



**LA APLICACIÓN PARA ADIESTRADORES**

. . . . .

**Gabriela barton cantín**

TUTOR: ALBERTO MAYOR

**Desarrollo de aplicaciones web**

CPIFP LOS ENLACES

**2023-2024**

KANDOG

Índice de contenido

[1. Descripción del proyecto 2](#_Toc164855804)

[1.1. Contexto del proyecto 2](#_Toc164855805)

[1.1.1. Ámbito y entorno 2](#_Toc164855806)

[1.1.2. Análisis de la realidad 3](#_Toc164855807)

[1.1.3. Solución y justificación 5](#_Toc164855808)

[1.1.4. Destinatarios 6](#_Toc164855809)

[1.2. Objetivo del proyecto 6](#_Toc164855810)

[1.3. Marco legal 8](#_Toc164855811)

[2. Acuerdo de proyecto 10](#_Toc164855812)

[2.1. Requisitos de la aplicación 10](#_Toc164855813)

[2.1.1. Requisitos funcionales 10](#_Toc164855814)

[2.1.2. Requisitos no funcionales 11](#_Toc164855815)

[2.2. Metodología a seguir para la realización del proyecto 12](#_Toc164855816)

[2.3. Tareas 12](#_Toc164855817)

[2.4. Planificación temporal de tareas 13](#_Toc164855818)

[2.5. Presupuesto 14](#_Toc164855819)

[2.6. Análisis de riesgos 16](#_Toc164855820)

[2.7. Pliego de condiciones 19](#_Toc164855821)

[3. Análisis y diseño 20](#_Toc164855822)

[3.1. Modelado de datos. Análisis y diseño de la base de datos 20](#_Toc164855823)

[3.1.1. Diagrama E/R 20](#_Toc164855824)

[3.1.2. Diagrama relacional 20](#_Toc164855825)

[3.2. Análisis y diseño del sistema funcional 23](#_Toc164855826)

[3.3. Análisis y diseño de la interfaz de usuario. Mockups. 24](#_Toc164855827)

[3.4. Diseño de la arquitectura de la aplicación 29](#_Toc164855828)

[3.4.1. Tecnologías/Herramientas usadas y descripción de las mismas 29](#_Toc164855829)

[3.4.2. Arquitectura de componentes de la aplicación 30](#_Toc164855830)

[4. Conclusiones 32](#_Toc164855831)

[5. Bibliografía y webgrafía 33](#_Toc164855832)

[6. Anexos 36](#_Toc164855833)

# Descripción del proyecto

“Kandog” es una aplicación web que proporciona recursos a los educadores caninos en Zaragoza con el objetivo de capacitar a la ciudadanía en el cuidado y educación de sus perros. Se pone una plataforma al servicio a nuestros clientes (adiestradores) para que gestionen sus propios clientes (propietarios).

En su *landing page* se ofrece una guía informativa y gratuita sobre cuidados, manejo básico y legislación vigente respecto a perros. Su primera versión contempla a los adiestradores como únicos usuarios, que a cambio de una suscripción podrán utilizar la plataforma para la gestión, diseño y almacenamiento de recursos en el desempeño de su profesión; y en lanzamientos posteriores, se incluirá a los propietarios de perros, que podrán crear su propio perfil para ponerse en contacto con los adiestradores, guardar informes, solicitar citas, etc.

## Contexto del proyecto

### Ámbito y entorno

Desde hace un par de décadas observamos un crecimiento exponencial en la tenencia de animales de compañía, llegando a su auge en la pandemia de 2020. Durante los tres años posteriores el número de mascotas en España aumentó un 38%, [según cifras de la Asociación Nacional de Fabricantes de Alimentos para Animales de Compañía](https://www.informacion.es/medio-ambiente/2023/01/09/espana-suma-15-millones-mascotas-68542763.html) (Benito, 2023). Zaragoza, siendo la quinta ciudad más grande del país, no se queda atrás:

“*La presencia de perros de compañía en las sociedades occidentales ha aumentado en los últimos años. En uno de cada tres hogares españoles ya hay, al menos, un perro. Sólo en Zaragoza hay cerca de 70.000 perros censados.”* –Ayuntamiento de Zaragoza, 2023.

En consecuencia, cada vez se ofrecen más facilidades para quienes conviven con perros. Para empezar, hay espacios donde se permite el esparcimiento canino reglado en prácticamente todos los distritos, así como áreas de suelta libre sin restricción de horario (los conocidos “pipicanes”, o parques específicamente caninos).

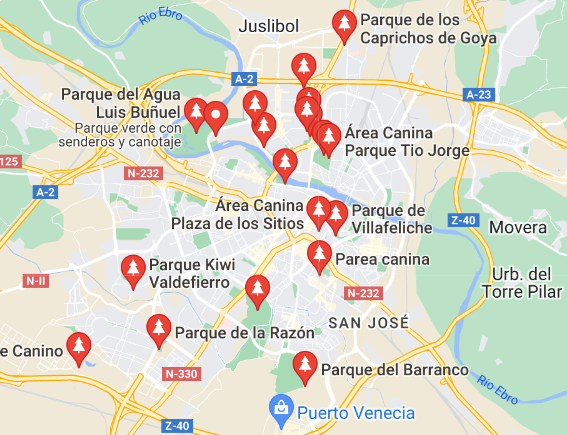


Figura 1. Búsqueda de "pipican Zaragoza" en Google Maps.

Respecto a otros espacios públicos, muchos bares y restaurantes en la ciudad permiten la entrada de perros, tanto en su interior como en sus terrazas. Se suman a la permisividad las grandes superficies comerciales: Puerto Venecia, Aragonia y más recientemente GranCasa son espacios donde los peludos son bienvenidos si cumplen con la normativa de conducta y manejo de cada centro. Desde febrero de 2023, también se permite el [acceso con tu perro a cualquier dependencia municipal](https://www.zaragoza.es/sede/servicio/noticia/318224) (*BOPZ decreto 265*), y si no se dispone de coche para moverse por la ciudad [está permitido utilizar el tranvía acompañado de tu mascota](https://www.tranviasdezaragoza.es/es/informacion/con-mascotas) siempre que sólo sea una y vaya asegurada en el último vagón, en su trasportín o con bozal (*Tranvía Zaragoza*, 2014).

En definitiva, estamos en un momento social en el que ver a un perro en el transporte público, asomando la cabeza bajo la mesa de un restaurante o entrando con su guía en una tienda ya no es especialmente llamativo en Zaragoza. Tendemos a olvidar, no obstante, que los derechos llevan aparejados unas responsabilidades. ¿Estamos preparados como sociedad para asumir todas estas concesiones y libertades con respecto a las mascotas?

### Análisis de la realidad

Que un lugar permita perros y que sea “*pet-friendly*” no son necesariamente sinónimos, del mismo modo que llevar a un niño pequeño a ver una película de terror puede estar permitido pero no es lo ideal. Incluso las familias con las mejores intenciones no siempre tienen todas las herramientas e información necesarias para tener un criterio confiable.

Para tomar decisiones respecto a nuestras mascotas sería necesario tener los conocimientos suficientes para tomar dichas decisiones de forma libre e informada. Mucha gente considera que tiene esos conocimientos, pero se sorprenden al encontrarse con que muchas cosas que consideraban ciertas en realidad eran bulos y falsedades. El problema es la multiplicidad de fuentes contradictorias y erróneas que hay al respecto de la educación canina que nos llevan a una comprensión incorrecta sobre cómo educar y entrenar a nuestros perros.

Hasta tal punto llega la desinformación que personajes como César Millán o Borja Caponni sacaron sus programas en la televisión española, “El Líder de la Manada” y “Malas pulgas”, respectivamente, en los cuales aplicaban unas técnicas de adiestramiento obsoletas y contraindicadas, como en el ejemplo de la *Ilustración 2*. Esto servía de ejemplo al más de millón de personas que vieron el programa (*FormulaTV, 2010*). Podríamos pensar que los tiempos de Millán y Caponni quedaron atrás y somos una sociedad más civilizada, pero [una encuesta de hace menos de 5 años](https://www.savethechildren.es/notasprensa/save-children-calcula-que-en-espana-mas-del-25-de-ninos-y-ninas-han-sido-victima-de) nos revela unos datos esclarecedores: más de un 80% de los padres consideraba que dar una bofetada a sus hijos para “educarles” no es una forma de maltrato, y que al menos un 25% de los niños y niñas españoles sufren algún tipo de maltrato a manos de sus cuidadores (*Save the Children, 2019*). Todos coincidimos en que “el maltrato está mal”, pero no acabamos de ponernos de acuerdo en *qué es maltrato*, a pesar de los esfuerzos de psicólogos, médicos, maestros y trabajadores sociales por definirlo. Si teniendo toda una flota de profesionales reglados con criterios unificados acerca del desarrollo infantil escribiendo guías, dando clases y orientando leyes aún nos cuesta enormes esfuerzos entender a nuestros hijos… la creencia de que entender y educar a un perro es *fácil* es, cuanto menos, poco realista.



Figura 2. Caponni, en “Malas Pulgas”, estrangula a un perro que intenta comer

La realidad es que tenemos una creciente población de perros mal educados que no tienen nociones sobre comportamiento social, en los mejores casos; y en los peores, que reaccionan catastróficamente ante un modelo de “educación” basado en castigos para suprimir conductas indeseadas. Esto desemboca en problemas de convivencia y/o riesgos de seguridad pública que pueden llevar al abandono del animal. Según [un estudio de la fundación Affinity](https://www.expertoanimal.com/causas-del-abandono-animal-y-como-evitarlo-25555.html), de los 300,000 animales que se abandonan anualmente en España se achaca un 13,2% a problemas de comportamiento (*AON – Seguros de Mascotas, 2024*).

La buena noticia es que, cada vez más, la sociedad española está cambiando su percepción materialista de los animales de compañía y comienza a percibirlos como seres sintientes con derechos. Podemos verlo reflejado en las corrientes legislativas, como en la nueva [ley de Bienestar Animal](https://www.boe.es/buscar/doc.php?id=BOE-A-2023-7936) (*BOE, 2023*) o [estudios recientes](https://www.fbbva.es/wp-content/uploads/2022/01/Presentacion-Estudio-de-Animales.pdf) donde se evidencia un incremento de consideración hacia nuestras mascotas (*BBVA, 2021*). La mayoría de las familias españolas hoy apuesta por ayudar a su animal cuando surgen problemas de conducta antes de abandonarlo.

Aquí es donde surge un nuevo problema: el mundo del adiestramiento y educación canina no está regulado en nuestro país. Eso significa que cualquier persona puede dedicarse a ello sin tener una debida capacitación profesional, lo que se traduce en falta de estándares de calidad, riesgos importantes para la seguridad y la salud de todos los involucrados, etc. Muchos perros empeoran sus conductas al ser expuestos a métodos basados en la supresión, metodología común a pesar de los numerosísimos estudios que demuestran su [falta de eficacia en comparación con el adiestramiento en positivo](https://www.frontiersin.org/articles/10.3389/fvets.2020.00508/full?fbclid=IwAR3tedKFmvPYmY) (*China, Mills & Cooper, 2020*) y el [impacto negativo en su bienestar](https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0225023&fbclid=IwAR3-QF576xA4aZFv8bd-dAjp9y30p8pLfzVXngQYIlWqRGrxA-mBz8RCHHs) (*De Castro et al., 2020*).

A pesar del intrusismo laboral, la falta de regulación, reconocimiento y recursos, en Zaragoza empiezan a surgir empresas de adiestramiento y profesionales que están debidamente formados (En Clave de Can, Tizas, Factor Amigo, Alé!…), lo que brinda esperanza al sector. Surge así mi interés por trabajar juntos hacia un objetivo común: promover el bienestar y la convivencia armoniosa entre la sociedad y sus perros.

### Solución y justificación

La propuesta es una aplicación web para que los adiestradores puedan gestionar agendas, servicios, recursos y mensajes de la clientela de forma unificada, y que en un futuro los clientes tengan acceso al listado de profesionales certificados suscritos a la aplicación.

Ideas similares ya han sido impulsadas en otros países, y tenemos algunas aplicaciones web y móviles como Dogo, Woofz o Pupy. Sin embargo se observan las siguientes limitaciones:

* PLATAFORMA: Están dirigidas a móviles. Las que tienen soporte web no ofrecen todas sus funcionalidades en un ordenador (en la *Ilustración 3* vemos que no hay inicio de sesión);
* LOCALIZACIÓN: Los adiestradores son mayoritariamente estadounidenses y ninguno de Aragón, lo que elimina posibilidad de sesiones presenciales y añade una barrera lingüística;
* MANEJO: Dogo es la plataforma más completa con acceso a una comunidad, planes de entrenamiento, artículos y novedades sin necesidad de pago, pero no se actualiza desde hace tiempo y además su app web es poco intuitiva. Woofz o Pupy son mucho sencillas en su uso pero también limitadas en sus utilidades gratuitas.
* DESTINATARIOS: El foco de estas aplicaciones está puesta en el cliente que necesita un adiestrador, pero a los adiestradores no se les ofrece ninguna ventaja por estar registrados (más allá del acceso a una base de clientes).

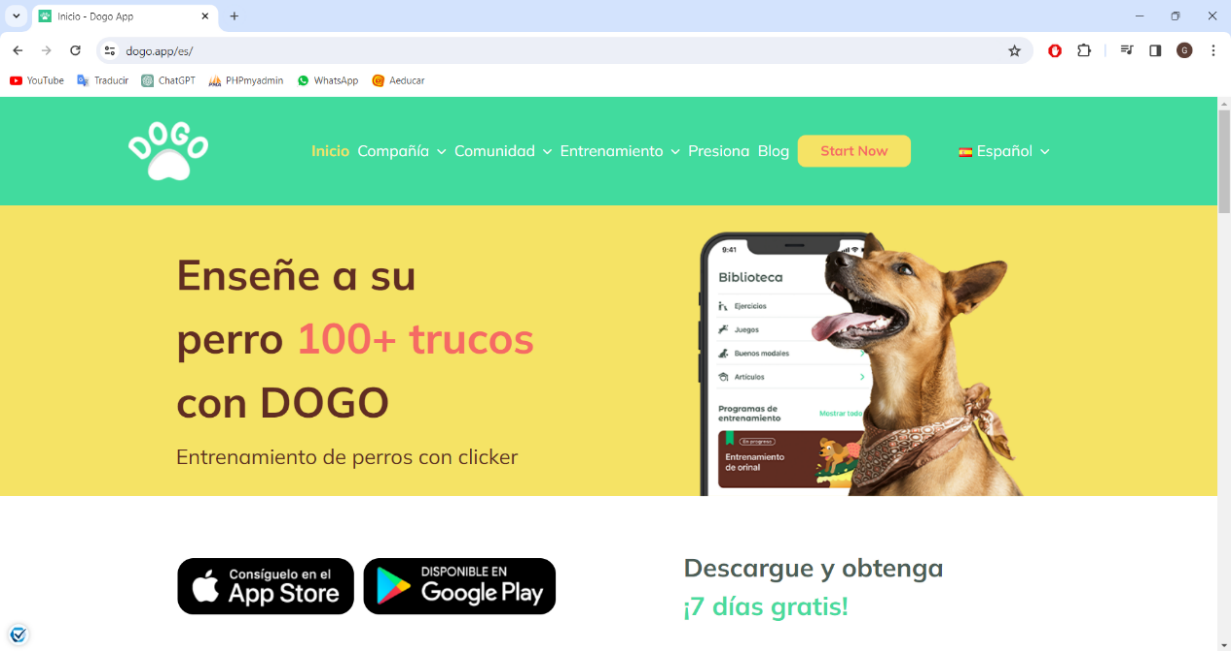


Figura 3. Aplicación web de "Dogo"

“Kandog” pretende dar respuesta a las carencias de aplicaciones análogas. El adiestrador tiene acceso a una base de datos donde almacena toda la información de los usuarios que solicitan su asistencia, así como herramientas para crear y compartir recursos, una agenda para organizar sus tareas y la opción de automatizar las reservas de sesiones de adiestramiento por ubicación y fecha. Se ofrecerá más adelante una información de calidad actualizada en español y la posibilidad de los propietarios de ponerse en contacto con adiestradores reales y certificados que residan prioritariamente en Zaragoza, Aragón y el resto de España.

### Destinatarios

Si bien el cliente objetivo es “el adiestrador” o “educador canino”, el tipo de perfil al que se dirige esta aplicación es aquella persona con una formación de base que esté habilitada para el manejo de perros y quiera prestar un servicio relacionado con ellos: cuidados, entrenamientos, rehabilitaciones… Además, se priorizará la presencia de usuarios locales a través de campañas a través de medios locales (radio, televisión, periódicos…).

Una vez que exista una base lo suficientemente extensa y sólida de trabajadores en “Kandog” se podrá dar luz verde a la introducción de quienes soliciten sus servicios, desarrollando funcionalidades destinadas para los tutores caninos y permitiendo su comunicación con el adiestrador en la misma plataforma.

## Objetivo del proyecto

Los objetivos principales que se recogen de lo expuesto anteriormente son:

• Unificar los criterios de bienestar animal y reunir adiestradores formados con objetivos y valores similares.

• Conceder una plataforma con herramientas útiles e intuitivas para los adiestradores que facilite por una parte la elaboración de informes, guías, estrategias y recursos online para sus clientes; y por otra la parte organizativa mediante agendas, recordatorios, etc.

• Permitir a propietarios de perros registrados inscribir a sus perros, contactar con adiestradores locales y llevar el seguimiento de su progreso desde su ordenador.

Otros objetivos serían reducir el intrusismo laboral en el mundo del adiestramiento y la educación canina, causante de un enorme perjuicio a los perros y las familias que necesitan de un profesional; prevenir problemas emocionales y de conducta en los perros, cuya experiencia en el primer año de vida encauza muchas de sus conductas y actitudes, y facilitar a las personas tanto la búsqueda como el contacto con trabajadores de confianza. Además, se pretende generar una comunidad de dueños y dueñas que puedan compartir sus experiencias, tejer una red de apoyo para todas esas personas que tienen dificultades relacionadas con sus canes.

La finalidad global de este proyecto es ayudar a los adiestradores a mejorar la calidad de vida de las familias, concienciar del compromiso real que requiere tener una mascota en todas sus posibles circunstancias y reducir el abandono animal a través de la educación real y efectiva de la ciudadanía.

Por último se encuentra mi objetivo y motivación personal de dar a conocer casos de éxito para ayudar a quienes se encuentran desesperanzados con sus perros. Hasta tres “profesionales” distintos dieron al mío por un caso perdido, y dos veterinarios se ofrecieron a sacrificarlo para “ahorrarme problemas”. Con el trabajo y los conocimientos necesarios para su rehabilitación, a día de hoy es un perro ejemplar que sirve de guía en mis sesiones de adiestramiento para otros perros con problemas emocionales y de conducta.

*The main objectives implicit in the aforementioned narrative are:*

*• To unify criteria for animal welfare and bringing educated trainers with similar goals and values together.*

*• To provide dog trainers with a platform that offers useful and intuitive tools to facilitate the creation of reports, guides, strategies, and online resources for their clients; and helps with the organizational aspects of the job, such as scheduling tasks and reminders.*

*• To allow registered dog owners to enroll their dogs, contact local trainers, and track their progress from their computers.*

*Other objectives would be to reduce labor intrusion in the dog-training field, which causes a huge harm to dogs and families who need a professional; to prevent emotional and behavioral problems in dogs, whose experiences in the first year of life shape many of their behaviors and attitudes, and to make it easier for owners to get in touch with trustworthy workers. In addition, there is an aim to generate a community of owners who can share their experiences, building a support network for all those experiencing difficulties related to their dogs.*

*The overall purpose of this project is to help trainers improve the quality of life for many families, to raise awareness of the real commitment required to have a pet in all their possible circumstances, and to reduce animal neglect or abandonment through real and effective public education.*

*Lastly, it is my personal goal and motivation to showcase success stories to help those who feel hopeless with their dogs. Up to three different "professionals" considered mine to be a lost cause, and two veterinarians offered to euthanize him to "save me the trouble." With the necessary work and knowledge for his rehabilitation, he is now an exemplary dog that serves as a guide in my training sessions for other dogs with emotional and behavioral problems.*

## Marco legal

La *web* tiene su propia regulación, tanto nacional como internacional, y tendremos que prestar atención a las leyes relacionadas con manejo de datos, publicidad, comercio electrónico, etc. sin olvidarnos de la normativa que exista relacionada con el manejo y bienestar animal.

Tomando como referencia [una publicación del Grupo Atico34](https://protecciondatos-lopd.com/empresas/requisitos-legales-pagina-web/), una consultoría nacional especializada en materia de protección de datos, las normas que debe cumplir cualquier página web para operar con legalidad en España serían las siguientes:

* [RGPD o Reglamento general de protección de datos](https://www.boe.es/buscar/doc.php?id=DOUE-L-2016-80807) (*DOUE*, 2016) y [LOPDGDD o Ley de Protección de Datos y Garantía de Derechos Digitales](https://www.boe.es/eli/es/lo/2018/12/05/3/con) (*BOE*, 2018): Para cumplir con todas las disposiciones de la RGPD y de la LOPDGDD se solicitará el consentimiento explícito de los usuarios antes de recopilar cualquier información personal y se garantizará la seguridad y confidencialidad de dicha información, incluyendo medidas específicas para proteger los derechos digitales de los usuarios. Esto puede traducirse en políticas de privacidad y de cookies, como veremos más adelante.
* [Ley General para la Defensa de los Consumidores y Usuarios](https://www.boe.es/eli/es/rdlg/2007/11/16/1/con) (BOE, 2007): Deben garantizarse los derechos de los consumidores y usuarios en todas las transacciones comerciales realizadas a través de la plataforma. Se proporcionará información clara y transparente sobre los productos y servicios ofrecidos, así como los derechos de desistimiento y devolución de los usuarios. Proveeremos atención al cliente accesible (incluyendo varias formas de contacto) y eficaz.
* [LSSICE o Ley de servicios de la sociedad de la información y de comercio electrónico](https://www.boe.es/eli/es/l/2002/07/11/34/con) (BOE, 2002): Esta ley regula las transacciones comerciales realizadas a través de una plataforma web. Se debe proporcionar información clara y accesible sobre la empresa (denominación social, dirección postal y electrónica, datos de registro mercantil…). Asimismo, en caso de hacer pagos en la plataforma, tendremos que asegurar que es un proceso seguro y generar recibos/justificantes de pago para el comprador, por ejemplo mediante un mensaje de correo electrónico.
* [Ley de Ordenación del Comercio Minorista](https://www.boe.es/buscar/act.php?id=BOE-A-1996-1072) (*BOE*, 1996) y [Ley de Competencia Desleal](https://www.boe.es/eli/es/l/1991/01/10/3/con) (*BOE*, 1991): Aunque se redunda en aspectos como la protección de los derechos de los consumidores y la obligación de informar de forma clara sobre precios y términos de servicios, se añaden aspectos como: evitar prácticas comerciales desleales que puedan perjudicar a otros negocios (en nuestro caso, en la industria de la educación canina), garantizar que la publicidad sea honesta y no engañosa, evitar realizar comparaciones falsas con otros servicios similares en el mercado y respetar los derechos de propiedad intelectual de terceros.
* [Ley de Propiedad Intelectual](https://www.boe.es/eli/es/rdlg/1996/04/12/1/con) (*BOE*, 1996) y [Ley de Marcas](https://www.boe.es/eli/es/l/2001/12/07/17/con) (*BOE*, 2001): Deben respetarse los derechos de propiedad intelectual de terceros, incluyendo los derechos de autor y marcas registradas. Si fuera necesario, se obtendrán los permisos necesarios para utilizar cualquier contenido protegido por derechos de autor. Se tomarán las precauciones necesarias a la hora de diseñar el logotipo, eslogan, etc. siguiendo los procedimientos establecidos por la Oficina Española de Patentes y Marcas.

En Aragón sólo se establecen [medidas para la implantación tecnológica relativa a los servicios públicos (*BOA*, 2023)](https://www.boa.aragon.es/cgi-bin/EBOA/BRSCGI?CMD=VEROBJ&MLKOB=1265652460505). Sin embargo, podemos tomar nota de sus requisitos, entre ellos la seguridad, la escabilidad, la perdurabilidad de la información, la portabilidad o la sostenibilidad.

Partiendo de las leyes estatales, cualquier página web debe tener una serie de textos o avisos necesarios, que son: el aviso legal, la política de privacidad y la política de cookies. Como además “Kandog” va a ofrecer un servicio de pago, habría que añadir también las condiciones de contratación.

Veamos cómo se desarrolla cada uno de ellos: (ANEXOS)

* Aviso legal: Por una parte, deben aparecer los datos registrales, como son el nombre de la empresa, su NIF, y los datos de contacto (domicilio social, teléfono, correo electrónico); y por otra parte los términos de uso de la web, en los que se incluyen condiciones de uso y contratación, cláusulas de propiedad intelectual y responsabilidad del titular en caso de caída de los servicios o mal funcionamiento.
* Política de privacidad: Implica informar sobre el registro de datos y aspectos relacionados con su tratamiento: cómo se recogen, quién es el responsable, quién es el encargado de su supervisión, qué finalidad tiene su recogida, si se van a enviar a terceros y/o fuera del país, durante cuánto tiempo se conservan esos datos y cómo rescindir del permiso otorgado por el usuario.
* Política de cookies: Debemos informar de que nuestra web utiliza cookies. En este texto tendremos que explicar al usuario qué son las cookies, dar información sobre las que utilizamos (nombre, titular, duración, tipo de cookie y finalidad) y especificar que requieren del consentimiento expreso y voluntario del usuario, que podrá rechazarlas si así lo desea.
* Condiciones de contratación: De cara a los pagos que se hagan en la plataforma habrá que constituir este documento especificando la identidad del vendedor, los precios reales de lo que se vende, procedimientos de pago, el derecho a desistimiento, medios para corregir errores y el proceso de devolución en caso de ser necesario.

Dentro de la aplicación no olvidaremos citar y poner a disposición del usuario las leyes y los puntos más importantes de las mismas sobre tenencia, bienestar y protección animal en Zaragoza, Aragón y el resto del territorio nacional, aunque específicamente relativo al ejercicio del adiestramiento no hay ninguna regulación estudiada.

# Acuerdo de proyecto

Este Acuerdo del Proyecto tiene como objetivo establecer los términos y condiciones para producir la aplicación “Kandog”. Normalmente se elabora entre una empresa o cliente y el equipo de desarrollo con el fin de definir claramente las responsabilidades, los requisitos y las expectativas de ambas partes durante el desarrollo y la implementación del proyecto. El proyecto no cuenta con un cliente real, así que haremos la aproximación más realista posible a las exigencias y requerimientos de los destinatarios *target* a los que se dirige el proyecto.

Para lograr los objetivos que establecíamos en el apartado anterior y darle forma a la plataforma donde los adiestradores puedan gestionar a sus clientes, es fundamental contar con un marco de trabajo que guíe todas las actividades y decisiones relacionadas con el proyecto.

## Requisitos de la aplicación

Una de las consultoras referentes de IT en España nos da [7 razones por las que establecer requisitos](https://www.northware.mx/blog/requerimientos-en-el-desarrollo-de-software-y-aplicaciones/) en el desarrollo de software y aplicaciones es fundamental (*Northware*, 2022). Entre ellas, con los requisitos que establezcamos a continuación podremos definir el alcance del proyecto, identificar potenciales riesgos, orientar los esfuerzos de forma eficaz y evitar sorpresas de última hora.

Los requisitos en desarrollo web se suelen clasificar como “requisitos funcionales”, que describen acciones que son necesarias para el funcionamiento del sistema, y “requisitos no funcionales”, que se orientan más a características que el software debería tener pero es prescindible para su uso.

### Requisitos funcionales

|  |  |  |
| --- | --- | --- |
| ID | Título | Descripción |
| *RF1* | Entrada de usuarios | Debe ser posible iniciar sesión para acceder a la plataforma. Para los usuarios no registrados se dispone de un botón que les redirija a una página de introducción de datos. |
| *RF2* | Salida de usuarios | Debe haber un botón visible en todo momento para salir de la sesión. |
| *RF3* | Gestión de cuenta | Los datos personales deben ser accesibles y modificables por parte del usuario a quien identifican. También tiene que existir la opción de borrar la cuenta de forma permanente. |
| *RF4* | Página de aterrizaje | Debe haber una página de aterrizaje que ofrezca unos contenidos gratuitos y otros que promuevan la aplicación de “Kandog” tanto a adiestradores como a propietarios de perros. |
| *RF5* | Página principal | Los dos tipos de usuarios (adiestradores y propietarios) deben tener una página principal o *homepage* similar, con características distintas según el tipo de usuario, pero que aúnen los módulos de: Mensajes, calendario, recursos, diario de trabajo, historial de pagos y ayuda (atención al cliente). |
| *RF6* | Mensajería | Los usuarios deben poder enviar mensajes directos a otros usuarios registrados, así como tener la posibilidad de ver y responder a los mensajes recibidos. |
|  | Búsqueda y filtros | Debe ser posible agilizar la búsqueda en cualquiera de las secciones o módulos del usuario a través de un *input* de texto o filtros específicos. Los filtros pueden personalizarse? |
| *RF7* | Notificaciones y recordatorios | Es necesario que el usuario sepa si le ha llegado un mensaje o tiene alguna notificación acerca de la disponibilidad de adiestradores o reservas. |
| *RF8* | Documentos y Recursos | Tiene que ser posible guardar documentos y otros recursos audiovisuales en una biblioteca personal donde poder acceder a ellos o recuperarlos en cualquier momento. También debe ser posible compartir y descargarlos. |
| *RF9* | Reservas online | El cliente debe poder visualizar la agenda disponible del adiestrador y hacer una reserva que el propio adiestrador tendrá la opción de confirmar o denegar. La reserva implica el desembolso que establezca el adiestrador. |
| *RF10* | Visualización de contenidos | Un usuario debe ser capaz de navegar por todo el sitio a través de la interfaz y acceder a los contenidos públicos en cualquier momento. |
| *RF11* | Historial | Debe haber un registro de la actividad de los usuarios como compras, reservas efectuadas, etc. Al que puedan acceder en cualquier momento. |
| *RF12* | Textos legales y ayuda | Tanto en la página de aterrizaje como en las páginas principales de usuario debe presentarse toda la documentación legal en forma de *link* o *pop-up*, así como una forma de contacto con el gestor de la aplicación para solucionar problemas o resolver dudas. |

Tabla 1. Requisitos funcionales de Kandog

### Requisitos no funcionales

|  |  |  |
| --- | --- | --- |
| ID | Título | Descripción |
| *RNF1* | Rendimiento | La aplicación web debe ser rápida y eficiente, con tiempos de carga mínimos y una navegación fluida para una mejor experiencia de usuario. |
| *RNF2* | Seguridad | La plataforma debe garantizar la seguridad y confidencialidad de los datos de los usuarios, implementando medidas de protección como el cifrado de datos y el control de acceso. |
| *RNF3* | Calidad | El contenido de la guía informativa debe ser veraz y útil para los usuarios que accedan a la misma. descargable? |
| *RNF4* | Interfaz | La navegación por la plataforma debe ser sencilla e intuitiva. |
| *RNF5* | Estética | El diseño web seguirá una guía de estilo unificada y característica de la marca que haga más agradable la experiencia de usuario. |
| *RNF6* | Documentación | La página debe disponer de la documentación legal necesaria que estará visible/accesible en todo momento. |
| *RNF7* | Idiomas | Para usuarios de habla extranjera debe existir la opción de traducir los contenidos al inglés. |
| *RNF8* | Mantenimiento | El código debe seguir unos criterios de claridad y modularidad para que la aplicación sea fácil de mantener y actualizar. |
| *RNF9* | Compatibilidad | La plataforma debe ser compatible con diferentes dispositivos y navegadores web. |
| *RNF10* | Accesibilidad | La página debe cumplir con los estándares de accesibilidad para garantizar que todos los usuarios puedan utilizarla sin problemas. |
| *RNF11* | Escalabilidad | La aplicación debe ser capaz de manejar un crecimiento en el número de usuarios/datos sin comprometer su rendimiento o funcionalidad. |

Tabla 2. Requisitos no funcionales de Kandog

## Metodología a seguir para la realización del proyecto

“Kandog” está producido mediante una metodología ágil. Basada en la colaboración y el trabajo en equipo, ésta nos ofrece “un enfoque flexible y adaptable que nos permite responder de manera efectiva a los cambios en los requisitos y prioridades del proyecto” (*EKCIT*, 2022). Consiste en un enfoque iterativo incremental: Se divide el proyecto en conjuntos de tareas (definidas más adelante en [*2.4: Planificación temporal de tareas*](#_Planificación_temporal_de)) que concluyen en una nueva capa de funcionalidad en el producto. Cuando se termina cada etapa, también llamada *sprint*, se presenta el producto al cliente, o en este caso al tutor, para recibir una retroalimentación que permite ajustar y mejorar el producto antes de su entrega final. Este enfoque brinda la flexibilidad necesaria para adaptarse a las circunstancias, permite una mejora continua del software y evita desvíos de los objetivos.

Atlassian, una de las empresas de software más populares (autora de *Jira,* entre otros), fue pionera en el desarrollo mediante metodologías ágiles, diferenciando dos grandes ramas principales: *Scrum* y *Kanban*. Una de sus redactoras (Dumond, s.f.) nos explica en un artículo la diferencia entre ellas: “Mientras que el *Scrum* se basa en las iteraciones de proyectos de duración fija, el *Kanban* se centra en un sistema de publicación continuo.” Puesto que tenemos una planificación, una demostración del *sprint* en las fechas previamente señalizadas y una limitación de tiempo, podemos decir que “Kandog” está más cerca de un modelo *Scrum* dentro de las metodologías ágiles.

Para llevar el proyecto a cabo se ha hecho una planificación de tareas y una temporalización de dos *sprints* principales que deben cumplir una serie de requisitos. Dentro de cada sprint es necesario seguir unos pasos comprendidos en establecer los requisitos (funcionales y no funcionales), hacer un análisis detallado de los mismos para diseñar la arquitectura del sistema, desarrollar el código necesario y finalmente hacer pruebas para comprobar que el software cumple con los requisitos iniciales. Más adelante, superadas las pruebas y cumplidos los requisitos, podrá considerarse su despliegue en un entorno de producción.

## Tareas

Estas son las tareas y la temporalización predefinidas sobre las que elaboraremos las demás:

|  |  |  |
| --- | --- | --- |
| Nombre de la tarea: | | Fecha límite (2024) |
| *TAREA I:* | *Propuesta proyecto* | 5 de febrero |
| *TAREA II:* | *Documentos de Descripción y Acuerdo de Proyecto* | 8 de abril |
| *TAREA III:* | *Documento Análisis y Diseño del primer sprint* | 22 de abril |
| *TAREA IV:* | *Primer Sprint y Pruebas Funcionales* | 22 de abril |
| *TAREA V:* | *Documento Análisis y Diseño del segundo sprint* | 6 de mayo |
| *TAREA VI:* | *Segundo Sprint y Pruebas Funcionales* | 6 de mayo |
| *TAREA VII:* | *Documentación completa* | 20 de mayo |
| *TAREA VIII:* | *Depósito del documento de memoria del proyecto* | 27 de mayo |

Tabla 3. Tareas principales del proyecto.

## Planificación temporal de tareas

Se ha utilizado *Jira* *Scrum* para la planificación temporal de este proyecto, ya que facilita la planificación de *sprints* con un alcance y unos objetivos definidos. Para proyectos de equipo organizados en tareas con límite temporal es conveniente utilizar diagramas de Gantt, puesto que ofrecen una representación visual clara y completa del cronograma y las actividades involucradas. Estos diagramas son una de las varias funcionalidades que ofrece *Jira*.

En el desarrollo de “Kandog” hay ocho grandes hitos o tareas a cumplir, que a su vez se subdividen en pequeñas tareas que pueden ser revisadas, actualizadas, eliminadas, intercambiadas o añadidas a medida que progrese el trabajo. Podemos ver un resumen de la cronología mensual en la *Ilustración 4*, con las tareas y sus subtareas, su estado (por hacer/finalizada) y sus momentos de inicio y final.

Gráfico

Descripción generada automáticamente

Figura 4. Diagrama de Gantt en Jira, cronología mensual de tareas de Kandog

## Presupuesto

Un presupuesto es “la suma de los costes estimados de todas las actividades del proyecto que se requieren para su compleción. En términos simples, es el total de dinero que necesita para completar el proyecto. (…) No basta con hacer una estimación y formalizarla. Después de la planificación viene el seguimiento, ajustes y actualización y análisis posterior.”  (*Stsepanets*, 2023).

“Kandog” es una aplicación generada por un equipo unipersonal sin más gastos que las horas de desarrollo e investigación y la amortización del equipo de trabajo preexistente. Es por ello que algunos gastos se calculan de forma orientativa haciendo un ejercicio hipotético de lanzamiento al mercado.

Para poder estimar de forma más precisa el presupuesto de “Kandog” se hace un desglose de los tipos de gasto y su aproximación económica (*ver tabla 4*). Se clasifican en costes fijos, que no varían en función de la producción o resultados; y los costes variables, que pueden cambiar en función de la actividad, por lo que requieren un mayor monitoreo por ser más impredecibles. Además, hace falta considerar si los costes fijos son pagos mensuales o anuales.

Por una parte, consideramos qué gastos reales vamos a tener durante los dos meses y medio que se extiende la temporalización de este proyecto: Si estimamos unas 100 horas de trabajo total, entre las cuales se incluirían las horas de investigación, desarrollo y pruebas, y planteamos un coste de 10€ por hora de tiempo invertido, entonces ya partimos de al menos 1000€ de coste de producción. A ello hay que sumarle los materiales, como el portátil de trabajo valorado en 600€ y los bolígrafos, folios, calculadora, etc. que podemos redondear en su conjunto a unos 10€ al mes.

Como el portátil ya estaba comprado, la fórmula para calcular su amortización mensual es:

Si calculo que le quedan 2 años más de actividad, teniendo en cuenta que costó unos 600€, el cálculo resulta en una amortización mensual de 25€.

Tomo como área de trabajo el alquiler de mi habitación, unos 300€ mensuales, junto con los suministros de agua, luz e internet. No se hacen desplazamientos, no voy a usar servidores físicos ni de pago y mi entorno de desarrollo, *Visual Code*, es gratuito incluso para usos comerciales, según su licencia de uso (*Microsoft*, 2022). Contando con todos los gastos enumerados hasta ahora tenemos aproximadamente 1400€ de coste asegurado el primer mes.

Si el proyecto fuese a lanzarse para generar beneficios, habría que darse de alta en autónomos y pagar una cuota, el IRPF sobre los beneficios y el IVA sobre las ventas. Del mismo modo, para promocionar la aplicación, habría que hacer un desembolso importante en marketing y publicidad para llegar al público objetivo, contratar un servicio de hosting web y tener en cuenta posibles gastos derivados de imprevistos. De esta forma se duplica el costo del primer mes.

Hay que tener en cuenta que en el inicio de una empresa lo habitual es tener pérdidas, por lo que es necesario partir de un presupuesto o pedir algún tipo de préstamo para hacer frente a los gastos e inversiones necesarias. Podemos aplicar algunas fórmulas para hacernos una idea de cuánto tendríamos que producir o a qué precio para alcanzar cierta rentabilidad.

|  |  |  |  |
| --- | --- | --- | --- |
| Gasto | Tipo de gasto | Descripción | Coste estimado |
| Recursos humanos: |  |  |  |
| Salario desarrollo | *Variable* | 10€/hora, ~80 horas | 800€ |
| Investigación: |  |  |  |
| Análisis contextual | *Variable* | Coste por horas invertidas de investigación y aprendizaje, 10€/hora | 100€ /vez |
| Estudio de mercado | *Variable* | 30€ /vez |
| Tecnologías | *Variable* | 70€ /vez |
| Recursos materiales: |  |  |  |
| Equipo de trabajo | *Fijo* | Amortización de portátil | 25€ /mes |
| Suministros oficina | *Fijo* | Papelería, escritorio… | 10€ /mes |
| Servidores | *Fijo* | --- | --- |
| Servicios externos: |  |  |  |
| Asesoramiento legal | *Fijo* | Asesor externalizado | 100€ /mes |
| Marketing | *Variable* | Pago por campaña | 1000€ /vez) |
| Licencias: |  |  |  |
| Entorno de desarrollo | *Fijo* | Gratuito (VS Code) | --- |
| Dominio | *Fijo* | Servidor virtual privado | (20€ /mes) |
| Seguridad web | *Fijo* |
| Gastos operativos: |  |  |  |
| Alquiler de oficina | *Fijo* | Coste del área de trabajo | 300€ /mes |
| Gastos luz y agua | *Variable* | Aproximación mensual para ambos | 50€ /mes |
| Seguro | *Fijo* | Responsabilidad Civil | 50€ /mes |
| Internet | *Fijo* | Plan fijo mensual | 50€ /mes |
| Desplazamientos | *Fijo* | --- | --- |
| Imprevistos (Reserva): |  |  |  |
| Otras contingencias | *Fijo* | Reserva mensual | 100€ /mes |
| Otras actividades: |  |  |  |
| Permisos municipales | *Variable* | Uso de espacios públicos | --- |
| Producción audiovisual | *Variable* | Equipo y edición propios | --- |
| Alquiler de espacios | *Variable* | Terrenos, naves… | --- |
| Impuestos: |  |  |  |
| Cuota autónomos | *“Fijo”* | Tarifa plana primer año | 80€ /mes |
| IRPF / IVA | *Variable* | 21% de beneficios / ventas | --- |
| TOTAL ESTIMADO: | | El primer mes: 2.785€ | |

Tabla 4: Presupuesto inicial de Kandog

El punto de equilibrio es el nivel de ventas necesario para igualar ganancias y pérdidas, de forma que a partir de ese punto se generarían beneficios. Su fórmula es:

Si pusiéramos de forma orientativa 12€ mensuales de suscripción a los adiestradores, necesitaríamos al menos unas 200 mensualidades, o 17 suscripciones anuales, para poder cubrir los gastos del primer mes.

## Análisis de riesgos

En cualquier proyecto con una mínima inversión es buena práctica dedicar un porcentaje de los recursos al estudio y la prevención de riesgos. Si se conocen los posibles riesgos, es posible evitarlos en la medida de lo posible y tener preparadas estrategias para corregirlos. De este modo se reduce el impacto de los imprevistos sobre el desarrollo de la aplicación y se trabaja con más fluidez y confianza.

Según la ingeniera electrónica y directora de proyectos Olga Patricia Torres (2022) algunas [formas de identificación y posterior análisis de los riesgos](https://www.piranirisk.com/es/blog/gestion-de-riesgos-proyectos-de-software) son la observación directa, el *brainstorming* o lluvia de ideas, el juicio de expertos, taxonomía de riesgos, diagrama de Ishikawa o entrevistas/grupos focales. En este caso, tendremos que fiarnos de la opinión de expertos y profesionales en el campo, así como estadísticas sobre los riesgos más comunes en el desarrollo web. Un artículo del equipo de desarrollo web StartTechUp nos enumera los [10 riesgos más comunes en el desarrollo de software](https://www.startechup.com/es/blog/10-common-software-development-risks/) (*Jacinto*, 2023). Con ellos elaboro, junto con algún riesgo adicional, una tabla que evalúa su probabilidad de suceso durante el desarrollo de “Kandog” y las medidas de prevención y corrección de dichos riesgos:

|  |  |  |
| --- | --- | --- |
| Riesgo: | Probabilidad: | Soluciones: |
| *Estimación incorrecta del tiempo / Falta de tiempo* | Alta | * Priorizar las tareas y características críticas para garantizar que se completen a tiempo. * Consultar al tutor o a expertos en la materia para una mejor gestión del tiempo. * Utilizar técnicas de gestión de proyectos como PERT, o de tiempo como la “técnica Pomodoro”. |
| *Malas prácticas* | Media | * Invertir más tiempo en la planificación. * Buscar y cumplir los estándares de desarrollo de código. |
| *Superar un presupuesto* | --- |  |
| *Riesgos externos inesperados* | Baja | * Identificar los riesgos externos antes de que ocurran. * Establecer planes de actuación para dichos riesgos. * Mantener una comunicación abierta con el tutor para conocer cualquier cambio en la situación de entrega. |
| *Código de baja calidad* | Media | * Prestar atención a la modularización del código. * Utilizar herramientas para identificar áreas problemáticas. * Invertir tiempo en mejorar las habilidades de codificación. |
| *Alcance impreciso del proyecto* | Alta | * Definir claramente los objetivos y requisitos del proyecto desde el principio. * No querer abarcar todas las funcionalidades posibles de la aplicación desde el principio: plantear la posibilidad de comenzar por una versión beta e ir incluyendo nuevas *features* con el tiempo. |
| *Escasez de desarrolladores* | --- |  |
| *Falta de atención a las vulnerabilidades de seguridad* | Alta | * Hacer pruebas recurrentes durante todo el ciclo de vida del desarrollo de la aplicación. * Utilizar herramientas de análisis de vulnerabilidades. * Dedicar tiempo a la formación en materia de seguridad. |
| *Falta de eficacia en las pruebas* | Media | * Automatizar las pruebas posibles para mejorar la eficiencia y la cobertura. * Diseñar un plan de pruebas que incluya pruebas de unidad, integración, regresión y rendimiento. * Consultar sobre qué pruebas son más adecuadas para este proyecto en concreto. |
| *Asociarse con la empresa equivocada* | --- |  |
| *Falta de conocimientos técnicos necesarios* | Alta | * + Preguntar y formarme en el entorno laboral de prácticas sobre herramientas, entornos de desarrollo y medidas que se implementan allí.   + Compartir conocimientos y experiencias con otros compañeros trabajando en sus propios proyectos.   + Dedicar tiempo a la investigación y el estudio de proyectos y aplicaciones similares. |
| *Problemas de salud / Accidentes o imprevistos* | Media | * Tratar de hacer las entregas con antelación para evitar las consecuencias derivadas de causas externas. * Priorizar unos hábitos de alimentación y sueño saludables, así como dedicar 30 minutos diarios al ocio o al ejercicio físico para evitar el conocido *burnout*. * Pausar en la medida de lo posible el resto de actividades laborales para priorizar la compleción de este proyecto. |

Tabla 5. Análisis de riesgos de desarrollo

No solo es importante identificar los riesgos o debilidades de un proyecto, sino también descubrir cuáles son sus puntos fuertes para poder aprovechar al máximo su potencial. Para ello es recomendable elaborar un análisis DAFO (Debilidades, Amenazas, Fortalezas y Oportunidades) que permita determinar de manera realista en qué situación nos encontramos y hacia dónde orientar los esfuerzos del equipo. En el blog de IEBS, una comunidad de profesores especializados en formación e innovación empresarial, se encuentra una publicación que detalla la importancia de un análisis DAFO en un proyecto: “Un análisis del macroentorno y del microentorno plasmado en el DAFO nos ayudará a establecer objetivos, estrategias y acciones” (Fuente, 2022).

Extraemos de la tabla anterior (*tabla 5*) aquellos riesgos más importantes de cara al lanzamiento comercial de la aplicación. Estos incluyen las dificultades técnicas por inexperiencia, la dependencia de agentes externos (adiestradores y usuarios) para su rentabilidad o la situación del mercado a nivel de competitividad y demanda.

Un aspecto positivo que ya se ha planteado desde el inicio es su alcance local: se centra en elaborar redes sólidas y sostenibles, que se traduce en servicios más eficientes y personalizados. También se le da importancia a la certificación de los adiestradores de la plataforma, previniendo estafas y malas prácticas en el sector. Por último, las oportunidades que dan lugar al posible crecimiento de la aplicación: las circunstancias (creciente conciencia sobre el bienestar animal, innovación tecnológica continua, más profesionales y voluntarios del sector…) apuntan a que hay proyección a futuro, tanto hacia expansión como hacia diversificación.

Con estos puntos, podemos elaborar el DAFO a continuación (*Figura 5*).



Figura 5. Análisis DAFO de Kandog

## Pliego de condiciones

El pliego de condiciones que detalla los términos que rigen la plataforma se encuentra en el Anexo de la documentación de “Kandog”. Su contenido incluye:

1. Responsabilidades y obligaciones: Los derechos y deberes tanto de los adiestradores como de los clientes de los adiestradores en relación con el uso de la plataforma “Kandog”. Contemplará aspectos como: la calidad del servicio prestado, el pago de las tarifas correspondientes, el cumplimiento de las normas de conducta, etc.
2. Confidencialidad y privacidad: La garantía de confidencialidad y la privacidad de la información tanto de los adiestradores como de sus clientes, abarcando aspectos como la protección de datos personales, la confidencialidad de la información comercial y cualquier otra información sensible compartida a través de la plataforma.
3. Términos de servicio: Información legal para el uso de la plataforma “Kandog”. Esto incluye aspectos como las condiciones de uso, las políticas de pago o las políticas de cancelación/devolución.
4. Protección de propiedad intelectual: Disposiciones para proteger los derechos de propiedad intelectual, concretamente los derechos de autor. Implica la prohibición de la reproducción no autorizada de contenido y de compartir recursos con terceros. Aquí se hace especial hincapié en la protección del trabajo del adiestrador.

# Análisis y diseño

## Modelado de datos. Análisis y diseño de la base de datos

Para poder estructurar y organizar la información en la aplicación de “Kandog” es imprescindible hacer un modelado de datos previo que nos permita relacionarlos eficientemente entre sí, darle escalabilidad, evitar redundancias o duplicidad de datos (por las incongruencias posteriores que pudieran acarrear) y, sobre todo, saber por dónde empezar a desarrollar. Una buena forma de optimizar las bases de datos es elaborar diagramas visuales que permitan entender la estructura del sistema antes de desarrollarlo.

### Diagrama E/R

Un diagrama E/R se basa en un modelo conceptual que representa entidades y sus relaciones. Para elaborarlo, es necesario identificar primero las entidades u objetos clave en el sistema de “Kandog”. Los imprescindibles son el adiestrador, los eventos que organiza (sesiones, charlas, demostraciones, etc.) y los perros a los que entrena. También es procedente incluir a los tutores caninos con la información necesaria para el adiestrador, como los datos de contacto o anotaciones específicas, y los recursos (informes, imágenes, vídeos, etc.) que el adiestrador maneje en la aplicación.

Identificadas las entidades y sus atributos, se procede a relacionarlos entre sí: El adiestrador es quien gestiona los eventos, que pueden ser sesiones o de otro tipo; las sesiones, principal actividad del adiestrador, se dirigen a un perro y pueden ser grupales o no; si son grupales, se identifica qué grupo es con un nombre y una descripción; y por último, los perros tienen un propietario o tutor legal. El adiestrador además tiene acceso a una tabla de recursos. El diagrama resultante se puede ver en la *Figura 6* de la siguiente página.

### Diagrama relacional

Una vez validado el diagrama E-R es posible transformarlo en un diagrama relacional, lo cual implica convertir el modelo conceptual en un modelo físico que represente las tablas y sus conexiones en una base de datos. Es necesario que cada tabla tenga una clave primaria que identifique cada registro de forma unívoca, y relacionar las tablas mediante claves foráneas que determinen la tabla “madre” y la tabla “hija”. El resultado esquemático se presenta en la *Figura 7* y puede verse ya desarrollado en una base de datos MySQL en la *Figura 8*.

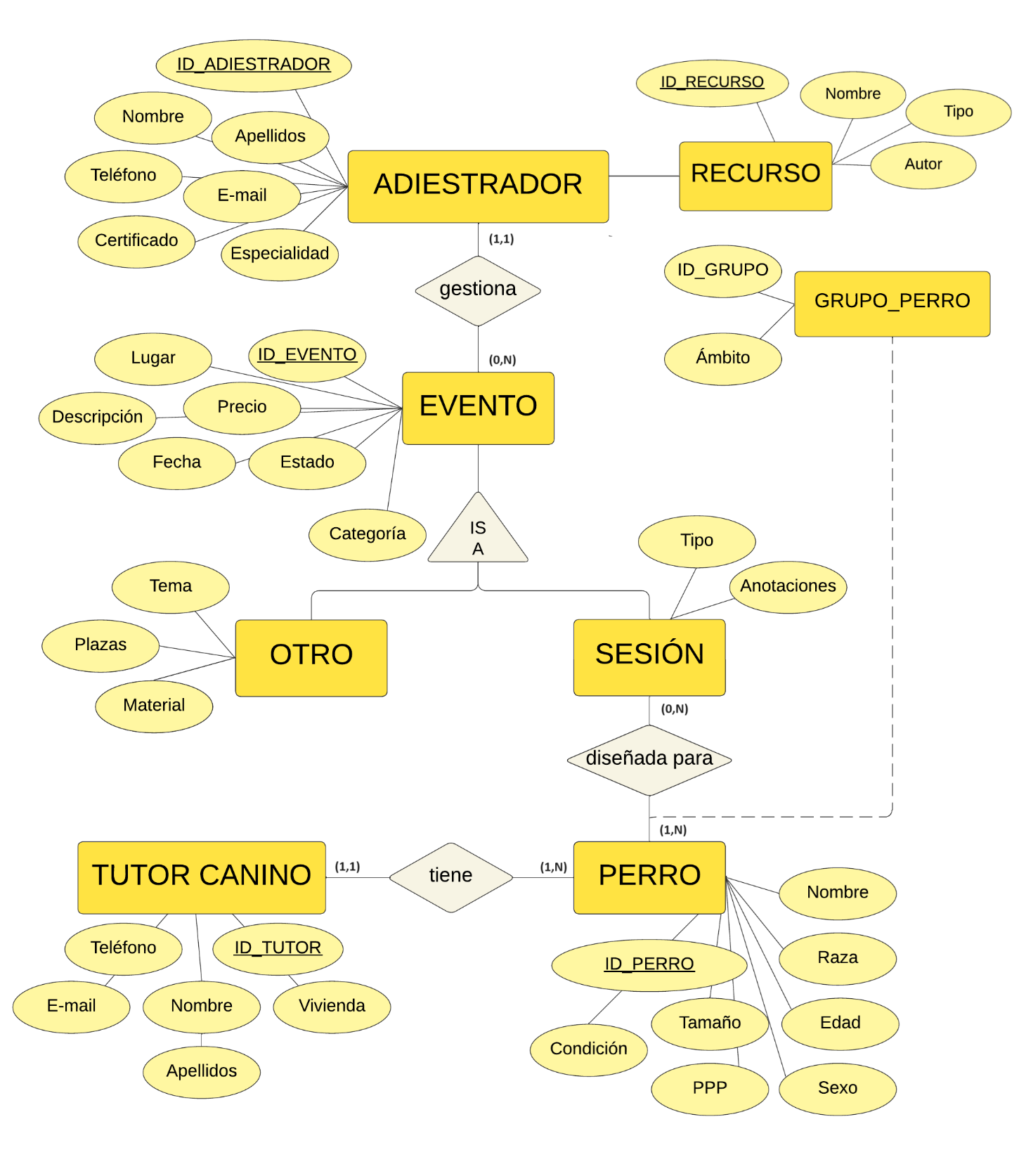
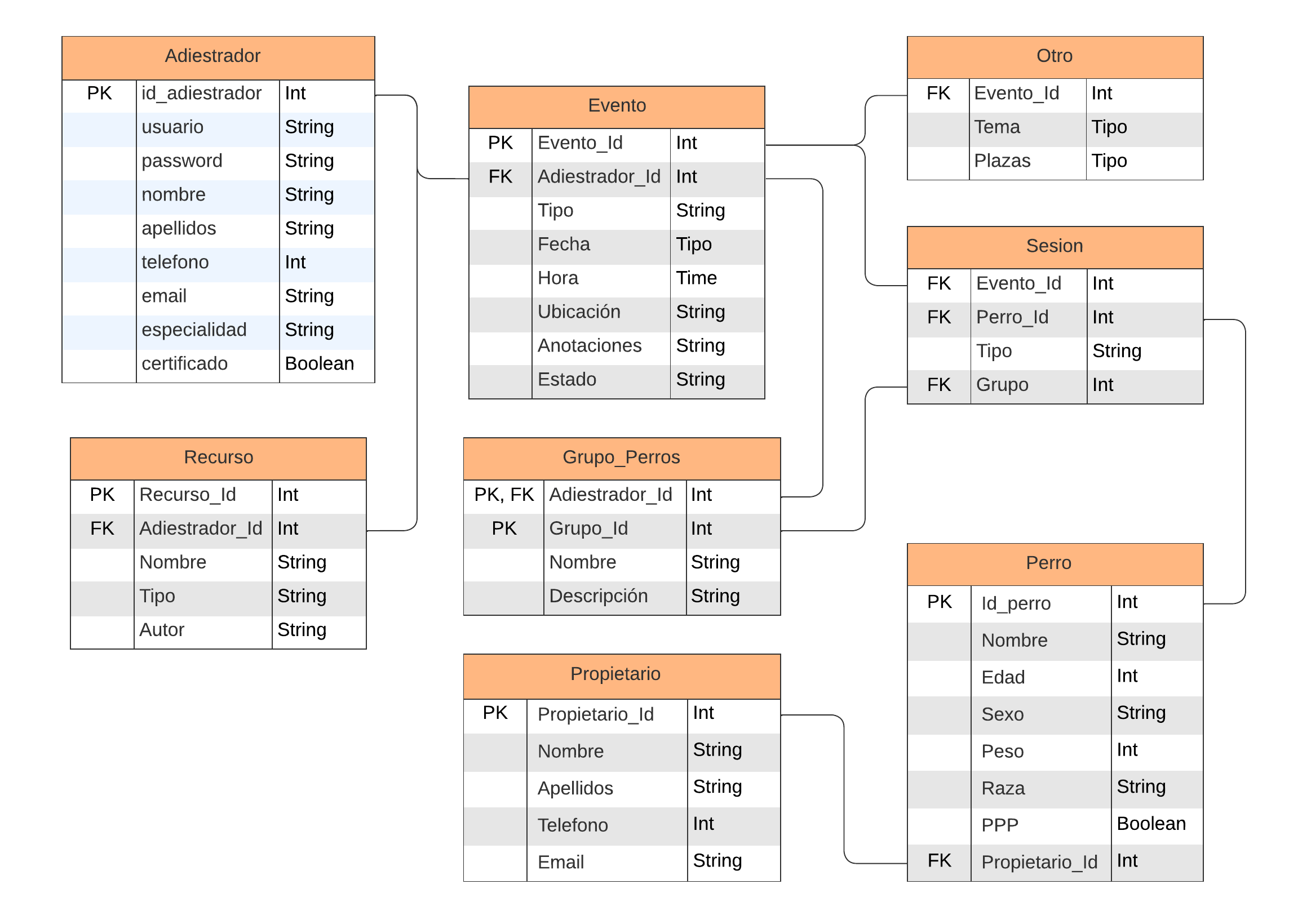


Figura 6. Diagrama de Entidad-Relación de “Kandog”. Realizado con Lucid.



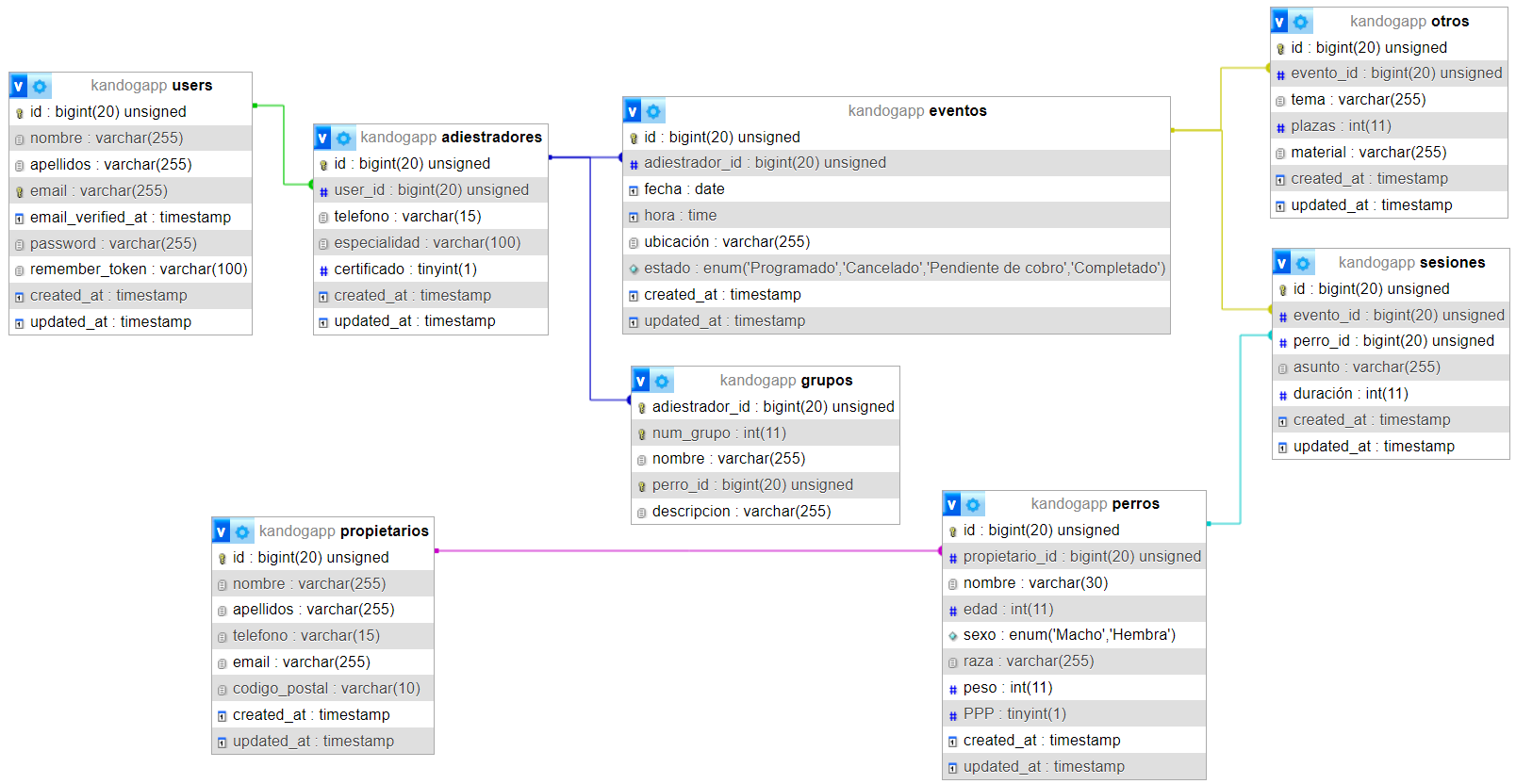
Figura 8. Diagrama Relacional de "Kandog". Realizado con Lucid.

Figura 7. Tablas relacionadas en una base de datos MySQL. PhpMyAdmin.

## Análisis y diseño del sistema funcional

“Kandog” es una aplicación para adiestradores caninos que permite gestionar clientes, perros, eventos de entrenamiento, y otros aspectos del negocio. Basándonos en sus requisitos funcionales podemos extraer el siguiente diagrama de casos de uso:

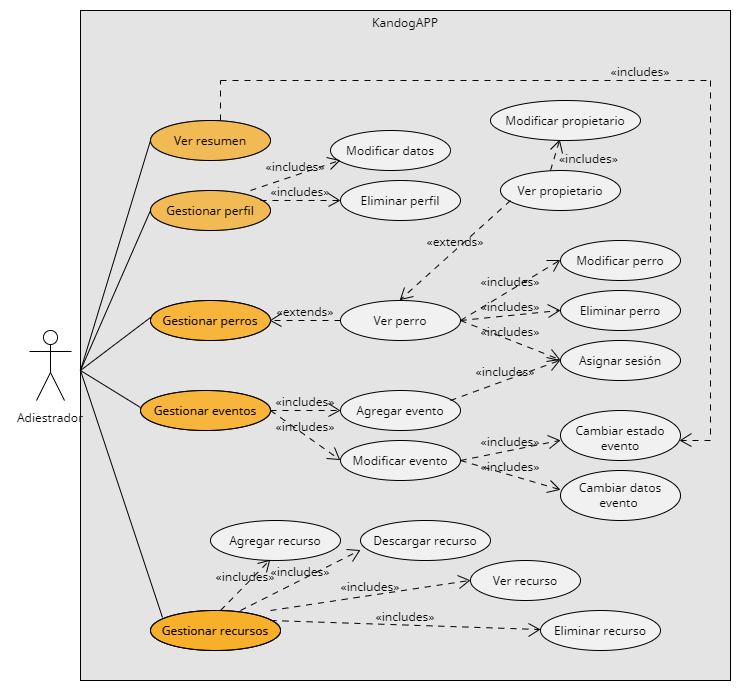


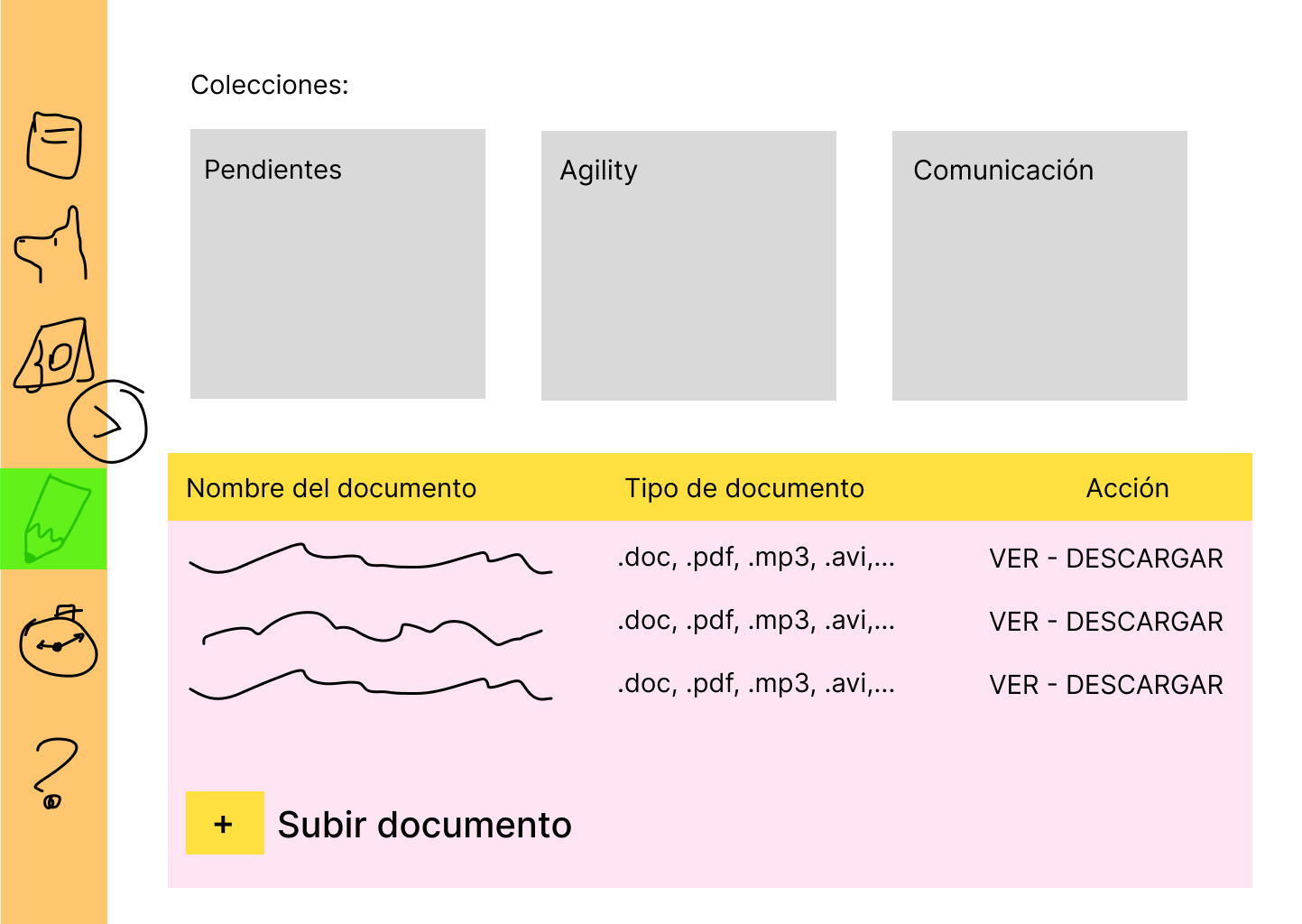
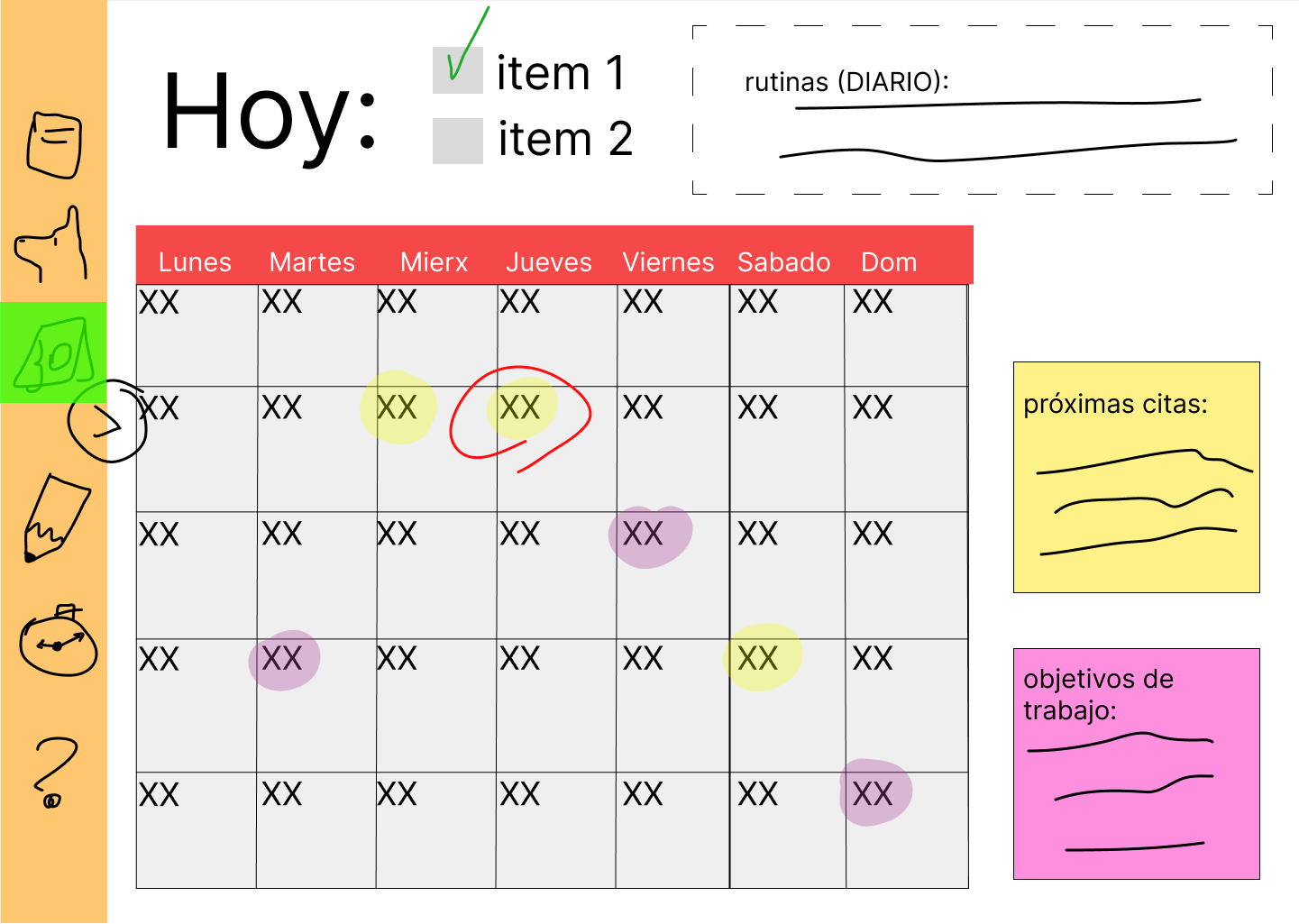
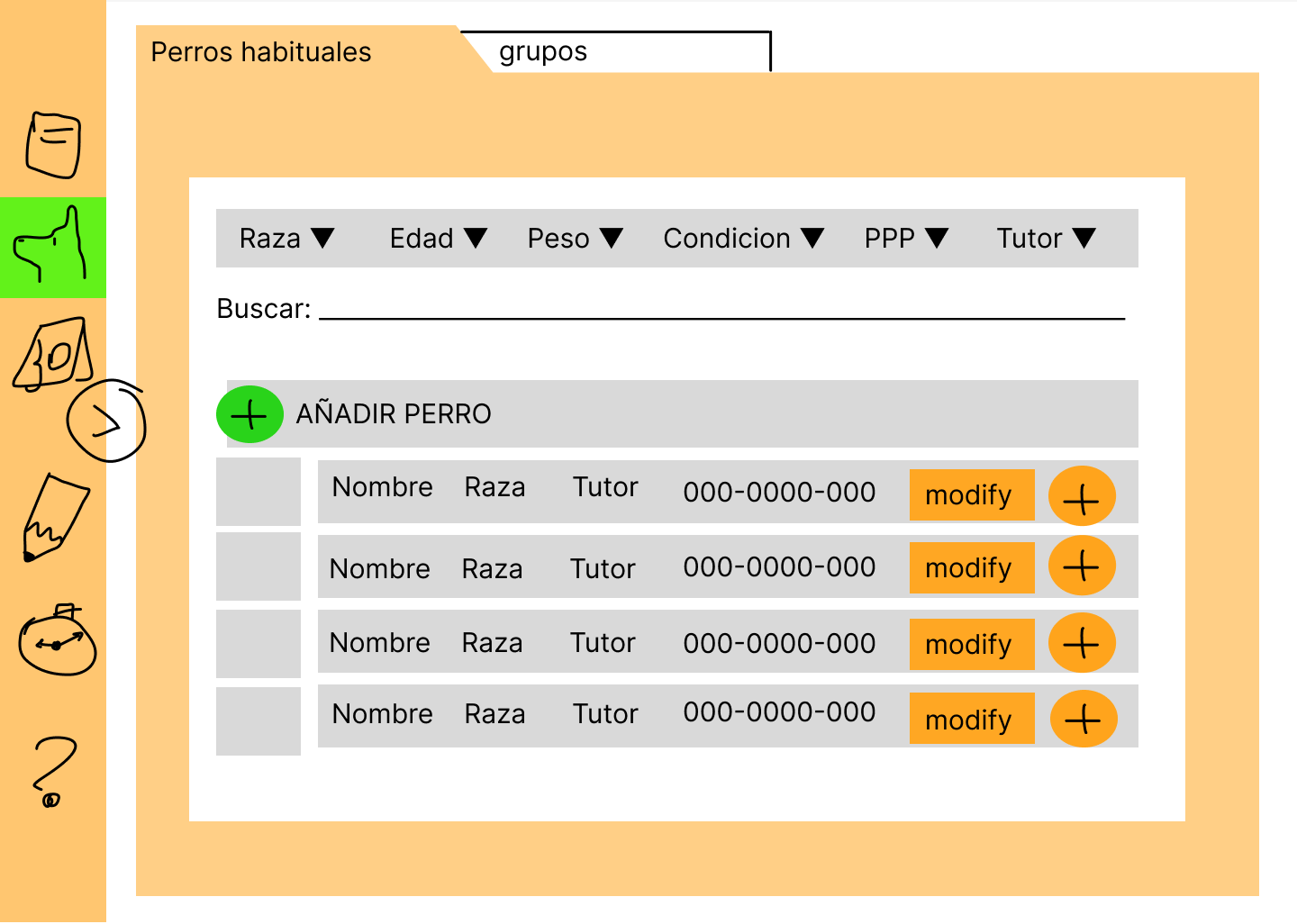
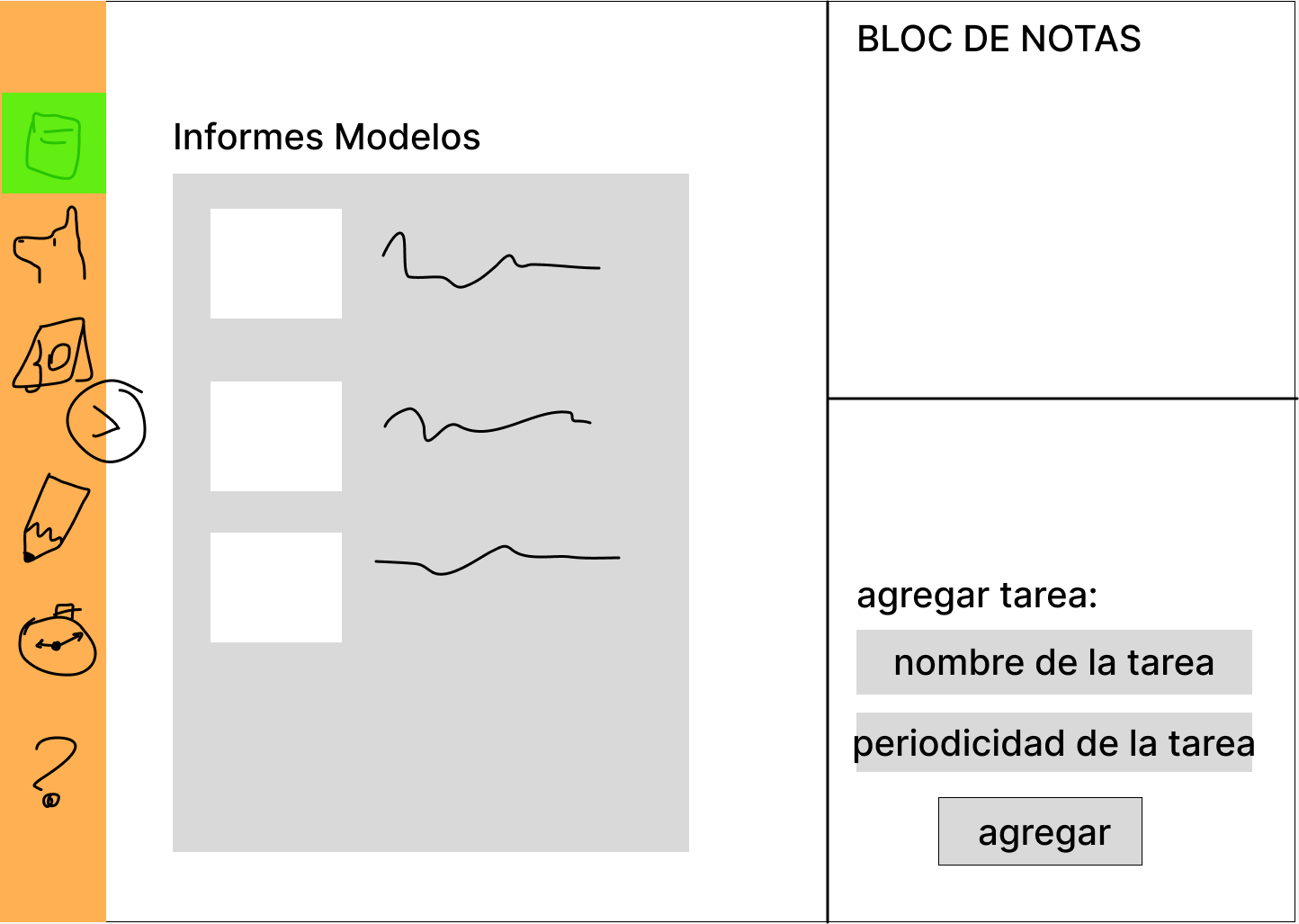
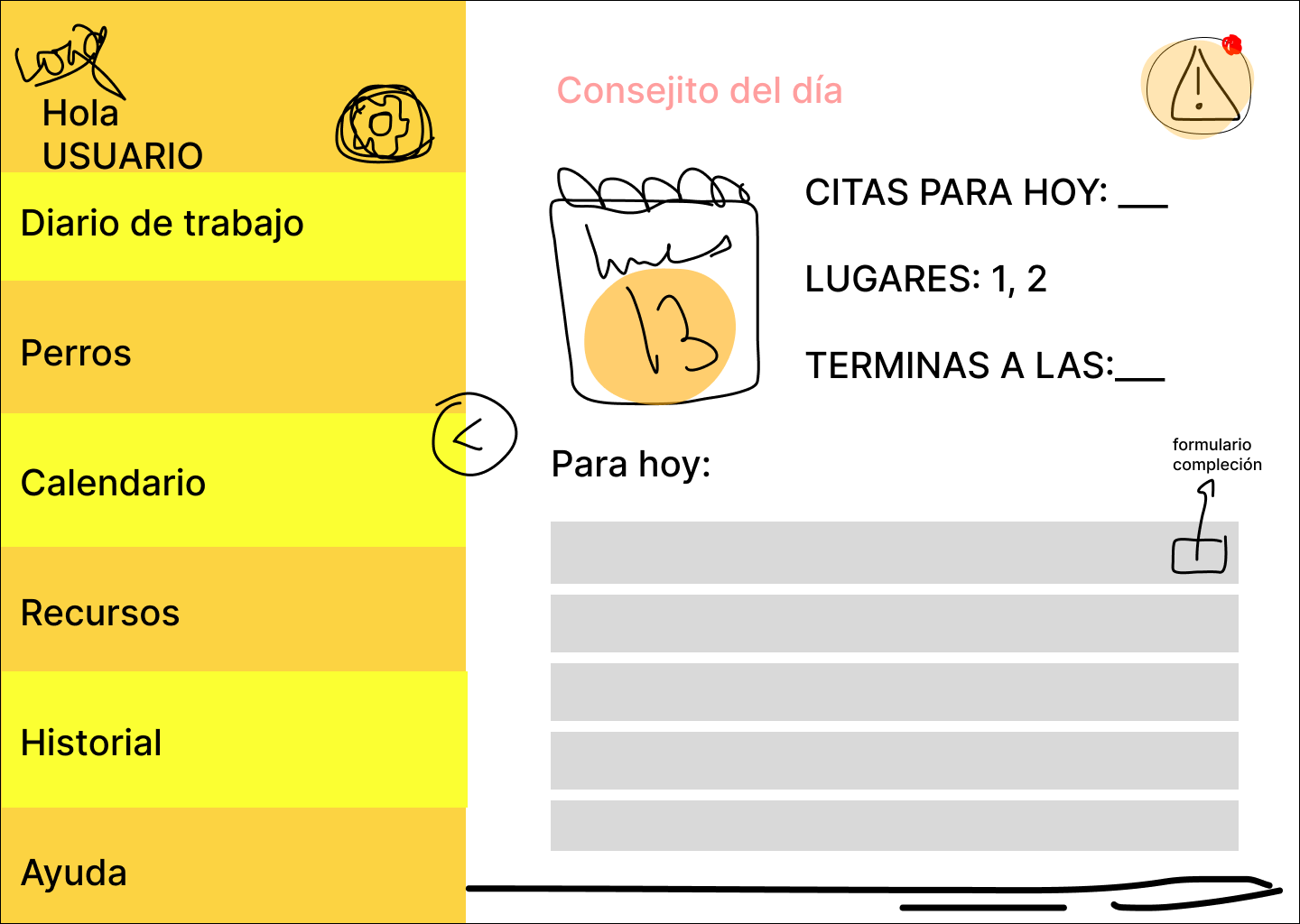
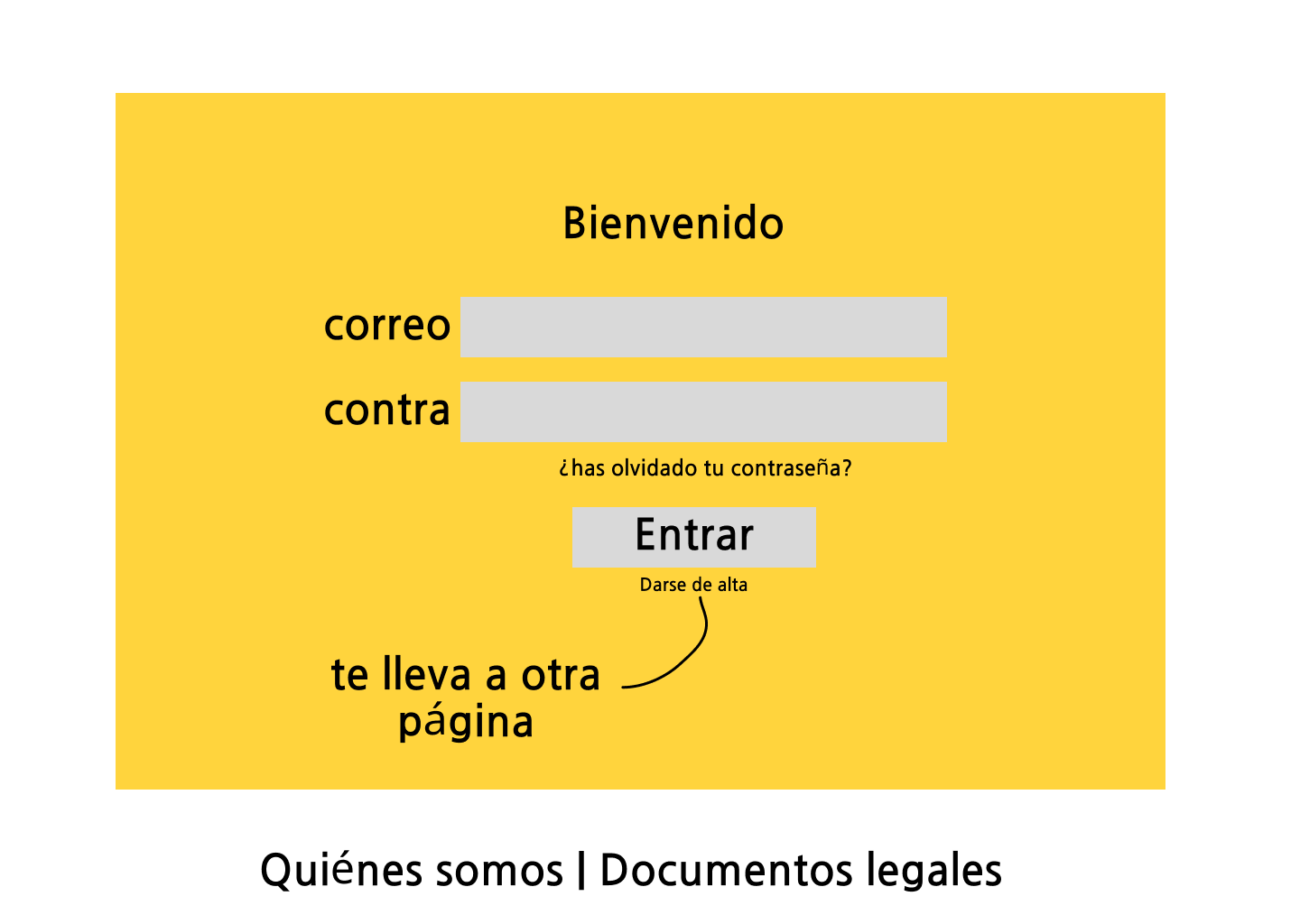
Figura 9. Diagrama de Casos de Uso. UMLetino.

Habría que considerar también los casos de uso del actor anónimo que llega a la página de aterrizaje, entre los cuales estaría el entrar como usuario (*login*) o registrarse como tal. Del mismo modo, el adiestrador que ha iniciado sesión podrá cerrarla en cualquier momento desde cualquier parte del sistema.

(Diagramas que procedan como el de ~~casos de uso~~, de clases, de secuencia, de actividad, de estados, de flujo...) Cuál más sería conveniente??

## Análisis y diseño de la interfaz de usuario. Mockups.

AUN NO HAY TITULOS NI RELACIONES CON CASOS DE USUARIO



¿Debería especificar los casos de uso uno por uno en una tabla y relacionarlos con los mock-ups? ¿Debería hacer otra cosa?

## Diseño de la arquitectura de la aplicación

### Tecnologías/Herramientas usadas y descripción de las mismas

En el desarrollo de la aplicación “Kandog” se han empleado las siguientes tecnologías:

* 1. Lenguajes de Programación: Para el manejo de datos e infraestructura lógica de la aplicación (*backend*) se utiliza PHP. Para el desarrollo de la interfaz y el contenido interactivo para el usuario (*frontend*) se ha optado por una combinación de HTML/CSS y JavaScript.
* Base de Datos: Se opta por una base de datos relacional MySQL que proporcione una estructura eficiente y organizada para almacenar la información de los usuarios.
  1. *Frameworks*:
* Laravel es un *framework* de desarrollo de aplicaciones web con PHP que sigue el patrón de diseño MVC (Modelo-Vista-Controlador). Es conocido por su elegante sintaxis y su conjunto de características que simplifican el desarrollo web, como son enrutamiento, mapeo relacional de objetos (ORM), plantillas varias y autenticación. Gracias a ellas es posible conseguir una aplicación web robusta y escalable.
* Bootstrap es un *framework* *front-end* de código abierto que facilita el desarrollo de interfaces de usuario responsive y estilizadas. Está basado en HTML, CSS y JavaScript, y ofrece una amplia gama de componentes y estilos predefinidos. Se utiliza en el desarrollo de la interfaz de usuario de “Kandog” con la intención de aprovechar sus componentes, estilos y plugins en la creación de una plataforma moderna y atractiva que ofrezca una experiencia de usuario óptima en cualquier pantalla.

Ambos frameworks son muy compatibles y se pueden utilizar juntos en diferentes capas de la misma aplicación web para potenciar conjuntamente tanto la eficiencia como la estética.

* 1. Gestión de Versiones y Colaboración: Se utiliza Git como sistema de control de versiones, así como GitHub para alojar el repositorio. Posibilita colaborar en el desarrollo y gestionar el código fuente de la aplicación.
  2. Entorno de Desarrollo Integrado (IDE): Visual Studio Code es una herramienta de edición de código avanzada que ofrece, entre otros, resaltado de sintaxis, autocompletado inteligente, integración con Git, una terminal integrada y extensiones opcionales que facilitan mucho el trabajo del programador. Es sencilla e intuitiva de utilizar y, a día de hoy, de las IDE más empleadas en el sector.
  3. Pruebas: La aplicación requiere de un entorno en el que poder probarla de forma local antes de publicarla en un servidor de hosting en línea. XAMPP proporciona un servidor web Apache, una base de datos MySQL e intérpretes de PHP para que podamos visualizar el producto terminado antes de su lanzamiento a producción.
  4. Despliegue y Hosting: Una vez terminado el proyecto y habiendo superado las pruebas iniciales, podrían considerarse servicios de alojamiento en la nube como Amazon Web Services (AWS), Google Cloud Platform (GCP), Microsoft Azure, o plataformas de despliegue específicas como Heroku, Netlify o Vercel.

### Arquitectura de componentes de la aplicación

Laravel, el *framework* o paradigma del que partimos, tiene una estructura predefinida basada en un modelo MVC (Modelo-Vista-Controlador).

Pantalla de computadora con letras

Descripción generada automáticamente con confianza mediaEn la *Figura 10* vemos la estructura predefinida por un proyecto de Laravel 11. Para el desarrollo de “Kandog” es suficiente conocer los siguientes elementos:

En la carpeta *app/* se encuentra la mayor parte del código de la aplicación, incluyendo modelos, controladores y servicios.

* Cada modelo representa una tabla o colección de datos, y permite interactuar con la base de datos de una manera más intuitiva y orientada a objetos. En lugar de escribir consultas SQL, se pueden usar métodos del modelo para obtener, insertar, actualizar o eliminar datos.
* Cada controlador sirve de puente entre la capa de vista y la capa de modelo. Se responsabiliza de recibir solicitudes, procesarlas y devolver respuestas.

*Config/* contiene archivos de configuración que permiten modificarla sin necesidad de cambiar el código fuente principal. Puede ser útil a la hora de configurar distintos idiomas.

El directorio *database/* es donde guardamos las migraciones y los seeders para interactuar con la base de datos.

En la carpeta *public/* tenemos el punto de entrada a la aplicación, que por defecto se llama *index.php*.

En *resources/* tenemos los recursos para la interfaz de usuario como imágenes, archivos CSS o Javascript. Uno de sus subdirectorios clave es *views/* o “vistas”, que es donde recopilaremos las distintas pantallas que verá luego el usuario.

Otra de las carpetas más importantes es *routes/*, donde tendremos archivos que definan las rutas de aplicación para nuestra web, API y consola. Estas instrucciones controlan el flujo de la aplicación y permiten que funcione correctamente determinando cómo se gestionan las solicitudes HTTP y qué controladores o acciones se ejecutan en respuesta a esas solicitudes.

Por último, tenemos el archivo *.env* para la configuración de las variables de entorno, que contiene información sensible que no pertenece a repositorios públicos. Nuestro sistema de control de versiones lo ignora por defecto (se incluye en *.gitignore*).

Figura 10. Arquitectura de Laravel 11.

Basándonos en esta arquitectura planteada, tendremos un controlador por cada modelo planteado, así como unos elementos comunes a todas las vistas (como la barra de navegación o el *footer*) que se guardarán en una carpeta llamada *partials/*.

Como se ha mecnionado, Laravel se encarga de generar y ejecutar las consultas SQL necesarias en segundo plano para realizar las operaciones CRUD en la base de datos utilizando el ORM Eloquent. Esto hace que el desarrollo sea más fácil, seguro y menos propenso a errores al evitar la necesidad de escribir consultas SQL manualmente. Sin embargo, las sentencias SQL necesarias que se ejecutarían en segundo plano se encuentran en el Anexo X.

¿Qué más?

# Conclusiones

# Bibliografía y webgrafía

AON - Seguros De Mascotas. (2023). Causas del abandono animal y cómo evitarlo. *ExpertoAnimal.* <https://www.expertoanimal.com/causas-del-abandono-animal-y-como-evitarlo-25555.html>

Ayuntamiento de Zaragoza. (2013). *Ordenanza municipal sobre la Protección, la Tenencia responsable y la Venta de Animales.* Oficina de Transparencia y Gobierno Abierto. <https://www.zaragoza.es/sede/servicio/normativa/4523>

Ayuntamiento de Zaragoza. (2023). El Ayuntamiento de Zaragoza permite el acceso de animales domésticos a las dependencias municipales. *Zaragoza Noticias*. <https://www.zaragoza.es/sede/servicio/noticia/318224>

Ayuntamiento de Zaragoza. (2023). Zaragoza quiere potenciar su carácter de «ciudad pet friendly» con la edición de un manual de convivencia con mascotas en el comercio y la hostelería. *Zaragoza Noticias*. <https://www.zaragoza.es/sede/servicio/noticia/319724>

BBVA. (2022). *Visión y Actitudes hacia los Animales en la Sociedad Española*. <https://www.fbbva.es/wp-content/uploads/2022/01/Presentacion-Estudio-de-Animales.pdf>

Benito, F. J. (2023). España ya tiene el doble de mascotas que niños. *Información*. <https://www.informacion.es/medio-ambiente/2023/01/09/espana-suma-15-millones-mascotas-68542763.html>

BOA - Ley 7/2023, de 23 de febrero, de medidas para la implantación y desarrollo en Aragón de tecnologías en la nube (Tecnologías Cloud)*.* *«BOE»*, de 13 de marzo de 2023. <https://www.boa.aragon.es/cgi-bin/EBOA/BRSCGI?CMD=VEROBJ&MLKOB=1265652460505>

BOE-A-1991-628 Ley 3/1991, de 10 de enero, de Competencia Desleal. *«BOE» núm. 10,* de 11 de enero de 1991. <https://www.boe.es/eli/es/l/1991/01/10/3/con>

BOE-A-1996-1072 Ley 7/1996, de 15 de enero, de Ordenación del Comercio Minorista, *«BOE» núm. 15*, de 17 de enero de 1996 <https://www.boe.es/buscar/act.php?id=BOE-A-1996-1072>

BOE-A-1996-8930 Real Decreto Legislativo 1/1996, de 12 de abril, por el que se aprueba el texto refundido de la Ley de Propiedad Intelectual, regularizando, aclarando y armonizando las disposiciones legales vigentes sobre la materia. *«BOE» núm. 97*, de 22 de abril de 1996. <https://www.boe.es/eli/es/rdlg/1996/04/12/1/con>

BOE-A-2001-23093 Ley 17/2001, de 7 de diciembre, de Marcas. *«BOE» núm. 294*, de 08 de diciembre de 2001. <https://www.boe.es/eli/es/l/2001/12/07/17/con>

BOE-A-2002-13758 Ley 34/2002, de 11 de julio, de servicios de la sociedad de la información y de comercio electrónico*.* *«BOE» núm. 166*, de 12 de julio de 2002. <https://www.boe.es/eli/es/l/2002/07/11/34/con>

BOE-A-2007-20555 Real Decreto Legislativo 1/2007, de 16 de noviembre, por el que se aprueba el texto refundido de la Ley General para la Defensa de los Consumidores y Usuarios y otras leyes complementarias. *«BOE» núm. 287*, de 30 de noviembre de 2007. <https://www.boe.es/eli/es/rdlg/2007/11/16/1/con>

BOE-A-2018-16673 Ley Orgánica 3/2018, de 5 de diciembre, de Protección de Datos Personales y garantía de los derechos digitales. *«BOE» núm. 294*, de 6 de diciembre de 2018. <https://www.boe.es/eli/es/lo/2018/12/05/3/con>

BOE-A-2023-7936 Ley 7/2023, de 28 de marzo, de protección de los derechos y el bienestar de los animales*.* *«BOE» núm. 75,* de 29 de marzo de 2023. <https://www.boe.es/eli/es/l/2023/03/28/7>

China, L., Mills, D. S., & Cooper, J. J. (2020). Efficacy of dog training with and without remote electronic collars vs. a focus on positive reinforcement. *Frontiers in Veterinary Science*, *7*, 547533. <https://www.frontiersin.org/articles/10.3389/fvets.2020.00508/full>

Decreto nº 265 del 18 noviembre 2014, sobre condiciones para la suelta de Perros en los Parques públicos. *Boletín Oficial Provincia Zaragoza, Oficina De Transparencia Y Gobierno Abierto*. <https://www.zaragoza.es/sede/servicio/normativa/5423>

De Castro, A. C. V., Fuchs, D., Morello, G. M., Pastur, S., de Sousa, L., & Olsson, I. A. S. (2020). Does training method matter? Evidence for the negative impact of aversive-based methods on companion dog welfare. *Plos one*, *15*(12), e0225023. <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0225023>

DOUE-L-2016-80807 Reglamento (UE) 2016/679 del Parlamento Europeo y del Consejo, de 27 de abril de 2016, relativo a la protección de las personas físicas en lo que respecta al tratamiento de datos personales y a la libre circulación de estos datos y por el que se deroga la Directiva 95/46/CE (Reglamento general de protección de datos). *«DOUE» núm. 119*, de 4 de mayo de 2016. <https://www.boe.es/buscar/doc.php?id=DOUE-L-2016-80807>

Drumond, C. (s. f.). *Gestión ágil de proyectos para equipos de software*. Atlassian. <https://www.atlassian.com/es/agile/project-management>

EKCIT - European Knowledge Center for Information Technology. (2022). *Agile: ¿Cómo aprovechar la metodología para el desarrollo iterativo de proyectos?* TIC Portal. <https://www.ticportal.es/glosario-tic/agile-project-management>

FormulaTV. (2010) Buen debut de 'Malas pulgas' en Cuatro. <https://www.formulatv.com/noticias/14932/audiencias-buen-debut-de-malas-pulgas-en-cuatro>

Fuente, O. (2022). *Qué es un Análisis DAFO: usos, ventajas y ejemplos*. IEBS. <https://www.iebschool.com/blog/analisis-dafo-creacion-empresas/>

Jacinto, A. (2024). *10 Common Software Development Risks and How to Deal With Them*. Startechup Inc. <https://www.startechup.com/es/blog/10-common-software-development-risks/>

Microsoft. (2022). *VS Code – FAQ*. <https://code.visualstudio.com/docs/supporting/faq>

Northware. (2022). *Requerimientos en el desarrollo de software y aplicaciones*. Northware. <https://www.northware.mx/blog/requerimientos-en-el-desarrollo-de-software-y-aplicaciones>

Ovando Ortega, D. J. (2019). *Bootstrap y Laravel, herramientas para el desarrollo de aplicaciones web.*  [Tesina, Universidad Politécnica de Sinaloa, México]. <http://repositorio.upsin.edu.mx/Fragmentos/tesinas/142016030030OvandoOrtegaDenzelJavier10843.pdf>

Ramírez, H. (2022). *Requisitos legales para crear una página web*. Grupo Atico34. <https://protecciondatos-lopd.com/empresas/requisitos-legales-pagina-web/>

Save the Children. (2020) *Save the Children calcula que en España más del 25% de niños y niñas han sido víctima de maltrato en su hogar*. <https://www.savethechildren.es/notasprensa/save-children-calcula-que-en-espana-mas-del-25-de-ninos-y-ninas-han-sido-victima-de>

Stsepanets, A. (2023). *Cómo hacer un presupuesto de un proyecto correctamente*. GanttPRO Blog. <https://blog.ganttpro.com/es/presupuesto-proyecto/>

Torres, O. P. (2022). *Gestión de riesgos en proyectos de software*. <https://www.piranirisk.com/es/blog/gestion-de-riesgos-proyectos-de-software>

Tranvía Zaragoza. (2014). *Con mascotas*. <https://www.tranviasdezaragoza.es/es/informacion/con-mascotas>

Visual Paradigm (s.f.) *Herramienta de diagramas de Gantt en línea*. Visual Paradigm Online. <https://online.visual-paradigm.com/es/diagrams/features/gantt-chart-tool/>

# Anexos

PARA ELABORAR MI AVISO LEGAL: <https://protecciondatos-lopd.com/empresas/modelo-aviso-legal-web/>

PARA ELABORAR POLÍTICA DE PRIVACIDAD: <https://protecciondatos-lopd.com/empresas/politica-de-privacidad-web/>

PARA ELABORAR POLÍTICA DE COOKIES: <https://protecciondatos-lopd.com/empresas/politica-de-cookies/>

PARA ELABORAR CONDICIONES DE CONTRATACIÓN: <https://protecciondatos-lopd.com/empresas/condiciones-generales-venta/>

PARA EL PLIEGO DE CONDICIONES: <https://es.sitew.com/Como-modificar-un-sitio-web/crear-pliego-de-condiciones-sitio-web>

Y esta tabla pa qué

|  |  |  |
| --- | --- | --- |
| Tabla: | Campos: | Métodos: |
| Adiestrador | * id\_usuario * Nombre * Apellidos * Email * Contraseña | ˪ *nuevoUsuario(nombre, correo, contraseña)*  ˪ *iniciarSesion(correo, contraseña)*  ˪ *cerrarSesion()*  ˪ *verPerfil()*  ˪ *editarPerfil(nombre, telefono, correo)* |
| Sesión | * id\_sesion * id\_usuario * fecha/hora * lugar * id\_perro * precio * estado * grupo | ˪ *agregarSesion(fecha, hora, lugar, id\_perro)*  ˪ *editarSesion(id\_sesion, fecha, hora, lugar, id\_perro)*  ˪ *eliminarSesion(id\_sesion)*  ˪ *listarSesiones(id\_usuario)*  ˪ *verSesion(id\_sesion)* |
| Perro | * id\_perro * nombre * sexo * peso * raza * PPP * id\_propietario | ˪ *agregarPerro(nombre, sexo, peso, raza, ppp, id\_propietario)*  ˪ *editarPerro(id\_perro, nombre, sexo, peso, raza, ppp, id\_propietario)*  ˪ *eliminarPerro(id\_perro)* |
| Propietario | * id\_propietario * nombre * teléfono * correo electrónico * comentario | ˪ *agregarPropietario(nombre, telefono, correo)*  ˪ *editarPropietario(id\_propietario, nombre, telefono, correo, comentario)*  ˪ *eliminarPropietario(id\_propietario)* |
| Recurso | * id\_recurso * id\_usuario * nombre * tipo * ruta | ˪ *agregarDocumento()*  ˪ *verDocumentos()*  ˪ *descargarDocumento()*  ˪ *eliminarDocumento()* |

ANEXO X:

|  |
| --- |
| Tabla *Usuarios* |
| INSERT INTO Usuarios (nombre, email, contrasena)  VALUES (%s, %s, %s); |
| Tabla *Sesiones* |
| CREATE TABLE Sesion (  id\_sesion INT AUTO\_INCREMENT PRIMARY KEY,  fecha DATE,  hora TIME,  lugar VARCHAR(255),  id\_perro INT,  precio DECIMAL(10, 2),  estado VARCHAR(50),  FOREIGN KEY (id\_perro) REFERENCES Perro(id\_perro)  ); |
| Tabla *Perros* |
| CREATE TABLE Perro (  id\_perro INT AUTO\_INCREMENT PRIMARY KEY,  nombre VARCHAR(255),  sexo VARCHAR(10),  peso DECIMAL(5, 2),  raza VARCHAR(255),  id\_propietario INT,  FOREIGN KEY (id\_propietario) REFERENCES Propietario(id\_propietario)  ); |
| Tabla *Propietarios* |
| CREATE TABLE Propietario (  id\_propietario INT AUTO\_INCREMENT PRIMARY KEY,  nombre VARCHAR(255),  telefono VARCHAR(20),  email VARCHAR(255)  ); |
|  |
|  |
| Tabla *Recursos* |
| CREATE TABLE Recurso (  id\_recurso INT AUTO\_INCREMENT PRIMARY KEY,  id\_usuario INT,  nombre VARCHAR(255),  autor VARCHAR(255),  tipo VARCHAR(50),  ruta VARCHAR(255),  FOREIGN KEY (id\_usuario) REFERENCES Usuario(id\_usuario)  ); |

En cuanto a los métodos relacionados con cada controlador de entidad, podemos encontrar:

* **Usuarios** (Adiestradores): id\_usuario, nombre, apellidos, correo electrónico, contraseña.

˪ *nuevoUsuario(nombre, correo, contraseña)*: Permite a un usuario nuevo registrarse en la aplicación proporcionando su nombre, correo electrónico y contraseña.

INSERT INTO Usuarios (nombre, email, contrasena)

VALUES (%s, %s, %s);

˪ *iniciarSesion(correo, contraseña):* Permite a un usuario iniciar sesión en la aplicación utilizando su correo electrónico y contraseña.

SELECT \*

FROM Usuarios

WHERE email = %s AND contrasena = %s;

˪ *cerrarSesion():* Permite a un usuario cerrar sesión en la aplicación. Esto se maneja en el lado del cliente, de forma que no necesita ninguna sentencia SQL.

˪ *verPerfil():* Permite al usuario ver su perfil, incluyendo información como nombre, correo electrónico y otros detalles relevantes.

SELECT nombre, apellidos, email

FROM Usuarios

WHERE id\_usuario = %s;

˪ *editarPerfil(nombre, telefono, correo):* Permite al usuario editar los detalles de su perfil, como el nombre, el número de teléfono y el correo electrónico.

UPDATE Usuarios

SET nombre = %s, apellidos = %s, email = %s

WHERE id\_usuario = %s;

˪ *eliminarPerfil():* Permite al usuario editar los detalles de su perfil, como el nombre, el número de teléfono y el correo electrónico.

DELETE FROM Usuarios

WHERE id\_usuario = %s;

* **Sesiones**: id\_sesion, id\_usuario, fecha, hora, lugar, id\_perro.

˪ *verSesion(id\_sesion)*: Permite obtener los detalles de una sesión específica en la base de datos utilizando su identificador único.

SELECT \* FROM Sesiones

WHERE id\_sesion = %s

˪ *agregarSesion(fecha, hora, lugar, id\_perro)*: Permite al adiestrador agregar una nueva sesión a su agenda, especificando la fecha, hora, lugar y el perro al que está dirigida la sesión.

INSERT INTO Sesiones (fecha, hora, lugar, id\_perro)

VALUES (%s, %s, %s, %s);

˪ *editarSesion(id\_sesion, fecha, hora, lugar, id\_perro)*: Permite al adiestrador editar los detalles de una sesión existente en su agenda.

UPDATE Sesiones

SET fecha = %s, hora = %s, lugar = %s, id\_perro = %s

WHERE id\_sesion = %s;

˪ *eliminarSesion(id\_sesion)*: Permite al adiestrador eliminar una sesión de su agenda.

DELETE FROM Sesiones

WHERE id\_sesion = %s;

˪ *listarSesiones(id\_usuario)*: Permite obtener una lista de todas las sesiones asociadas a un adiestrador específico en la base de datos.

SELECT \* FROM Sesiones

WHERE id\_usuario = %s

˪ *listarSesionesHoy(id\_usuario)*: Permite ver al adiestrador una lista de las sesiones para ‘hoy’.

SELECT \* FROM Sesiones

WHERE id\_usuario = %s

AND DATE(fecha) = CURDATE();

˪ *resumenSesiones(id\_usuario)*: Permite obtener un resumen de las sesiones completadas por un adiestrador específico incluyendo el dinero ganado.

SELECT COUNT(\*) AS sesiones\_completadas, SUM(precio) AS dinero\_ganado

FROM Sesiones

WHERE id\_usuario = %s AND estado = 'completada';

˪ sesionesMes(id\_usuario): Permite saber cuántas sesiones ha realizado un adiestrador en un mes determinado.

SELECT COUNT(\*) AS sesionesMes

FROM Sesiones

WHERE id\_usuario = %s AND MONTH(fecha) = %s AND YEAR(fecha) = %s

* **Perros**: id\_perro, nombre, sexo, peso, raza, id\_propietario.

˪ *agregarPerro(nombre, sexo, peso, raza, id\_propietario)*: Permite al adiestrador agregar un nuevo perro a su lista de perros, especificando detalles y propietario del perro.

INSERT INTO Perros (nombre, sexo, peso, raza, id\_propietario)

VALUES (%s, %s, %s, %s, %s);

˪ *editarPerro(id\_perro, nombre, sexo, peso, raza, id\_propietario)*: Permite al adiestrador editar los detalles de un perro existente en su lista.

UPDATE Perros

SET nombre = %s, sexo = %s, peso = %s, raza = %s, id\_propietario = %s

WHERE id\_perro = %s;

˪ *eliminarPerro(id\_perro)*: Permite al adiestrador eliminar un perro de su lista.

DELETE FROM Perros

WHERE id\_perro = %s;

˪ *detallePerro(id\_perro)*: Permite al adiestrador ver los detalles de un perro concreto.

SELECT id\_perro, nombre, sexo, peso, raza, id\_propietario

FROM Perros

WHERE id\_perro = %s;

* **Propietarios**: id\_propietario, nombre, teléfono, correo electrónico.

˪ *agregarPropietario(nombre, telefono, correo)*: Permite al adiestrador agregar un nuevo propietario a su lista, especificando su nombre, número de teléfono y correo electrónico.

INSERT INTO Propietarios (nombre, telefono, email)

VALUES (%s, %s, %s);

˪ *editarPropietario(id\_propietario, nombre, telefono, email, comentario)*: Permite al adiestrador editar los detalles de un propietario existente en su lista.

UPDATE Propietarios

SET nombre = %s, telefono = %s, email = %s, comentario = %

WHERE id\_propietario = %s;

˪ *eliminarPropietario(id\_propietario)*: Permite al adiestrador eliminar un propietario de su lista.

DELETE FROM Propietarios

WHERE id\_propietario = %s;

˪ *detallePropietario(id\_propietario)*: Permite al adiestrador ver un propietario concreto.

SELECT id\_propietario, nombre, telefono, email

FROM Propietarios

WHERE id\_propietario = %s;

* **Recursos**: id\_documento, id\_usuario, nombre, tipo, ruta.

˪ *agregarDocumento()*: Permite cargar nuevos documentos al repositorio. Esto implica almacenar el archivo físico en el servidor y registrar su información en la tabla Documento.

INSERT INTO Recursos(id\_usuario, nombre, autor, tipo, ruta)

VALUES (%s, %s, %s, %s, %s)

˪ *verDocumentos()*: Permite ver la lista de documentos que se han subido al repositorio, junto con información relevante como el nombre y el tipo de documento.

SELECT nombre, tipo

FROM Recursos

WHERE id\_usuario = %s;

˪ *descargarDocumento()*: descargar un documento específico del repositorio. No requiere una consulta SQL explícita puesto que se realiza en el lado del servidor.

˪ *eliminarDocumento()*: eliminar un documento existente del repositorio.

DELETE FROM Recursos

WHERE id\_documento = %s;

CHECK

1. Portada. Datos del proyecto: Título, Ciclo, Curso, Fecha, Centro, Logo del centro.

Datos del alumno y datos de los profesores tutores

2. Descripción del proyecto

2.1. Contexto del proyecto

2.1.1. Ámbito y entorno

2.1.2. Análisis de la realidad.

2.1.3. Solución y justificación de la solución propuesta

2.1.4. Destinatarios.

2.2. Objetivo del proyecto

2.2. Objetivo del proyecto en lengua extranjera.

2.3 Marco legal

3. Documento de Acuerdo del proyecto.

3.1. Requisitos funcionales y no funcionales / Historias de usuario

3.2. Tareas

3.3. Metodología a seguir para la realización del proyecto.

3.4. Planificación temporal de tareas.

3.5. Presupuesto (gastos, ingresos, beneficio)

3.7. Análisis de riesgos

3.6. Pliego de condiciones

4. Documento de análisis y diseño

4.1 Modelado de datos. Análisis y diseño de la base de datos

4.1.1 Diagrama E/R

4.1.2 Diagrama relacional

4.2 Análisis y diseño del sistema funcional (Diagramas que procedan como el de casos

de uso, de clases, de secuencia, de actividad, de estados, de flujo...)

4.3 Análisis y diseño de la interfaz de usuario. Mockups.

4.4 Diseño de la arquitectura de la aplicación

4.4.1 Tecnologías/Herramientas usadas y descripción de las mismas

4.4.2 Arquitectura de componentes de la aplicación

5. Documento de implementación, pruebas e implantación del sistema

5.1 Implementación

5.2 Pruebas

5.2.1 Pruebas unitarias

5.2.2 Pruebas funcionales

5.3. Instalación/Despliegue y configuración.

5.4. Manual de usuario

6. Documento de cierre

6.1. Resultados obtenidos y conclusiones.

6.2. Diario de bitácora

7. Bibliografía.

8. Anexos.