

Formando parte del proyecto:



MÓDULO PROYECTO

CFGS Desarrollo de Aplicaciones Multiplataforma
Informática y Comunicaciones

WildDetect

Tutor individual: Cristina Silván Pardo

Tutor colectivo: Cristina Silván Pardo

Año: 2025

Fecha de presentación: 22/05/2025

Nombre y Apellidos: Jose Luis Mayo Calvo

Email: josel.mayc1@gmail.com



Contenido de la memoria

1	Identificación proyecto	4
2	Organización de la memoria	4
3	Descripción general del proyecto	6
3.1	Objetivos	6
3.2	Cuestiones metodológicas	7
3.3	Entorno de trabajo (tecnologías de desarrollo y herramientas)	7
4	Descripción general del producto	8
4.1	Visión general del sistema: límites del sistema, funcionalidades básicas, usuarios y/o otros sistemas con los que pueda interactuar.	8
4.2	Descripción breve de métodos, técnicas o arquitecturas(m/t/a) utilizadas.	9
4.3	Despliegue de la aplicación indicando plataforma tecnológica, instalación de la aplicación y puesta en marcha	10
5	Planificación y presupuesto	10
6	Documentación Técnica: análisis, diseño, implementación y pruebas.	12
6.1	Especificación de requisitos	12
6.2	Análisis del sistema	14
6.3	Diseño del sistema:	15
6.3.1	Diseño de la Base de Datos	15
6.3.2	Diseño de la Interfaz de usuario.	16
6.3.3	Diseño de la Aplicación.	18
6.4	Implementación:	18
6.4.1	Entorno de desarrollo.	18
6.4.2	Estructura del código.	18
6.4.3	Cuestiones de diseño e implementación reseñables.	18



6.5	Pruebas.....	20
7	Manuales de usuario	22
7.1	Manual de usuario	22
7.2	Manual de instalación.....	22
8	Conclusiones y posibles ampliaciones	22
9	Bibliografía	24
10	Anexos.....	24

1. Identificación proyecto

El proyecto es un trabajo de investigación consta de la realización de una aplicación web, que pueda reconocer imágenes de los distintos animales que existen en nuestra región (Castilla y León), mediante la creación de una o varias redes neuronales que, siendo entrenadas, pueda llegar a tener una fiabilidad a la hora de reconocer distintas imágenes medioambientales.

Este proyecto ha sido propuesto por una institución pública para su realización. Dado que se trata de un proyecto complejo, no se ha establecido un alcance mínimo definido, lo que justifica su enfoque como trabajo de investigación.

2. Organización de la memoria

Punto 1: descripción breve y superficial del proyecto

Punto 3: metodología del proyecto (uso) y su adaptación a una página web para el uso de los usuarios corrientes.

Punto 3.1: objetivos principales del proyecto.

Punto 3.2: descripción de la metodología de desarrollo del proyecto.

Punto 3.3: descripción y detalle de las tecnologías de desarrollo y herramientas utilizadas durante la realización del proyecto

Punto 4: justificación de la importancia de llevar a cabo este proyecto y de la necesidad real que existe en la sociedad actual de contar con una herramienta de este tipo.

Punto 4.1: se detallan los límites del sistema, sus funcionalidades básicas, los usuarios involucrados y los sistemas o componentes con los que interactúa, proporcionando una visión integral de cómo funciona y qué abarca el sistema desarrollado.

Punto 4.2: se resumen la metodología de desarrollo, las técnicas de inteligencia artificial utilizadas (por ejemplo, deep learning y redes neuronales convolucionales), y la arquitectura del sistema, destacando la utilización del framework Django para estructurar la aplicación web.

Punto 4.3: descripción del despliegue de la aplicación.

Punto 5: describe las etapas temporales del crecimiento, cambio en los requisitos, actividades que se han cumplido en cada etapa, duración, y observaciones importantes acerca de simultaneidad o condiciones especiales de trabajo.

Punto 6.1: se presentan estructurados los requisitos funcionales y técnicos que debe cumplir el sistema (requisitos de la página web, requisitos de los modelos de inteligencia artificial y requisitos relacionados con las herramientas y tecnologías con que se debe trabajar en el desarrollo).

Punto 6.2: se describe el funcionamiento general de la aplicación web, su estructura técnica basada en Django, y se enumeran claramente las funcionalidades disponibles para el usuario.

Punto 6.3.1: explica cómo está estructurada la base de datos, qué entidades principales la componen (User, Animal y Lugar), y se detallan las relaciones entre tablas mediante asociaciones OneToMany.

Punto 6.3.2: se describe cómo está implementada la interfaz usando Django, HTML5, CSS y JavaScript, organizada en distintas pantallas accesibles mediante un menú hamburguesa.

Punto 6.3.3: se explica el análisis y arquitectura que ha llevado a cabo la página web en el framework Django.

Punto 6.4.1: esta parte describe el entorno de desarrollo del proyecto, detallando las herramientas, lenguajes, frameworks, librerías, bases de datos, metodologías y entornos de programación

Punto 6.4.2: explicación del código tanto del desarrollo de IA como del desarrollo de la página web en Django.

Punto 6.4.3: descripción de las dificultades y retos encontrados durante el desarrollo del proyecto.

Punto 6.5: descripción y comprobación de alguna de las acciones a realizar por la página web.

Punto 7.1: manual de usuario página web.

Punto 7.2: manual de instalación de las herramientas para el desarrollo de la página web.

Punto 8: desarrollo y explicación de partes ampliadas en el proyecto para su puesta en el proyecto real, así como las partes a desarrollar para su posterior funcionamiento. También, se comentan las conclusiones sacadas tras el desarrollo del proyecto.

Punto 9: bibliografía de las páginas web o herramientas de apoyo durante el desarrollo del proyecto.

Punto 10: documentos anexados.

3. Descripción general del proyecto

El proyecto consta de un trabajo de investigación, como hemos comentado anteriormente, en el que mediante un dron y la incorporación de una cámara RGB y una cámara térmica, se puedan avistar y reconocer desde dicho dispositivo y con la realización de fotos con estas cámaras, los animales que existan en el terreno sobre el que sobrevuele el dron.

Las cámaras RGB y térmicas, realizarán fotos al mismo tiempo, obteniendo la misma imagen en ambas cámaras. Una vez realizadas estas imágenes, los modelos de Inteligencia Artificial pasarán a intentar reconocer dichas imágenes, con la ayuda de un modelo térmico que realizará un recorte en la imagen para que estos modelos centren su atención en el animal y no en otras zonas de la imagen que puedan llevar a confusión a estos modelos, dando un resultado erróneo del reconocimiento.

Para la presentación y uso de personas corrientes de esta IA, habilitaremos una página web, en el que incorporaremos dichos modelos, salvo el térmico, ya que estas personas no disponen de una cámara térmica para poder realizar este recorte en la imagen.

a. Objetivos

Los objetivos principales de la realización del proyecto son:

- Desarrollo de una aplicación web completa y funcional
- Implantación de nuevas tecnologías, como una IA que aprenda a reconocer imágenes.
- Aprendizaje de nuevos frameworks y herramientas para realizar distintos proyectos
- Diseño de una base de datos eficiente con la finalidad de almacenar información necesaria para el buen funcionamiento de la aplicación
- Aprender a trabajar y a desenvolverse en el mundo empresarial relacionado con la programación
- Implantación y uso de librerías externas
- Conseguir un modelo de IA que consiga reconocer de manera exitosa los distintos tipos de animales a reconocer

b. Cuestiones metodológicas

Para la realización de este proyecto he usado metodologías tradicionales. En este caso, el modelo en cascada que se distingue de las demás por desarrollar el software necesario por fases, en la que se necesita que cada fase se finalice antes de que se pase a la siguiente, consiguiendo una mejor planificación del proyecto.

He usado esta metodología, ya que la idea del proyecto estaba claro desde el momento en el que se empezó a realizar, por lo que no es necesario tener en cuenta la posibilidad de cambios además de que, al empezar pronto con el proyecto, no había problema en llegar a los plazos de entrega, por lo que, si alguna parte del proyecto se atrancaba, me centraba en acabar esa parte sin pensar en los plazos de entrega, que son en principio los problemas usuales a la hora de usar este tipo de metodología.

c. Entorno de trabajo (tecnologías de desarrollo y herramientas)

Las tecnologías utilizadas para la creación de mi proyecto han sido:

- Lenguaje de Programación: Python
- Frameworks: Django
- Librerías: TensorFlow y Keras para el desarrollo de la red neuronal
- Base de datos: MongoDB y SQLite
- Herramienta para la colaboración y almacenamiento del código Python: GitHub
- Metodología: Metodología de modelo en cascada
- IDE: Visual Studio Code
- Herramienta interactiva de notebooks: Jupyter

Los estudios previos realizados han sido todos los relacionados con el matching learning, como, por ejemplo:

- Qué tipo de matching learning es el más apropiado para el procesamiento de las imágenes.
- El código para programar de manera correcta una red neuronal.
- La existencia de datasets para la conformación del modelo de la red neuronal
- Los valores que se ajustan mejor a la red y a su modelo para un mejor entreno y funcionamiento de la IA a la hora de implantarla en la aplicación.
- Estudio sobre el uso de Jupyter para la escritura y ejecución de código en tiempo real.
- Importación, uso y funcionalidad de las librerías de tensorflow y keras.

- Uso e implantación del código Python en una aplicación realizada en Django.

4. Descripción general del proyecto (contexto y justificación)

Existen muchos estudios de la cantidad de fauna existente en nuestro medio ambiente y la necesidad de cuidarla, ya que, cada vez, complicamos más la existencia de esta en nuestros bosques y campos debido a la contaminación, actividades como la caza ilegal y masiva, y la construcción del hombre en el territorio animal, lo que provoca que invadamos su espacio, produciendo la muerte de cada vez más individuos, haciendo que el número y variedad de especies en nuestra región sea cada vez más pequeña.

El propósito de realizar el proyecto viene por la petición de un cliente de la empresa de la ejecución del mismo, con la finalidad de poder avistar de manera aérea la cantidad de individuos de las distintas especies que hay en nuestros bosques y campos, para tener una idea de que tenemos en nuestra región.

El conocimiento de las especies y cantidad de miembros de cada especie es beneficioso para el humano, con el fin de poder llevar a cabo planes de conservación de la biodiversidad (en caso de que la cantidad de fauna en la región sea pobre) o de prevención de una especie en concreto. Además, así, se podría estudiar un posible caso de integración de una nueva especie en nuestra fauna, para combatir ese mismo problema en otras regiones del país o del continente.

Este proyecto puede ser interesante para muchas instituciones dedicadas a este tema de conservación del medio ambiente, ya que este mismo proyecto llevado a cabo con drones, es mucho más económico para las empresas que la implantación de un satélite estadounidense para realizar esa misma función, por lo que, nuestro propósito, es realizar un proyecto más económico e igual de efectivo.

a. Visión general del sistema: límites del sistema, funcionalidades básicas, usuarios y/o otros sistemas con los que pueda interactuar.

El sistema desarrollado es parte de un proyecto de investigación cuyo objetivo principal es avistar e identificar animales utilizando un dron equipado con una cámara RGB y una cámara térmica. Un dron sobrevuela un área determinada, tomando fotos superpuestas con ambas cámaras, lo que genera dos imágenes de la misma área: una en el espectro visible (RGB) y otra en térmico.

El sistema se encarga de procesar tales imágenes con modelos de inteligencia artificial entrenados específicamente para el reconocimiento de animales, mientras que para aumentar la precisión, se utiliza un modelo térmico que recorta automáticamente la región de interés que contiene al animal, dirigiendo así la atención del modelo de reconocimiento y minimizando las posibilidades de error debido al desorden de fondo irrelevante. Los límites del sistema se centran en la captura de imágenes, el procesamiento con modelos de IA, así como en la visualización y gestión de resultados a través de una interfaz web. No incluye el control del dron, la navegación autónoma o el entrenamiento de modelos desde cero (parte de modelos preentrenados).

Las funcionalidades básicas del sistema son:

- Captura simultánea de imágenes RGB y térmicas mediante un dron.
- Procesamiento de las imágenes para el reconocimiento de animales.
- Recorte térmico automático para enfocar el área relevante.
- Presentación de resultados en una plataforma web accesible para el usuario.

Los usuarios del sistema incluyen:

- Investigadores, que utilizan el sistema completo incluyendo el análisis térmico.
- Usuarios comunes, que acceden a los modelos de reconocimiento a través de una página web, pero sin disponer del modelo térmico (al no contar con cámaras térmicas).

El sistema interactúa con:

- Cámaras montadas en el dron (RGB y térmica) para la adquisición de imágenes.
- Modelos de IA (tanto térmicos como de reconocimiento) que procesan las imágenes.
- Una base de datos o sistema de almacenamiento donde se guardan las imágenes y los resultados de los análisis.
- Una página web desde la cual los usuarios pueden subir imágenes RGB, obtener predicciones del modelo de reconocimiento y consultar resultados.

b. Descripción breve de métodos, técnicas o arquitecturas(m/t/a) utilizadas.

La metodología adaptada es una metodología incremental basada en ciclos de prueba y mejora continua, lo que nos permite integrar el modelo de inteligencia artificial con la interfaz web de forma

progresiva. Además, se emplearon prácticas propias del aprendizaje supervisado para el entrenamiento del modelo de clasificación de imágenes, en el que entrenamos los distintos modelos numerosas veces comprobando su eficacia y viendo si los cambios realizados para cada entrenamiento eran beneficiosos para los mismos.

Las técnicas utilizadas para el desarrollo del modelo de IA han sido *deep learning*, específicamente redes neuronales convolucionales (CNN), implementadas con las bibliotecas TensorFlow y Keras en el entorno Jupyter Notebook. También, se aplicaron técnicas de preprocesamiento de imágenes, normalización y aumento de datos (*data augmentation*) para mejorar el rendimiento de los modelos.

La arquitectura utilizada ha sido una basada en el framework Django, que nos permite separar el código en tres secciones: la lógica del programa (vistas), la forma en la que se muestra al usuario (HTML) y donde se almacena la información (base de datos). Esta separación de código nos permite tener una página web mucho más estructurada.

c. Despliegue de la aplicación indicando plataforma tecnológica, instalación de la aplicación y puesta en marcha

La página web no va a ser desplegada, ya que, al ser un proyecto privado de la empresa, junto con una entidad pública, este, ha de ser retocado y mejorado con la finalidad de realizar con una mayor afinidad dicho proyecto, como, por ejemplo, la implementación en el dron de la parte térmica.

La página web se podrá acceder en local durante la presentación de dicho proyecto, ejecutando los archivos configurados en Visual Studio Code, y entrando mediante la IP y puerto correspondiente en el navegador que deseemos.

5. Planificación y presupuesto

El proyecto ha constado de dos partes distintas durante los 3 meses que ha abarcado el mismo en el cómputo global. Estas dos fases diferenciadas se deben a un cambio completo de los requisitos del proyecto por parte de la entidad que lo demanda, cubriendo la primera fase un total de 1 mes, quedando el resto cubierta por la segunda fase (2 meses). Estos dos han sido:

1º FASE

TÍTULO	DESCRIPCIÓN	DURACIÓN	NOTAS
Propuesta y estudio del proyecto	Definición de los requisitos y estudio para la realización de redes convolucionales.	1 semana	
Desarrollo del modelo IA	Recogida de información y herramientas para realización y posterior entrenamiento de dicho modelo.	3 semanas	
Estudio del framework Django	Estudio sobre el uso y desarrollo de aplicaciones web en Django, tanto en la parte del back-end como la del front-end.	3 días	Se hace de manera simultánea con el desarrollo de modelo IA.

2º FASE

TÍTULO	DESCRIPCIÓN	DURACIÓN	NOTAS
Propuesta y estudio de proyecto	Definición de nuevos requisitos, así como la adaptación de parte del proyecto adaptado a esos nuevos requisitos.	1 semana	Se realizo de manera conjunta con el estudio del uso de Django, ya que el framework para realizar dicha página web no varía.
Desarrollo de los modelos IA y script térmico	Recogida de imágenes y realización de script en Jupyter para el entrenamiento de 4 modelos de IA, además de	2 meses	

	desarrollo del script para la cámara térmica.		
Desarrollo de la página web	Desarrollo de la página web en el framework Django y modelaje de la base de datos (SQLite).	5 semanas	Se realiza de manera conjunta con los desarrollos de los modelos IA y script térmico.

6. Documentación Técnica: análisis, diseño, implementación y pruebas.

a. Especificación de requisitos

La página web tiene una especie de requisitos a cumplir. Siendo estos:

- Respecto a la página web:
 - Desarrollo de una página web funcional y accesible para los usuarios.
 - Desarrollo de una página web llamativa y cómoda.
 - Implementación de una base de datos que almacene la información tanto de usuarios como de animales y lugares (ubicación).
 - Desarrollo de una página web donde los usuarios puedan registrarse.
 - Desarrollo de una página web donde los usuarios puedan iniciar sesión.
 - Desarrollo de una página web donde estén implementados unos modelos de IA.
 - Desarrollo de una página web donde los usuarios puedan reconocer animales ubicados en imágenes.
 - Desarrollo de una página web segura en el que no se permite la salida y entrada de datos no esperados en torno al usuario.
 - Implementación de un mapa en el que se pueda mostrar de manera más visual la ubicación de cada animal encontrado por el usuario.
 - Posibilidad de cambiar la contraseña.
 - Posibilidad de recuperar la contraseña en caso de que el usuario lo deseé.

- Desarrollo de una página web en el que se muestren los errores de manera sencilla y personalizada para cada tipo.
- Desarrollo de una página web que permita al usuario realizar imágenes al entorno desde la misma página.
- Desarrollo de una página web que permita subir archivos almacenados en el dispositivo del usuario.
- Desarrollo de una página web que pueda acceder a la cámara del dispositivo.
- Desarrollo de una página web que pueda acceder a la ubicación del dispositivo.
- Desarrollo de una página web que pueda acceder a la ubicación almacenada en una imagen (en caso de que la tenga).
- Implementación de un campo para filtrar los animales reconocidos por el usuario en el mapa.
- Diseño que se adapte a la pantalla de móvil y pc.
- Respecto a los modelos de Inteligencia Artificial:
 - Recogida de imágenes suficientes y adecuadas para el entrenamiento de los modelos de IA.
 - Uso de técnicas que ayuden a mejorar el entrenamiento de los modelos de Inteligencia Artificial.
 - Desarrollo de unos modelos con unos mínimos de precisión.
 - Desarrollo de un modelo que detecte los reptiles existentes en la comunidad de Castilla y León.
 - Desarrollo de un modelo que detecte los anfibios existentes en la comunidad de Castilla y León.
 - Desarrollo de un modelo que detecte los animales domésticos o animales de granja existentes en la comunidad de Castilla y León.
 - Desarrollo de un modelo que detecte las aves existentes en la comunidad de Castilla y León.
 - Desarrollo de un modelo que detecte los animales salvajes existentes en la comunidad de Castilla y León.
- Respecto a las herramientas a usar para el desarrollo de la página web y modelos de IA:
 - Realizar una página web con el framework Django.
 - Realizar una página web con el IDE Visual Studio Code.
 - Realizar modelos de Inteligencia Artificial en la herramienta Jupyter.

- Realizar modelos de Inteligencia Artificial en lenguaje de programación Python.
- Realizar una página web con lenguaje de programación Python.
- Almacenamiento de datos relacionados con la página web en bases de datos relacionales.

b. Análisis del sistema

Esta página web permite a los usuarios subir una imagen de un animal, procesarla mediante un modelo de inteligencia artificial entrenado previamente, y devolver el nombre del animal reconocido. Además, el sistema permite guardar ese resultado junto con una ubicación geográfica (manual o por defecto), y consultarlo posteriormente en un mapa.

El sistema está basado en el framework Django y estructurado en varias apps o módulos, cada uno con una responsabilidad clara.

El usuario puede realizar las siguientes acciones:

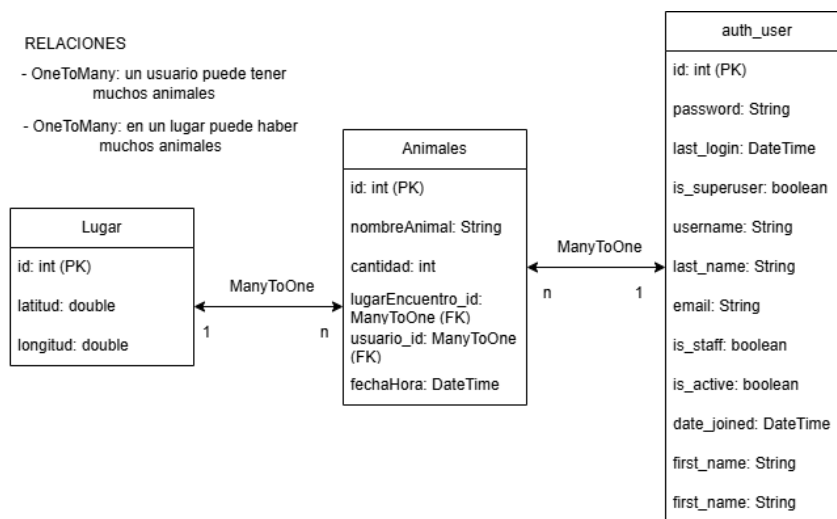
1. Registro del usuario.
2. Inicio de sesión del usuario.
3. Recuperación de la contraseña (en caso de pérdida o robo).
4. Subir una imagen a la página web.
5. Realización de una imagen en la página web.
6. Los modelos de IA procesan la imagen adjuntada.
7. El modelo de IA predice un animal.
8. Almacenamiento en la base de datos con posición geográfica adquirida de manera automática o manual (dada por el usuario).
9. El usuario puede acceder a consultar los animales almacenados junto con su posición geográfica en un mapa.
10. Filtración de animales en el mapa mediante inserción del nombre del mismo en un campo de texto.
11. Cierre de sesión de la cuenta del usuario.
12. Eliminación de la cuenta de usuario.
13. Cambio de contraseña desde dentro de la sesión.

c. Diseño del sistema:

i. Diseño de la Base de Datos

La base de datos está modelada de la siguiente manera:

- User (usuario)
- Animal
- Lugar



```

class Lugar(models.Model):
    latitud = models.FloatField()
    longitud = models.FloatField()
    #obligamos al programa a que la tabla en la base de datos se llame "Lugar"
    class Meta:
        db_table = 'Lugar'

#creamos la tabla Animales con sus atributos correspondientes
class Animales(models.Model):
    user = get_user_model()

    usuario = models.ForeignKey(user, on_delete=models.CASCADE, related_name='animales') # Relación many-to-one
    nombreAnimal = models.CharField(max_length=50)
    cantidad = models.IntegerField()
    lugarEncuentro = models.ForeignKey('Lugar', on_delete=models.CASCADE) # Relación con la tabla Lugar
    fechaHora = models.DateTimeField(auto_now_add=True)
    #obligamos al programa a que la tabla en la base de datos se llame "Animales"
    class Meta:
        db_table = 'Animales'

```

El framework django crea automáticamente la tabla Usuario (User), con los campos dictados anteriormente.

En la imagen mostramos el diagrama realizado, con las distintas relaciones que existen entre las distintas relaciones, siendo:

- OneToMany entre las tablas User y Animales (un usuario puede tener muchos animales, y un animal puede referirse a un único usuario).
- OneToMany entre las tablas Animales y Lugar (en un lugar puede haber muchos animales, pero un animal puede ser encontrado en un único lugar).

Además, se muestra el código Python desarrollado para la creación de la base de datos. Observamos que en el código no existe ninguna referencia a la creación de la tabla User, esto es porque el framework Django, crea automáticamente tablas usadas de forma general en todos los proyectos, como puede ser la tabla usuario. Las variables creadas automáticamente para esta tabla vienen referenciadas en el diagrama.

ii. Diseño de la Interfaz de usuario.

La interfaz del usuario ha sido implementada con el framework Django y Visual Studio Code, utilizando concretamente HTML5, CSS y JavaScript, que están integrados dentro de la carpeta templates de Django.

La estructura tomada ha sido una navegación por menús, en donde el usuario mediante un menú hamburguesa, puede acceder a las distintas partes de la página web. Esta, se estructura de la siguiente manera:

- Pantalla de inicio de sesión
 - Formulario para iniciar sesión
 - Botón de recuperación de contraseña (en esa misma página se despliega el formulario para recuperarla)
 - Botón para registrarte con una cuenta
 - Gestión de errores
- Pantalla de registro
 - Formulario para registrar un usuario
 - Gestión de errores
- Pantalla de inicio
 - Opción para abrir la cámara del dispositivo
 - Opción para realizar una foto con la cámara del dispositivo
 - Opción para adjuntar una imagen ya realizada
 - Opción para reconocer la imagen realizada o el archivo adjuntado
 - Gestión de errores

- Acceso al menú hamburguesa
- Pantalla de resultado del reconocimiento
 - Se muestra el resultado del reconocimiento
 - Opción para volver sin guardar el reconocimiento
 - Opción para almacenar el reconocimiento en la base de datos (en caso de no tener información de la ubicación del usuario o de la imagen adjuntada, se podrá incluir de manera manual)
 - Gestión de errores
- Pantalla de historial de reconocimientos
 - Se muestra un mapa con la posición geográfica de los animales reconocidos
 - Pinchando sobre cada punto nos muestra información del animal reconocido y la fecha y hora de cuando se reconoció.
 - Campo de texto para filtrar los animales del mapa por nombre
 - Gestión de errores
 - Acceso al menú hamburguesa
- Pantalla de ajustes
 - Saludo cordial personalizado
 - Información con el correo que ha iniciado sesión la cuenta
 - Opción de cambiar la contraseña (poniendo la contraseña actual y dos veces la nueva contraseña)
 - Opción para eliminar la cuenta de manera definitiva
 - Gestión de errores
 - Acceso al menú hamburguesa
- Pantalla de recuperación de contraseña
 - Campo de texto para poner la nueva contraseña dos veces, por razones de seguridad
 - Gestión de errores
- Menú hamburguesa
 - Acceso a la pantalla de inicio
 - Acceso a la pantalla de historial de reconocimientos
 - Acceso a la pantalla de ajustes
 - Acceso a un cierre de sesión si el usuario lo desea

iii. Diseño de la Aplicación.

El diseño de la aplicación, análisis y arquitectura del código viene explicado de manera profunda en el siguiente anexo:

[ANEXO 1: ANÁLISIS Y ARQUITECTURA DJANGO WILDDTECT](#)

d. Implementación:

i. Entorno de desarrollo.

Las herramientas utilizadas durante el proyecto han sido:

- Lenguaje de Programación: Python
- Frameworks: Django
- Librerías: TensorFlow y Keras para el desarrollo de la red neuronal
- Base de datos: SQLite
- Herramienta para la colaboración y almacenamiento del código Python: GitHub
- Metodología: Metodología de modelo en cascada
- IDE: Visual Studio Code
- Herramienta interactiva de notebooks: Jupyter

Además del uso de internet y de bastantes páginas web de internet y alguna inteligencia artificial para poder llevar a cabo y documentar de la manera más completa posible este proyecto.

ii. Estructura del código.

La estructura, explicación y análisis del código tanto de la página web como del desarrollo de los modelos de Inteligencia Artificial vienen configurados en:

[ANEXO 1: ANÁLISIS Y ARQUITECTURA DJANGO WILDDTECT](#)

[ANEXO 2: DESARROLLO DE MODELOS IA WILDDTECT](#)

iii. Cuestiones de diseño e implementación reseñables.

El proyecto ha tenido muchas dificultades, ya que se han usado tecnologías nuevas para todas las actividades realizadas.

Uno de los principales retos encontrados durante el desarrollo del proyecto estuvo relacionado con la creación y entrenamiento de los modelos de inteligencia artificial para el reconocimiento de imágenes. La herramienta utilizada, Jupyter, era completamente nueva para mí, lo que supuso una curva de aprendizaje considerable al inicio del proceso. Familiarizarme con su entorno, su forma de trabajar por celdas y la integración con librerías específicas para el tratamiento de imágenes y el entrenamiento de modelos fue de lo que más ha costado en la realización de dicho proyecto.

Además, la propia implementación de un sistema de IA capaz de detectar y clasificar imágenes también fue un desafío técnico importante, ya que era la primera vez que abordaba este tipo de problemas. Esto implicó comprender el funcionamiento de redes neuronales convolucionales, el manejo de datasets, el preprocesamiento de imágenes, y la configuración adecuada de los parámetros de entrenamiento.

A ello se sumó el hecho de que cada entrenamiento de los modelos suponía un tiempo de espera bastante elevado, generalmente entre 5 y 8 horas por sesión, lo que dificultó la posibilidad de realizar muchas pruebas o ajustes finos. Esto obligó a planificar cuidadosamente cada entrenamiento, prever errores antes de ejecutarlo, y optimizar el código lo máximo posible para evitar pérdidas de tiempo.

Otra de las principales dificultades durante el desarrollo del proyecto fue el uso del framework Django para la creación de la aplicación web. Era la primera vez que trabajaba con esta tecnología, y su forma de organización interna me resultó bastante compleja al principio. Django presenta una estructura de archivos muy particular, donde los distintos componentes del proyecto (modelos, vistas, plantillas, URLs, etc.) están separados en diferentes carpetas y ficheros, lo cual requiere entender bien su lógica.

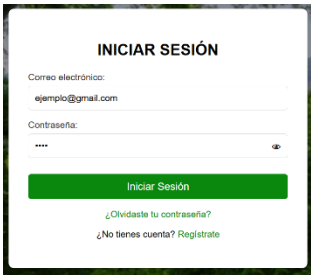
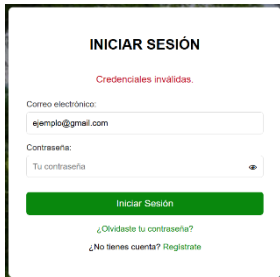

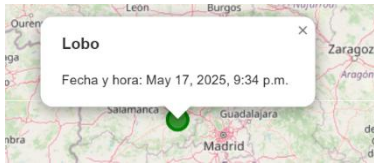
Al no estar familiarizado previamente con esta metodología de trabajo, me costó adaptarme a su sistema de mapeo entre rutas, controladores y plantillas, así como a la manera en que se gestionan los formularios, las sesiones de usuario o la comunicación con la base de datos mediante el ORM. Todo esto representó una barrera inicial que tuve que superar investigando y practicando mediante ejemplos y documentación oficial.

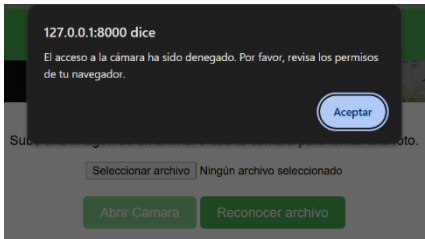

Además, una de las partes principales del proyecto fue combinar los modelos de inteligencia artificial entrenados anteriormente con la página web realizada con Django. Esto se convirtió en el desafío, porque había que sincronizar el comportamiento del modelo con el medio web. Se implementó una función para que el usuario pudiera subir una imagen, tratarla en el backend y

obtener la predicción del modelo, señalándolo después en la interfaz. Se trataron también detalles técnicos como la carga del modelo, el manejo de los datos de entrada y la gestión de errores.

Estos han sido algunos de las mayores que ha tenido el proyecto, entre otras muchas.

e. Pruebas.

PROPÓSITO	1ª IMAGEN	RESULTADO
A la hora de iniciar sesión, cuando rellenas el formulario con credenciales no registradas (ingresadas en la base de datos), salta un error personalizado al usuario (credenciales inválidas).		
Cuando guardas los cambios, al reconocer una imagen de un animal, estos datos se almacenan de manera correcta en la base de datos configurada.		

<p>A la hora de cerrar sesión, salta al usuario un formulario de si está seguro de cerrar la sesión, antes de enviarlo a la pantalla de inicio de sesión.</p>		
<p>Si el usuario no autoriza a la página web el acceso a la cámara, y el usuario pulsa el botón de “Abrir Cámara”, se mostrará un error al usuario.</p>		
<p>En la ventana de inicio de sesión, cuando el usuario introduce su contraseña y pulsa el ojo para visualizar con letras la contraseña introducida, el icono cambia y se muestra la contraseña de manera correcta.</p>		

7. Manuales de usuario

a. Manual de usuario

El manual de usuario de la página web viene explicado y analizado en el anexo 3:

[ANEXO 3: MANUAL DE USUARIO WILDDTECT](#)

b. Manual de instalación

El manual de instalación de las herramientas utilizadas para el desarrollo de la página web y de los modelos de Inteligencia Artificial viene explicado y analizado en el anexo 4:

[ANEXO 4: MANUAL DE INSTALACIÓN WILDDTECT](#)

8. Conclusiones y posibles ampliaciones

Tras el desarrollo de este proyecto he sacado muchas conclusiones, más beneficiosas que perjudiciales para mí, entre ellas:

1. Aprendizaje técnico y de nuevos conocimientos

A lo largo del proyecto he adquirido nuevos conocimientos en el uso de herramientas como Jupyter Notebook y Django, que eran totalmente nuevas para mí. Esto me ha permitido entender mejor cómo se entrenan e implementan modelos de inteligencia artificial y cómo se integran en una aplicación web funcional.

2. Superación de retos

Aunque surgieron diversas dificultades, como el tiempo de entrenamiento de los modelos o la complejidad de trabajar con Django por primera vez, pude superarlas mediante investigación, prueba y error, y una mejor planificación de las tareas.

3. Integración completa del sistema

Uno de los logros más destacables fue conseguir que el modelo de IA se integrara correctamente en la web, permitiendo al usuario interactuar fácilmente con el sistema y obtener predicciones de forma práctica.

4. Mejora en habilidades generales:

Este proyecto me ha permitido mejorar mis habilidades de programación, manejo de entornos de desarrollo, trabajo con bases de datos y diseño de interfaces web orientadas al usuario.

5. Aplicabilidad del proyecto:

El resultado final tiene potencial para aplicarse en contextos reales donde se requiera el reconocimiento de imágenes, y podría mejorarse aún más añadiendo nuevas funcionalidades o modelos.

En resumen, este proyecto me ha ayudado a aprender muchísimo, tanto en lo personal como en lo profesional. He tenido que enfrentarme a herramientas y conceptos que no conocía, como Django o los modelos de inteligencia artificial, y aunque al principio fue complicado, poco a poco fui entendiendo cómo funcionaban y cómo usarlos. También he mejorado mi forma de organizarme, de buscar soluciones por mi cuenta y de no rendirme ante los errores. Ha sido un trabajo intenso, pero muy útil, y me llevo muchos conocimientos nuevos y la satisfacción de haberlo sacado adelante.

Como ampliaciones reales del proyecto, se prevé implementar los modelos de inteligencia artificial en un dron para su uso práctico en un caso real que se destine a una institución pública. Siendo esta la integración que permitirá al sistema trabajar en ambientes reales, haciendo detecciones desde el aire de manera automatizada. Además, se pretende aprovechar la evolución del script térmico (se proporciona en el Anexo 2) para la introducción de la funcionalidad de análisis de imagen térmica, haciendo que los modelos puedan dar su respuesta y predicción con mucha más fiabilidad, centrando el uso de estos en la parte de la imagen donde se encuentre concretamente el animal.

Además, se podrá plantear con la empresa la posibilidad del lanzamiento de la página web para su uso, ya que me parece una página web interesante, accesible y cómoda para los usuarios.

9. Bibliografía

Estudio realización IA

<https://learn.microsoft.com/es-es/azure/machine-learning/tutorial-auto-train-image-models?view=azureml-api-2&tabs=cli>

[Detección e Identificación de animales mediante cámaras infrarrojas](#)

[Observación de animales con cámara termográfica | ¿Cual camara?](#)

<https://platzi.com/tutoriales/6935-ia-data-ml/26097-como-crear-una-inteligencia-artificial/>

<https://www.ibm.com/es-es/think/topics/data-augmentation>

https://www.tensorflow.org/tutorials/images/data_augmentation?hl=es-419

Recogida imágenes IA

<https://www.kaggle.com/>

<https://images.cv/>

<https://universe.roboflow.com/>

Estudio desarrollo página web Django

https://developer.mozilla.org/es/docs/Learn_web_development/Extensions/Server-side/Django

<https://docs.python.org/3/>

<https://docs.djangoproject.com/en/5.2/>

<https://django-us.com/es/Blog/building-ai-powered-applications-with-django/>

<https://django-us.com/es/Blog/building-ai-powered-applications-with-django/>

10. Anexos

[ANEXO 1: ANÁLISIS Y ARQUITECTURA DJANGO WILDDetect](#)

[ANEXO 2: DESARROLLO DE MODELOS IA WILDDTECT](#)

[ANEXO 3: MANUAL DE USUARIO WILDDTECT](#)

[ANEXO 4: MANUAL DE INSTALACIÓN WILDDTECT](#)