÍA

- Aires Machado, S. (23 de marzo de 2025). Modepran reclama un nuevo centro de acogida de animales en València ante la saturación actual. *Cadena SER*. <https://cadenaser.com/comunitat-valenciana/2025/03/23/modepran-reclama-un-nuevo-centro-de-acogida-de-animales-en-valencia-ante-la-saturacion-actual-radio-valencia/>
- Android Developers. (s. f.). *Android Studio*. Recuperado el 5 de junio de 2025 de <https://developer.android.com/studio>
- Firebase. (s. f.). *Firebase*. Recuperado el 5 de junio de 2025 de <https://firebase.google.com>
- Fundación para el Asesoramiento y Acción en Defensa de los Animales [FAADA]. (s.f.) <https://faada.org>
- Fundación Affinity. (s.f.). *Las cifras del abandono de perros y gatos aún lejos de descender*. <https://www.fundacion-affinity.org/en/library/las-cifras-del-abandono-de-perros-y-gatos-aun-lejos-de-descender>
- Git. (s. f.). *Git SCM. Distributed version control system*. Recuperado el 5 de junio de 2025 de <https://git-scm.com>
- GitHub. (s. f.). *GitHub: Let's build from here*. Recuperado el 5 de junio de 2025 de <https://github.com>
- Google Maps Platform. (s. f.). *Maps JavaScript API*. Recuperado el 5 de junio de 2025 de <https://developers.google.com/maps>
- IBM Developer. (28 de febrero de 2021). *Topología cliente/servidor*. Recuperado el 5 de junio de 2025 de <https://www.ibm.com/docs/es/iis/11.5.0?topic=topologies-clientserver-topology>
- JetBrains. (s. f.). *Kotlin Programming Language*. Recuperado el 5 de junio de 2025 de <https://kotlinlang.org>
- JSON. (s. f.). *Introducing JSON*. Recuperado el 5 de junio de 2025 de <https://www.json.org>
- López, I. (14 de junio de 2024). ¿Pensando en adoptar? Seis aplicaciones gratuitas para conocer perros y gatos que buscan un hogar. *20 minutos*.

<https://www.20minutos.es/noticia/5483738/0/pensando-adoptar-seis-aplicacion-es-gratuitas-para-conocer-perros-gatos-que-buscan-un-hogar/>

Material Design. (s. f.). *Material 3*. Recuperado el 5 de junio de 2025 de <https://m3.material.io>

Microsoft. (19 de marzo de 2024). *Información general sobre ASP.NET MVC*. <https://learn.microsoft.com/es-es/aspnet/mvc/overview/older-versions-1/overview/asp-net-mvc-overview>

Peters, G. (17 de enero de 2025). Alerta sobre presuntos fraudes en redes sociales en adopciones de animales. *Cadena SER*. <https://cadenaser.com/castillayleon/2025/01/17/la-asociacion-espanola-de-consuidores-alerta-de-un-presunto-fraude-en-redes-sociales-en-adopciones-de-animales-radio-palencia/>

PHP. (s. f.). *What is PHP?* Recuperado el 5 de junio de 2025 de <https://www.php.net>

phpMyAdmin. (s. f.). *phpMyAdmin*. Recuperado el 5 de junio de 2025 de <https://www.phpmyadmin.net>

Render. (s. f.). *Render - Cloud Hosting for Developers*. Recuperado el 5 de junio de 2025 de <https://render.com>

Sánchez, E. (20 de mayo de 2025). El 70 % de los perros y el 95 % de los gatos abandonados o perdidos no llevan microchip. *El País*. <https://elpais.com/clima-y-medio-ambiente/2025-05-20/el-70-de-los-perros-y-el-95-de-los-gatos-abandonados-o-perdidos-no-llevan-microchip.html>

Square. (s. f.). *Retrofit*. Recuperado el 5 de junio de 2025 de <https://square.github.io/retrofit/>

# ANEXO I: GLOSARIO

**AdoptaTuPet:** Aplicación móvil para Android que centraliza animales en adopción de diferentes protectoras, permitiendo buscarlos, ver su ficha, contactar con asociaciones y participar en un foro.

**Android Studio:** Entorno de desarrollo integrado (IDE) usado para crear aplicaciones Android.

**API REST:** Conjunto de reglas que permiten a distintas aplicaciones comunicarse a través de internet mediante peticiones HTTP.

**Backend:** Parte del sistema que gestiona la lógica del servidor, la base de datos y las operaciones internas (en este caso, desarrollado en PHP con base de datos MySQL).

**Base de datos (MySQL):** Sistema de almacenamiento estructurado de información, donde se guardan datos como usuarios, animales y publicaciones del foro.

**Controlador (Controller):** Parte del modelo MVC que gestiona las acciones del usuario y conecta los datos con la vista.

**Diagrama de clases:** Representación visual de las clases de programación, sus atributos y relaciones.

**Diagrama de casos de uso:** Diagrama que representa las funcionalidades principales de la app y quién puede realizarlas (usuarios, protectoras, etc.).

**Diagrama entidad-relación (ER):** Muestra cómo se relacionan las tablas de la base de datos entre sí (por ejemplo, animales con usuarios).

**FAADA:** Fundación para el Asesoramiento y Acción en Defensa de los Animales.

**Firebase:** Plataforma de Google para el desarrollo de apps que ofrece servicios como autenticación de usuarios y almacenamiento en la nube.

**Foro:** Espacio dentro de la app donde los usuarios pueden escribir mensajes, compartir experiencias o pedir ayuda relacionada con animales.

**Fragment:** Componente reutilizable en Android que representa una parte de la interfaz de usuario.

**Frontend:** Parte visual de la app que ve e interactúa el usuario final (en este caso, hecha en Java/Kotlin para Android).



**Google Maps API:** Servicio de mapas usado para mostrar la ubicación de refugios en la app.

**JSON:** Formato de intercambio de datos utilizado entre la app y el servidor.

**Material Design:** Sistema de diseño visual desarrollado por Google que se sigue en la interfaz de la app.

**Modelo - Vista - Controlador (MVC):** Patrón de arquitectura de software que separa la lógica del negocio, la interfaz y la gestión de eventos.

**Petfinder:** Plataforma internacional para buscar animales en adopción, usada como referencia en el análisis de alternativas.

**PHP:** Lenguaje de programación usado en el servidor para gestionar datos y responder a la app.

**phpMyAdmin:** Herramienta web para gestionar bases de datos MySQL.

**Prototipo:** Diseño visual previo al desarrollo de la app que permite planificar su funcionamiento y apariencia.

**Protectora:** Asociación que acoge temporalmente animales abandonados hasta que encuentran una familia.

**RecyclerView:** Componente de Android para mostrar listas de elementos desplazables, como los animales o los mensajes del foro.

**Retrofit:** Biblioteca de Android que facilita las llamadas HTTP a servicios web (como el backend en PHP).

**SharedPreferences:** Sistema de almacenamiento local en Android para guardar preferencias del usuario, como su sesión iniciada.

**Usuario registrado:** Persona que ha creado una cuenta en la app y puede acceder a funciones como el foro o la adopción.

**XML:** Lenguaje de marcado usado para definir la interfaz gráfica de Android.