



**LOS LIBERTADORES**  
FUNDACIÓN UNIVERSITARIA

**SISTEMA INTELIGENTE RECONOCEDOR DE LA HUELLA DIGITAL DEL  
PERRO, POR MEDIO DE REDES NEURONALES Y MACHINE LEARNING (DINO)**

**JOHAN SEBASTIAN CIPRIAN ANZOLA  
JUAN CAMILO CASTRO YARA**

**FUNDACIÓN UNIVERSITARIA LOS LIBERTADORES  
FACULTA DE INGENIERÍA  
PROGRAMA DE INGENIERIA DE SISTEMAS  
BOGOTÁ, D.C  
MAYO 2022**

**SISTEMA INTELIGENTE RECONOCEDOR DE LA HUELLA DIGITAL DEL  
PERRO, POR MEDIO DE REDES NEURONALES Y MACHINE LEARNING (DINO)**

**JOHAN SEBASTIAN CIPRIAN ANZOLA**

**JUAN CAMILO CASTRO YARA**

**PROYECTO DE GRADO PARA OPTAR POR EL TITULO DE INGENIERO DE  
SISTEMAS**

**DIRECTORA,**

**ING. LUCY NOHEMY MEDINA VELANDIA**

**FUNDACIÓN UNIVERSITARIA LOS LIBERTADORES**

**FACULTA DE INGENIERÍA**

**PROGRAMA DE INGENIERIA DE SISTEMAS**

**BOGOTÁ, D.C**

**MAYO 2022**

Nota de aceptación

---

---

---

---

---

Firma del presidente del  
jurado

---

Firma del jurado

---

Firma del jurado

Bogotá D.C., julio de 2022

## DEDICATORIA

Dedico con todo mi corazón este proyecto de grado a mi familia, pues sin ella no lo habría conseguido. Esto lo hago para demostrarles que los sueños si se cumplen y que las metas que se propongan con objetivos definidos se pueden lograr, que a pesar de las adversidades y dificultades que se presenten hay que levantar la mirada y continuar hasta culminar el camino que Dios nos trazó.

*Johan Sebastian Ciprian Anzola*

Este logro es principalmente dedicado a mis padres, quienes fueron los pilares fundamentales para lograr el sueño de ser ingeniero en sistemas, de tener una alta gama de conocimiento y lograr un proyecto de tanta calidad, a mi novia que estuvo apoyándome a estar atento y ser mejor cada día, mis abuelos, que han estado conmigo pidiéndole a Dios que todos los sueños que vienen se me hagan realidad, Gracias a Dios por permitir este acontecimiento tan importante en mi vida y que mi familia está conmigo acompañándome en este camino que se llama vida.

*Juan Camilo Castro Yara*

## AGRADECIMIENTOS

Este proyecto de grado debe ser reconocido como una labor en conjunto de varias personas, realizado con mi tutora de proyecto Lucy Nohemy Medina Velandia, quien es una persona excepcional, una parte de lo que soy como ingeniero y como persona se lo debo a sus enseñanzas y jaladas de oreja, a mi compañero Juan Camilo Castro Yara, quien fue uno de los pilares fundamentales en el desarrollo y culminación de este proyecto, apoyándome y estando al pendiente de hasta el más mínimo detalle para que este trabajo culminara con la mayor perfección. Mis más sinceros agradecimientos por todo el conocimiento, paciencia y trabajo en equipo que se realizó para producir todo lo que en este documento se presenta.

*Johan Sebastian Ciprian Anzola*

El proyecto ha sido la experiencia más gratificante que he tenido en mi vida, primero que nada, muchas gracias a mi tutora ingeniera Lucy Nohemy Medina Velandia, a sus enseñanzas, su conocimiento que me permitió avanzar en mi conocimiento y que por su exigencia y guía nos ayudó en la formación como ingenieros. Gracias incondicionales a mi compañero y amigo Johan Sebastian Ciprian Anzola quien me ha acompañado en este sueño de ser ingeniero, por hacerme parte de este proyecto tan maravilloso y enriquecedor, por su atención y apoyo en todo momento, gracias al trabajo en equipo es que se ha podido culminar este proyecto y lograr la meta propuesta.

*Juan Camilo Castro Yara*

## TABLA DE CONTENIDO

Resumen.....	ix
Abstract.....	ix
Introducción.....	11
1    Aspectos de la investigación.....	12
1.1    Planteamiento del problema.....	12
1.2    Formulación de la pregunta de investigación.....	12
2    Objetivos.....	13
2.1    Objetivo general.....	13
2.2    Objetivos específicos.....	13
3    Justificación.....	13
3.1    Aspecto diferenciador.....	15
3.2    Alcance.....	15
4    Marco Teórico.....	16
4.1    Visión artificial.....	16
4.1.1    Visión artificial.....	<b>¡Error! Marcador no definido.</b>
4.1.2    Funcionamiento del sistema.....	19
4.2    Detección y reconocimiento.....	21
4.2.1    Comparación entre detección y reconocimiento.....	21
4.2.2    Detección.....	22
4.2.3    Proceso de detección de imágenes y video.....	22
4.2.4    Reconocimiento.....	25
4.3    Procesamiento digital de imágenes.....	27
4.3.1    Restauración y reconstrucción.....	28
4.3.2    Reconocimiento de patrones.....	29
4.3.3    Interpretación física.....	29
4.3.4    Mejora de la imagen.....	29
4.3.5    La transformada de Fourier.....	30
4.4    Redes neuronales.....	31
4.4.1    ¿Qué es una red neuronal?.....	31
4.4.2    Redes neuronales.....	32
4.4.3    Ventajas de las redes neuronales.....	33
4.4.4    Funcionamiento de una red neuronal.....	36
4.4.5    Utilidad de redes neuronales en sistemas de reconocimiento facial.....	37
4.5    Deep learning vs Machine learning.....	38
4.5.1    Introducción al concepto de inteligencia artificial.....	38
4.5.2    Machine learning.....	40
4.5.3    Deep learning.....	50
4.6    Aplicaciones móviles.....	58
4.6.1    Introducción al concepto de aplicaciones móviles.....	58
4.6.2    ¿Cuál es el origen de las aplicaciones móviles?.....	58
4.6.3    Sistemas operativos móviles.....	58
4.6.4    Desarrollo nativo (iOS).....	59

4.6.5	Android .....	59
4.7	Lenguajes de programación .....	61
4.8	Plataformas.....	63
4.8.1	Plataformas nativas .....	64
4.8.2	Plataformas híbridas.....	65
4.9	Bases de datos .....	66
4.9.1	SQL o NoSQL.....	67
4.9.2	Bases de datos NOSQL.....	68
4.9.3	SQL.....	69
4.10	Algoritmos de redes neuronales .....	70
4.10.1	Numba.....	70
4.10.2	TensorFlow .....	70
4.10.3	Keras .....	71
4.10.4	OpenCv .....	71
4.10.5	Pytorch .....	73
5	Marco Conceptual.....	73
5.1	Huella digital del perro (La nariz).....	74
5.2	Silueta del perro .....	76
6	Marco legal .....	77
6.1	Protección de datos personales.....	77
6.2	Derechos de los animales .....	85
7	Diseño Metodológico.....	88
7.1	Tipo de investigación .....	88
7.2	Fases del Proyecto .....	88
7.3	Procedimientos, herramientas y técnicas para recolectar la información .....	89
8	Desarrollo del proyecto.....	90
8.1	Inicio.....	91
8.1.1	Visión del proyecto.....	91
8.1.2	Desarrollo de Épicas .....	92
8.1.3	Creación del backlog priorizado del producto .....	94
8.1.4	Planificación del lanzamiento .....	97
8.2	Planificación y estimación .....	98
8.2.1	Identificación y lista de tareas.....	98
8.2.2	Estimación de tareas .....	99
8.2.3	Crear entregables .....	100
8.2.4	Envío de entregables.....	100
8.3	Análisis de requerimientos .....	100
8.3.1	Requerimientos funcionales.....	101
8.3.2	Requerimientos no funcionales.....	102
8.4	Diagramas de casos de uso.....	103
8.5	Diagrama de clases.....	109
8.6	Diagramas de secuencia .....	111
8.7	Diagrama componentes .....	118
8.8	Interfaz de usuario.....	118
9	Desarrollo software.....	120
9.1	Sprint 1:.....	120

9.1.1	Frontend: Pantalla (interfaz) principal, registro e inicio de sesión. ....	120
9.1.2	Backend: Base de datos y API .NET .....	127
9.2	Sprint 2: .....	132
9.2.1	Frontend: Pantalla (interfaz) mascotas.....	132
9.2.2	Backend: API Python.....	133
9.3	Sprint 3 .....	134
9.3.1	Frontend: Pantalla (interfaz) perfil. (3) y Pantalla (interfaz) cámara.....	134
10	Pruebas del proyecto .....	135
10.1	Set de pruebas .....	135
10.2	Evidencias set de pruebas .....	136
10.2.1	Punto de verificación 1 – registro usuario .....	136
10.2.2	Punto de verificación 2 – autenticación usuarios registrados .....	139
10.2.3	Punto de verificación 3 – registro perro.....	143
10.2.4	Punto de verificación 4 – reconocimiento huella digital perro .....	147
10.2.5	Punto de verificación 5 – reconocimiento silueta perro.....	149
10.2.6	Punto de verificación 6 – actualización datos usuario .....	155
10.2.7	Punto de verificación 7 – actualización datos perro .....	156
11	Análisis y Resultados .....	157
11.1	Reconocimiento de la silueta .....	158
11.1.1	Preparación de los datos.....	158
11.1.2	Creación del set de validación .....	162
11.1.3	Construcción del modelo .....	164
11.1.4	Entrenamiento .....	165
11.1.5	Validación .....	169
11.2	Reconocimiento huella digital .....	172
11.2.1	Preparación de los datos.....	172
11.2.2	Construcción del modelo .....	173
11.2.3	Entrenamiento .....	173
11.2.4	Validación .....	176
12	Conclusiones .....	178
13	Referencias.....	179



## **Resumen**

El presente trabajo consiste en desarrollar una aplicación móvil que cuente con un sistema para el reconocimiento de la huella dactilar de perros, utilizando inteligencia artificial. Para ello, se realizó un estudio en profundidad de las características básicas que hacen efectivo un sistema de este tipo. Además, para hacerlo innovador se buscaba que, además de contar con la identificación de la huella dactilar del perro, fuera capaz de reconocerlo a partir de la silueta. Una vez entendidos los fundamentos que hacen posible el reconocimiento, se decidió ir un poco más allá para elegir el modelo más adecuado.

Esta aplicación proporciona un sistema de identificación de perros que brinda a los usuarios una herramienta accesible y confiable en caso de un percance, brindando tranquilidad y seguridad tanto al usuario como a sus mascotas.

## **Abstract**

The present work consists of developing a mobile application that has a system for the recognition of the fingerprint of dogs, using artificial intelligence. For this purpose, an in-depth study of the basic characteristics that make a system of this type effective was carried out. In addition, in order to make it innovative, it was sought that, in addition to having the identification of the dog's fingerprint, it would be able to recognize them based on the silhouette. Once the fundamentals that make recognition possible were understood, it was decided to go a little further to choose the most suitable model.

This application provides a dog identification system which gives users an accessible and reliable tool in the event of a mishap, offering peace of mind and security to both the user and their pets.

## Introducción

En la actualidad existen numerosos tipos de sistemas, aplicaciones de seguridad y control que brindan la posibilidad a las personas de sentirse más seguras. Lo anterior, está enfocado como su nombre lo indica, a las “personas” pero ¿qué pasa con las mascotas?, este es precisamente el objetivo que el presente proyecto persigue, utilizar y aprovechar la tecnología disponible para beneficiar a las familias y personas en general cuando se ha extraviado la mascota.

La tecnología crece de una manera exorbitante y día tras día se desarrolla y mejora una gran variedad de software, esto permite que se pueda colocar a la disposición y utilizarlo de una manera eficaz para beneficio propio y de la comunidad. Lo anterior hace referencia a uno de los campos que más está creciendo en los últimos tiempos, la inteligencia artificial y los dispositivos móviles.

En el presente proyecto se busca que por medio de una aplicación móvil y un sistema de reconocimiento facial y de silueta, reconozca un canino que posiblemente se hubiere extraviado, además, dar la posibilidad de registrar la información personal del dueño y de la mascota, para que de esta forma se tenga un registro y pueda se llevar un control.

Se ha decidido hacer uso de la tecnología, en especial de las Redes neuronales, el *Machine Learning* y algunos algoritmos inteligentes para beneficiar a las personas y proporcionar una solución a problemas habituales como la pérdida de su mascota, a la vez, que tengan control sobre sus animales y generar una mayor sensación de seguridad y confort a su familia.

## **1 Aspectos de la investigación**

### **Planteamiento del problema**

Hoy en día, adquirir una mascota es un proceso sencillo, existen hogares u organizaciones legales que brindan el servicio de resguardo para animales y ofrecen la posibilidad de la adopción, o lugares donde se pueden comprar animales domésticos, ya sea para regalo o para al mismo comprador. Esto brinda a los perros una nueva oportunidad de ser acogidos en un hogar lleno de cariño y amor.

Debido a lo anterior, se han incrementado los niveles de adquisición de perros. Las mascotas se han convertido en una parte fundamental del núcleo familiar y con ello, el aumento en la participación de la familia al incluirlas, esto ha propiciado el aumento en los índices de eventos adversos que a su vez traen consigo amenazas como las pérdidas, accidentes, robos y las apropiaciones indebidas de las mascotas por individuos ajenos a la familia.

Lo anterior, es un problema que afecta directamente a los propietarios y por supuesto a sus mascotas; la familia sufre cuando su mascota se pierde. Precisamente, este aspecto es el que se busca minimizar a través de una solución de software que frene directamente esta problemática para proporcionarle tranquilidad a cada uno de los propietarios con sus mascotas.

### **Formulación de la pregunta de investigación**

¿Cómo reconocer la silueta de un perro y su huella digital (la nariz), por medio de un sistema inteligente que utiliza Machine learning y Redes neuronales?

## 2 Objetivos

### Objetivo general

Desarrollar un sistema inteligente, para que por medio de *Machine learning* y Redes neuronales se reconozca la huella digital y la silueta de un perro.

### Objetivos específicos

- Analizar cuál red neuronal y algoritmo de *Machine learning* son los más convenientes para el reconocimiento de la huella digital (nariz) del perro.
- Elaborar un módulo para que la aplicación reconozca la raza del perro por su silueta, en razas determinadas.
- Construir un módulo para que la aplicación reconozca el perro por su huella digital (la nariz).
- Diseñar un algoritmo inteligente que integre los algoritmos escogidos de *Machine learning* y la red neuronal, para reconocer la huella digital y la silueta del perro.
- Implementar una instancia en una base de base de datos para registrar los datos de cada perro y su dueño.

## 3 Justificación

La idea surge de la problemática que se presenta con las mascotas como son: robos, pérdidas, abandonos, accidentes, etc.; la familia y el animal sufre cuando estos sucesos ocurren. Por lo anterior, cada uno de estos factores influye tanto en la vida de las personas que incluyen estos animales en su núcleo familiar, puesto que, en el seno del hogar, los animalitos son considerados como lo que son, seres vivientes, que sienten, sufren, aman y también tienen derechos (según la Ley 1774 del 2016 del congreso de Colombia). Esto conlleva a conceptualizar la solución con el

desarrollo de una aplicación móvil y el uso de las últimas tecnologías para que se mitigue en gran parte toda esta situación.

Así como los seres humanos tienen rasgos y características propias que los identifican, tienen también las huellas dactilares, consideradas como los patrones de las yemas de los dedos, de las palmas de las manos o de las plantas de los pies. Es así como los perros también cuentan con una huella, de las que muchos pensarán que está ubicada en los dedos de sus patas, efectivamente, no es así, sino que los orificios nasales y las líneas que los conforman no son iguales a las de otro perro, a través de la nariz se conforma la huella nasal que ayuda a identificar a los amigos de cuatro patas.

Otra característica adicional que se puede usar para reconocer un perro es su silueta, ya que a partir de ella se pueden sacar datos como: peso promedio, altura, color, tipo de pelaje, etc. Por ende, este trabajo se basó en esas dos características para realizar el proceso de reconocimiento, los cuales son considerados suficientes para desarrollar con éxito el proyecto.

En consecuencia, debido a la constante pérdida de las mascotas de las familias, al secuestro, o cualquier otro inconveniente que se presente con estos, es que se ha pensado en la solución aquí presentada como es el reconocimiento del amigo de cuatro patas por medio de la silueta y/o de su huella digital, con el fin de dar mayor seguridad y solventar en algo la tristeza de los dueños de la mascota.

### **Aspecto diferenciador**

DiNo es la única aplicación colombiana que le permite al usuario registrar la huella digital de un perro, además no necesita tener al perro físicamente para realizar el registro.

DiNo permite el reconocimiento de la huella digital (la nariz) por medio de múltiples fotografías.

DiNo cuenta con un sistema de reconocimiento para identificar la raza del perro basándose en la silueta.

### **Alcance**

La aplicación DiNo está enfocada principalmente en el registro de perros, con el objetivo de poder rastrearlos en caso de un posible extravío. Esto se realizará por medio de un registro de identificación canina, que está dada por la raza del animal y la huella dactilar ubicada en la nariz.

Las razas del perro serán reconocidas por medio de algoritmos de reconocimiento de siluetas, ya que los perros tienen diferencias muy notables, por otro lado, la huella digital será reconocida por medio de redes neuronales asociadas al *Machine learning*, esta huella digital (la nariz) es la base primordial de la aplicación, debido a que, a la hora de encontrar un perro será el punto que lo diferenciara de otros perros que sean de la misma raza o simplemente, que se parezcan.

## **4 Marco Teórico**

### **Visión artificial**

La visión artificial se refiere a un sistema de adquisición de imágenes a través de sistemas ópticos conectados a un computador, mediante el cual se es capaz de analizar y extraer información. El objetivo principal de este sistema es que funcione como la visión humana, es decir, los ojos captan una imagen que es llevada hasta el cerebro donde se interpreta y se obtiene información de ella.

Estos sistemas son flexibles basados en un software de procesado rápido que puede detectar objetos en movimiento, así como su color y forma, midiendo la luz que se refleja en este, con el fin de automatizar tareas, realizar controles a determinadas personas u objetos, reducir el tiempo de ciclo de procesos industriales, entre otras.

En el año 1826 el químico francés Niepce fue capaz de realizar la primera fotografía en su laboratorio. Esto supuso el punto de partida de la obsesión que surgió en el ser humano por entender el funcionamiento del sistema óptico y como el cerebro es capaz de extraer de información de estas imágenes.

Pasaron muchos años hasta que los científicos comenzaron a introducirse en el campo del modelado de la percepción. Los pioneros en realizar investigaciones en este campo fueron Kepler, sobre en análisis geométrico de la formación de la imagen en el ojo, Newton, sobre visión a color y Helmholtz sobre la óptica fisiológica. (Universidad de Oviedo, s.f.)

Pero esto no supuso un comienzo real en lo que ha modelado, digitalización e interpretación de imágenes se refiere. EL hito más importante se produjo en la primera mitad del siglo XX con la



creación de la escuela de la psicología de la Gestalt, cuyo fin era explicar las propiedades de los mecanismos psicofísicos.

Esta escuela no triunfó, pronto desapareció y no fue hasta los años 50 cuando se volvió a retomar el estudio en el que hubo un cambio de rumbo en la temática de la investigación. En aquella época los científicos trataron de explicar el funcionamiento de la visión recurriendo a explicaciones al más bajo nivel celular (potencialidades de las neuronas), entre los que se pueden destacar a personajes como Hubel, Weisel y Barlow, que consiguieron dar respuesta a cómo nuestro mecanismo codifica, representa y reconoce las diferentes informaciones espaciales. (Universidad de Oviedo, s.f.)

Años más tarde, se trató de enfocar el problema a través de dos teorías diferentes. La primera, cuyo máximo exponente fue Azriel Rosenfeld, buscaba técnicas empíricas basadas en modelos matemáticos para la interpretación de imágenes. El segundo enfoque, se basaba en el estudio de bloques blancos iluminados sobre un fondo negro. Mediante esta experiencia Waltz (Científico informático) y Mackworth (profesor de Ciencias de la Computación y Cátedra de Investigación de Canadá en Inteligencia Artificial) consiguieron realizar interpretaciones de dibujos lineales a partir de imágenes.

Lo anterior, no resultó suficiente, por lo que en años posteriores se trató de elaborar un modelo más completo. Los primeros pasos los dio Horn (científico estadounidense que trabaja en el campo de la inteligencia artificial y la visión por computadora) quien, mediante el uso de ecuaciones diferenciales, puede expresar el proceso de formación de imágenes. Posteriormente, investigadores como Julesz (neurocientífico visual y psicóloga experimental en los campos de la percepción visual y auditiva) consiguieron elaborar una teoría mediante la cual usando solo los parámetros de

profundidad, distancia y textura pudo explicar el mecanismo de formación de imágenes 3D en el ojo humano. (Universidad de Oviedo, s.f.)

En los últimos años del siglo XX se han realizado innumerables trabajos en lo que al campo de la visión artificial se refiere. Instituciones como el MIT (Massachusetts Institute of Technology), bajo la dirección de David Marr, crearon la metodología modular tipo “Bottom Up”, que consistía en el procesamiento de una imagen recibida. Este proceso se podía dividir en tres grandes bloques:

- Realización de un esbozo primitivo de la imagen con el fin de caracterizar los rasgos básicos (nivel 2D)
- Construcción de superficies desde el punto de vista del observador, gracias a la información que proporciona el esbozo primitivo y la información tridimensional proporcionada por la estereoscopía.
- Por último, construcción del módulo y comparación con los datos almacenados (nivel 3D)

Este estudio realizado por Marr (neurocientífico y fisiólogo, integró los resultados de la psicología, la inteligencia artificial y la neurofisiología en nuevos modelos de procesamiento visual) supuso las bases en lo que a representación y análisis de imágenes digitales se refiere, ya que consiguió descomponer la información de los niveles 2D y 3D junto a sus diferentes esquemas de representación.

Además de lo mencionado, con la llegada de los computadores en los años 50-60 surge la necesidad de disponer de técnicas de transmisión y procesamiento de imágenes por satélite, junto con el análisis de imágenes obtenidas por rayos X. (Universidad de Oviedo, s.f.)

En otras partes del mundo como en Europa no se quedaron atrás en estos campos y decidieron investigar al mismo tiempo que lo hacia Estados Unidos. En los años setenta algunos profesores de la “*Ecole National Supérieure des Mines*” consiguieron distinguir diferentes tipos de rocas mediante la visión artificial gracias a la caracterización y medición de formas junto con un estudio de cálculo de probabilidades.

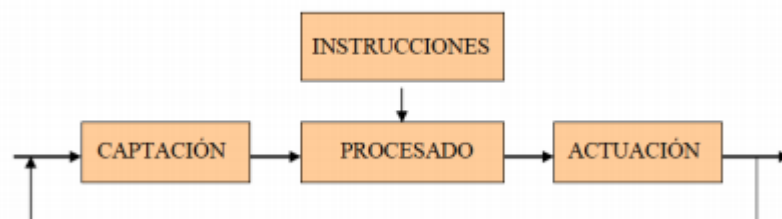
Finalmente, debido a que la visión artificial se usa en infinidad de campos, y para su funcionamiento se pueden utilizar multitud de técnicas diferentes, se conoce por visión artificial como el conjunto de técnicas encaminadas a la captura, codificación y representación de imágenes con el fin de que posteriormente sean analizadas por un ordenador. (Universidad de Oviedo, s.f.)

#### 4.1.1 Funcionamiento del sistema

El proceso de funcionamiento de un sistema basado en visión artificial ([Imagen 1](#)) se puede dividir en tres pasos bien diferenciados: captación de imágenes, procesado y actuación. Dentro del procesado es muy importante resaltar que esta etapa se realiza en concordancia con unas instrucciones previamente programadas por los operadores. (educación, 2012)

*Imagen 1*

*Esquema de pasos seguidos por un sistema de visión artificial*



*Nota.* Imagen tomada de sistema de visión artificial, (educación, 2012).

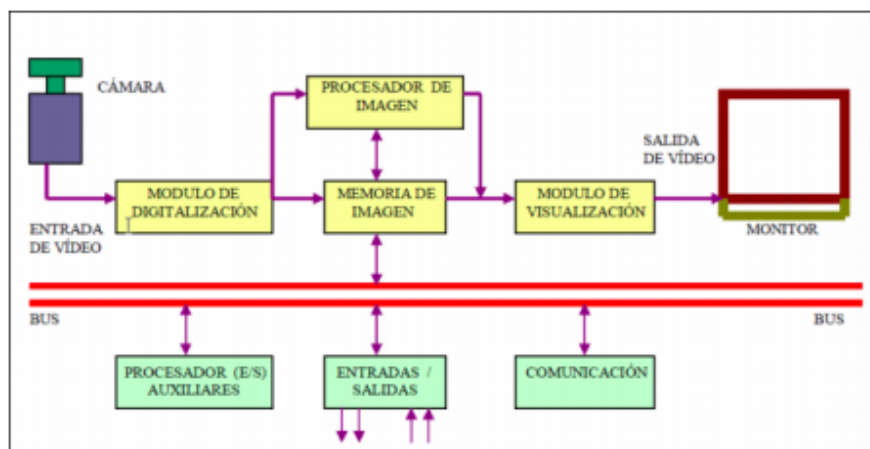
El sistema en su conjunto se compone de seis bloques ([Imagen 2](#)) que permiten que se pueda desarrollar el proceso. El primer paso que sigue el sistema consiste en que una cámara capte que tiene un objeto delante. A continuación, entra en juego lo que se conoce como módulo de digitalización, que básicamente es el encargado de transformar la señal analógica recibida por la cámara en una imagen digital.

En el siguiente paso, se producen dos operaciones a la vez: el procesador de imágenes (que interpreta las imágenes captadas por la cámara), y la memoria de la imagen (que se encarga de almacenar la señal que procede del módulo de digitalización). Por último, el módulo de visualización se encarga de transformar la señal almacenada en memoria de la imagen para que se puedan mostrar los resultados en un monitor. (educación, 2012)

No se puede terminar de hablar de este tipo de sistemas sin explicar en qué consiste el módulo de entradas y salidas, cuya función se basa en proporcionar el sincronismo entre la entrada de imágenes y salidas de control que actúan sobre controladores externos en función del resultado obtenido a la hora de interpretar la imagen.

## Imagen 2

Diagrama de bloques de un sistema de visión artificial



*Nota.* Imagen tomada de sistema de visión artificial, (educación, 2012)

## Detección y reconocimiento

### 4.1.2 Comparación entre detección y reconocimiento

Cuando se habla de aplicaciones que implique el reconocimiento de imágenes, hay términos que se usan indistintamente y que muchas veces se confunden entre ellos. Estos son detección y reconocimiento. Además, a estos dos en algunas ocasiones se le agrega el termino de rastreo; especialmente, cuando se habla de algoritmos de reconocimiento.

A principios de los años 80, los científicos *Sagi* y *Julesz*, fueron los primeros que afirmaron que el proceso de detección e identificación son dos procesos diferentes que ocurren de forma paralela. El problema que se presenta es establecer el final de la detección y el comienzo del reconocimiento, ya que ambos procesos poseen una relación muy amena entre ellos y en la mayoría de las situaciones, si falla uno también falla el otro. (Straube & Fahle, 2011)

Es muy importante, en el campo de la visión artificial, hablar de las diferencias de estos términos ya que, aunque a priori se puede estar hablando de cosas similares, en los algoritmos asociados a cada proceso se encuentran numerosas diferencias entre cada uno de ellos.

La idea reside en que, cuando se habla de detección de objetos, se hace referencia a que la cámara detecta que un objeto se ha posicionado delante de ella y a que se ha descubierto algo. Por el contrario, si se habla de reconocimiento de objetos hace referencia a que la cámara ha comprendido el objeto, sabe que es lo que tiene delante.

Parece que la idea está clara, pero a la hora de hablar de algoritmos asociados a cada uno de los procesos mencionados surgen diversos problemas, los cuales se trataran de resolver a continuación. (Grevelink, 2017)

### **4.1.3 Detección**

El diccionario de la lengua española define detección como la “Acción y efecto de detectar” (Española, 2021). En este sentido, el objetivo de la detección se basa en descubrir un objeto, averiguar que el sistema tiene algo delante de él. En palabras técnicas, el sistema debería ser capaz de distinguir un pequeño grupo de píxeles de entre un gran número de píxeles, los cuales constituyen el fondo estático.

### **4.1.4 Proceso de detección de imágenes y video**

La principal diferencia que presentan las imágenes respecto al video es que son estáticas, por lo que no se puede basar la detección en el movimiento del objeto. El problema que se presenta reside en que en la mayoría de las imágenes aparecen una gran cantidad de objetos juntos; esta proximidad hace que sea muy complicado establecer los límites de cada uno de los objetos.

Este fenómeno se conoce como “detección de bordes” y es lo que ayuda a determinar que hay objetos presentes en una fotografía. Para ello, lo que se busca es variar la intensidad a lo largo de la fotografía, es decir. Se buscan cambios de color abruptos en la escala de grises. Esto no es útil solo para simples objetos que se encuentren separados a una distancia determinada, sino que también ayuda cuando se tienen dos objetos, los cuales se encuentran superpuestos (un perro sentado en una silla). (Grevelink, 2017)

### *Imagen 3*

*Fotografía que se le presenta al sistema*



*Nota.* Imagen tomada de transformación de imágenes para procesamiento, (educación, 2012)

#### *Imagen 4*

*Transformación de la imagen a blanco y negro*



*Nota.* Imagen tomada de transformación de imágenes para procesamiento, (educación, 2012)

#### *Imagen 5*

*Detección de los límites de la imagen*



*Nota.* Imagen tomada de transformación de imágenes para procesamiento, (educación, 2012)

En las imágenes anteriores se puede ver secuencialmente un ejemplo de este proceso: De entrada se le proporciona a el Sistema una imagen en formato normal ([Imagen 3](#)), luego para que



el proceso sea más sencillo se cambia el formato de la imagen a blanco y negro ([Imagen 4](#)), para que finalmente el sistema identifique los límites de los objetos presentes en ella ([Imagen 5](#)). (Grevelink, 2017)

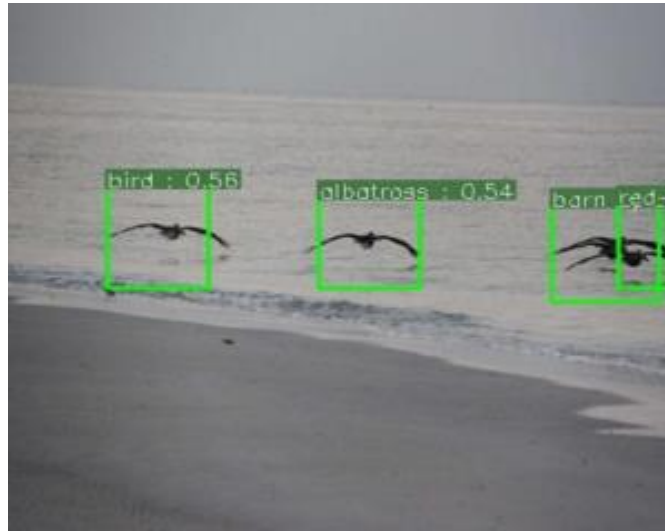
#### **4.1.5 Reconocimiento**

Para (educación, 2012), el reconocimiento es la capacidad de saber qué es algo en base a un conocimiento adquirido a través de la experiencia; por lo que, en el campo de reconocimiento de objetos, consiste en que un sistema tiene que ser capaz de saber qué tipo de objetos se encuentran en una determinada imagen.

Las posibles soluciones que se presentan ante este problema consisten en la creación de algoritmos que están capacitados para realizar integraciones entre imágenes que tiene en su base de datos: los algoritmos que aprenden de la experiencia o que son capaces de reconocer patrones. De esa forma, es significativo resaltar que estos algoritmos suelen indicar la probabilidad con la que reconocen un objeto; en la ([Imagen 6](#)) se presenta un ejemplo de un algoritmo de reconocimiento y se puede ver que con un 0,56 de probabilidad afirma que un objeto es un pájaro y para otro de ellos asegura con un 0,54 que se trata de un albatros. (Grevelink, 2017)

## Imagen 6

### Ejemplo de un algoritmo de reconocimiento de aves



*Nota.* Imagen tomada de ejemplo de reconocimiento de aves, (educación, 2012)

Se puede afirmar que los procesos de detección y reconocimiento se encuentran ampliamente diferenciados en la gran cantidad de sistemas de reconocimiento facial y cada uno de ellos desarrolla una función imprescindible para el proceso de identificación de un objeto: detección no es un sinónimo de reconocimiento. La detección se basa en diferenciar entre los objetos válidos y los que no; el reconocimiento consiste en saber qué tipo de objeto es el que hay en la imagen.

En este orden de ideas, este proceso no supone un proceso intercambiable; es decir, para que funcione de manera óptima se debe desarrollar en ese orden (detección y reconocimiento).

Para culminar, la separación de los procesos antes mencionados permite que se puedan combinar diferentes tipos de algoritmos con la finalidad de eliminar debilidades en el sistema. Es decir, si el sistema presenta debilidades en la fase de reconocimiento se puede utilizar otro

algoritmo de otro desarrollador para hacer que nuestro sistema sea más robusto y completo en esta fase.

### **Procesamiento digital de imágenes**

Hoy en día, los seres humanos están sujetos a la visión casi el 100% de su vida, de los cuales, están en constante aprendizaje, tanto los humanos, animales y en entornos tecnológicos sistemas autónomos con la capacidad de adquirir diferentes características de un entorno o de alguna u otra forma, una imagen. En un ambiente matemático, se podría expresar en un plano cartesiano, la imagen cuenta con un eje X y un eje Y, y el F que es el espacio donde está la imagen en sí, una imagen no cuenta solo con medidas comunes, ancho y largo, cuenta con los pixeles, todas las imágenes digitales cuentan con ellos. Un píxel es un orden de matrices o puntos de color en escala de grises, que ordenadas de una forma coherente muestran lo que es una imagen, es por ello que al ampliar una imagen o foto digitalmente, se pueden ver cuadros, es ahí donde se ven los pixeles, por ello entre más pixeles posea una imagen, más nítida y clara será.

En el entorno cotidiano los humanos están en constante aprendizaje, por ejemplo, como se llega a la capacidad de conocer que un carro tiene la forma como se conoce hoy en día, por medio de la visión, ya que, al observar una imagen de un carro, el cerebro humano procesa la información obtenida y llega al punto de lograr reconocer un carro en el futuro, y así es con cada cosa, objeto, material, etc.

El procesamiento digital se entiende como el proceso de analizar, obtener y adquirir una imagen al llegar al punto de aprender de ella, es por ello por lo que el sentido de la vista es fundamental y es el más avanzado de todos, es sorprendente todo el conocimiento que se puede adquirir por medio de ello. Pero ¿qué sucedería si en algunos casos, la visión humana estuviese limitada y no lograra detectar todo lo que una computadora pudiera observar?

Para responder a la pregunta anterior, puede decirse que es precisamente ahí donde entra el procesamiento digital de imágenes, pues la computación es parte fundamental de la vida cotidiana, y en muchas ocasiones es superior al entendimiento del ser humano; es así que el procesamiento digital de imágenes es entendido por Vilet (2004) como la capacidad de un sistema de reconocer patrones, señales, bordes, colores, píxeles, características y demás componentes que comprende una imagen, es esa disposición de encontrar lo que el ojo del ser humano no encuentra a simple vista, un ejemplo de ello son los códigos QR, donde se realiza el escaneo de una serie de combinaciones de módulos de blanco y negro que al ser procesados por la cámara de un celular los redirige a un sitio web, es esa habilidad de reconocer qué es esa imagen y como resultado es el proceso de obtener qué es y dónde es.

Una de las características fundamentales del PDI (Procesamiento Digital de Imágenes) es el uso de la computadora, la visión y la capacidad de aprender es inimaginable, tanto así que percibe cosas que el ojo humano no logra hacer; uno de estos elementos es el espectro electromagnético desde rayos gamma a las ondas de radio, respectivamente, que el ojo humano no distingue, pero que la visión computacional como área de la IA sí logra. Aparte de la computadora, está la imagen, el PDI utiliza innumerables métodos de la matemática aplicada, como son, el álgebra lineal, cálculo diferencial e integral, ecuaciones diferenciales, probabilidad y estadística (Vilet, 2004). Cada uno de ellos se aplican en los 3 problemas comunes del PDI como:

#### **4.1.6 Restauración y reconstrucción**

En pocas palabras, es un método matemático llamada deconvolución por mínimos cuadrados, utilizado para estimar una imagen con la original de la tal forma que la diferencia sea mínima por mínimos cuadrados, el algoritmo que realiza este procesos más fácil y rápida es la transformada de Fourier.

#### **4.1.7 Reconocimiento de patrones**

Es el resultante del proceso anteriormente realizado, la imagen se modifica de manera que se tenga una mejor lectura de la misma, se manipula la imagen a escala de grises para obtener dicho resultado.

#### **4.1.8 Interpretación física**

El principal objetivo es identificar las propiedades de la imagen ya que tienen un significado físico, primordialmente se pretende identificar términos de propiedades como lo son la radiación electromagnética de donde la imagen ha sido derivada para deducir la estructura y propiedades de los objetos capturados en la imagen.

#### **4.1.9 Mejora de la imagen**

El objetivo primordial de mejorar una imagen en específico es que al utilizarla o procesarla llegue al punto de que su objetivo final sea el más conveniente, ya que en muchos entornos no se encuentra una buena calidad ni resolución para cada fin, hay distintos métodos de mejoramiento de imágenes, pero, no quiere decir que uno solo se use con todas las imágenes, no se puede tratar una imagen tomada con una cámara de la misma forma de una radiografía, es por ello que se utilizan varias técnicas (Inmaculada Garcia, 1985):

1. Corrección de histogramas
2. Realce y extracción de bordes
3. Adelgazamiento y vectorización
4. Transformadas y filtrado

El mejoramiento de una imagen se divide en dos categorías, una de ellas es método del dominio espacial y métodos del dominio de la frecuencia. El método del dominio espacial trabaja

directamente sobre el plano de la imagen, lo que quiere decir, sus píxeles, sus bordes, etc. En cambio, el método de dominio de la frecuencia trabaja sobre la transformada de Fourier.

Hoy en día no se encuentra o no hay una teoría general, por esto es por lo que es un proceso altamente subjetivo en el caso de los humanos ya que básicamente se deja llevar por cuál de los métodos existentes es el mejor, cuando la imagen se procesa para ser percibida por una máquina es mucho más fácil.

#### **4.1.10 La transformada de Fourier**

Es un conjunto de técnicas, que hacen que desde una imagen origen se obtenga una imagen diferente, es allí donde entra este método, los principales objetivos según (C.Pinilla, 1997) son los siguientes:

- Suavizar la imagen: reducir la variación de píxeles vecinos
- Eliminar ruido: eliminar los píxeles con mayor intensidad que los demás
- Realzar bordes: Realzar los bordes de una imagen
- Detectar bordes: Identificar un cambio brusco de píxeles en la imagen.

El filtrado de frecuencia se define como la suma de dos componentes con dos diferentes escalas espaciales, la imagen con las bajas y altas frecuencias es por ello por lo que se utiliza el teorema de Fourier que corresponde la convolución correspondiente.

El teorema de Fourier se define en el uso de dos frecuencias de una imagen, frecuencias bajas y altas, una frecuencia baja es aquella que disminuye la frecuencia alta para suavizar la imagen, y una frecuencia alta se define por la atenuación de las frecuencias bajas, es más que todo utilizado para detectar bordes en la imagen por medio de los píxeles, donde se presentan cambios bruscos en la misma. (Vilet, 2004)

## **Redes neuronales**

### **4.1.11 ¿Qué es una red neuronal?**

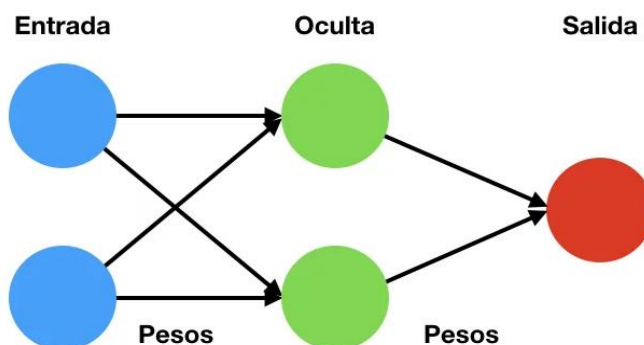
Desde la creación del primer computador en el año de 1946 el ser humano ha perseguido el deseo de emular maquinas que sean capaces de realizar por nosotros tareas, cálculos largos y tediosos. Una red neuronal se puede definir como otro elemento más de ayuda para el ser humano, cuya función principal se basa en la capacidad de memorizar y asociar hechos. Para ello los diferentes retos que se le presentan a la red los irá resolviendo, basándose en la experiencia previa adquirida. (Cabello Pardos, 2004)

Uno de los fines u objetivos de una red neuronal consiste en crear una máquina que sea capaz de emular al cerebro humano, por lo que muchos autores definen a la red neuronal como: “un nuevo sistema para el tratamiento de información, cuya unidad básica de procesamiento está inspirada en la célula fundamental del sistema nervioso humano: la neurona.” (Cabello Pardos, 2004)

Una definición un poco más técnica de lo que es una red neuronal ([Imagen 7](#)) sería: conjunto de unidades de procesamiento que intercambian información entre ellas para reconocer patrones, imágenes, manuscritos y secuencias de tiempo, al tiempo que con capaces de aprender y mejorar su funcionamiento. Para conseguir esto se realiza un modelo matemático de elementos procesales organizado en niveles que es capaz de dar diferentes respuestas en función de determinadas entradas externas (inputs). (Ruiz & Basualdo, 2001)

## Imagen 7

### Ejemplo de una red neuronal



*Nota.* Imagen tomada de [www.google.com](http://www.google.com)

#### 4.1.12 Redes neuronales

El concepto de red neuronal comenzó a gestarse en el año de 1936 cuando Alan Turing realizó una búsqueda entre el cerebro y el mundo de la programación. En cambio, los primeros en conseguir elaborar una teoría que relacionaba el cerebro humano con la red neuronal fueron los científicos Warren McCulloch y Walter Pitts que consiguieron realizar una red neuronal a través de circuitos eléctricos.

Posteriormente, en el año 1949, Donal Hebb consiguió enunciar el funcionamiento del aprendizaje en el ser humano, algo que se considera como uno de los pilares de la Teoría de las Redes Neuronales. Por otro lado, el importante reseñar el congreso que tuvo lugar en Dartmouth, que se puede considerar como el nacimiento de la inteligencia artificial. (Ruiz & Basualdo, 2001)

Un año después del congreso de Dartmouth, Franz Rosenblatt fue capaz de construir la que se conoce como la primera red neuronal, llamada Perceptrón. Su función consistía en reconocer patrones, pudiendo además generalizar; es decir, era capaz de reconocer patrones similares para lo



que no se la había entrenado. Gracias a este trabajo, en el año de 1960, Bernard Widroff y Marcian Hoof crearon la primera red neuronal con fines comerciales. (Ruiz & Basualdo, 2001)

No fue nada fácil el desarrollo de las redes neuronales, ya que a lo largo de la historia han aparecido muchos detractores de estas, como lo fueron, Marvin Minsky y Seymour Papert, quienes demostraron matemáticamente que el Perceptrón era muy débil porque no era capaz de aprender funciones de tipo no lineal usadas extensamente en el mundo de la informática. (Ruiz & Basualdo, 2001)

La principal consecuencia que tuvo la demostración de Minsky y Papert es que consiguieron que se desechase el concepto de red neuronal hasta que Paul Werbos y Stephen Grossberg enunciaron algunas teorías que impulsaron la idea. Werbos diseñó un algoritmo basado en el principio de propagación hacia atrás. Por otro lado, Grossberg creó la teoría de resonancia adaptada (TRA), con la que introducía un nuevo concepto de arquitectura de red similar al funcionamiento de nuestro cerebro.

A partir de 1986, se produjo la explosión del concepto en lo que a investigación y desarrollo se refiere, llegándose a publicar una gran cantidad de trabajos con relación a redes neuronales.

#### **4.1.13 Ventajas de las redes neuronales**

Hay una gran similitud entre las redes neuronales y el funcionamiento del cerebro, esto hace que sean utilizadas en multitud de áreas, ya que presentan una gran cantidad de ventajas; aprendizaje adaptativo, autoorganización, tolerancia a fallos, operación en tiempo real e introducción sencilla en la tecnología actual.

##### ***4.1.13.1 Aprendizaje adaptativo***

Las redes neuronales tienen la capacidad de aprender a realizar tareas en base a un entrenamiento previo; es decir, cuando se presenta una tarea para la cual no ha sido entrenada, la red, por similitud con las tareas para las que se ha entrenado, resuelve la tarea inicial.

Debido a lo anterior, es posible afirmar que las redes neuronales son adaptables y dinámicas. Son redes adaptables porque la capacidad de autoajuste que poseen las neuronas de la red; son dinámicas porque son capaces de cambiar acorde con los cambios que se producen en el medio que las rodea.

Teniendo en cuenta lo mencionado, la duda que surge consiste en que si la red aprende por sí sola: ¿cuál es el papel del desarrollador a la hora de programarla? La función de este a la hora de programar la red consiste en diseñar un algoritmo de aprendizaje lo suficientemente resistente como para que la red pueda adaptarse de la mejor manera a los cambios que sufra en el medio donde se encuentra. (Ruiz & Basualdo, 2001)

#### ***4.1.13.2 Autoorganización***

En este aparte se mencionará la característica más importante de las redes neuronales, que consiste en que la red es capaz de distribuir la diferente información que almacena a lo largo del proceso de aprendizaje; es decir, la red se organiza por sí sola con el fin de conseguir un objetivo específico.

La capacidad de autoorganizarse es lo que permite a la red que sea capaz de generalizar situaciones; es decir, que es capaz de dar una respuesta adecuada cuando se le presenta un problema para el que no ha sido entrenada. Esto se da cuando la información de entrada es difusa o se encuentra incompleta. (Ruiz & Basualdo, 2001)

#### ***4.1.13.3 Tolerancia a fallos***

Este concepto se basa en la inmunidad que tiene la red ante determinados fallos; es decir, en la actualidad hay una gran cantidad de sistemas informáticos en lo que si uno de sus componentes se ve afectado produce una caída repentina del sistema. En cambio, si en las redes neuronales se produce un fallo repentino de memoria (un número de neuronas se ve afectado), puede ser que el sistema se vea influenciado, pero puede continuar realizando su función.

De lo anterior, hay dos aspectos que son importantes reseñar: con la finalidad de que la red sea más resistente, el desarrollador puede crearla de tal manera que la red reconozca determinados patrones de ruido con la finalidad de que sean descartados, además, también existen redes que se diseñan con el fin de que, si parte de la red es destruida, ésta pueda seguir realizando su función.

Esta resistencia a fallos reside en que la red almacena la información de manera no localizada; es decir, la información se almacena en las conexiones entre neuronas y de manera redundante. (Ruiz & Basualdo, 2001)

#### ***4.1.13.4 Operación en tiempo real***

Esta característica es una de las más importantes, ya que hoy lo que buscan los usuarios en que los sistemas sean capaces de realizar las operaciones en el menor tiempo posible. Las redes neuronales son las más adecuadas para esto ya que gracias a la gran cantidad de neuronas que poseen, realizan las diferentes tareas de manera casi instantánea, debido a que entre ellas se produce una división del trabajo a la hora de resolverlas. (Ruiz & Basualdo, 2001)

#### ***4.1.13.5 Introducción en la tecnología actual***

Las redes neuronales se diseñan y desarrollan con el fin de realizar una sola tarea, pero esto no impide que dispongan de una gran versatilidad ya que pueden ser rápidamente entrenadas y

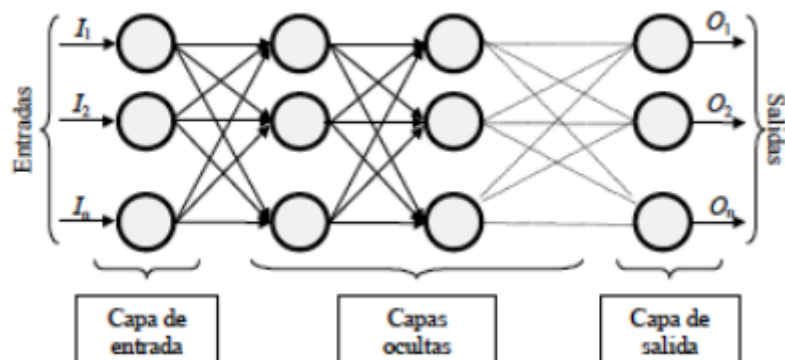
verificadas para realizar otra tarea. Además, las redes neuronales se suelen utilizar como mejoras a los sistemas existentes con el fin de que estos tengan un desarrollo más amplio. (Ruiz & Basualdo, 2001)

#### 4.1.14 Funcionamiento de una red neuronal

Dentro de la estructura básica de una red neuronal es importante reseñar que se puede dividir en tres funciones principales [Imagen 8](#): función de entrada, función de activación y función de salida.

*Imagen 8*

*Esquema básico de una red neuronal*



*Nota.* Imagen tomada de capas de una red neuronal, (Ruiz & Basualdo, 2001)

##### 4.1.14.1 Función de entrada

La utilidad de esta función consiste en que es capaz de condensar las diferentes entradas que tiene la red (input), bajo una sola entrada global (global input).

Este proceso se constituye de diferentes pasos. En primer lugar, se multiplica cada valor de entrada por un peso; este peso será mayor o menor dependiendo de la importancia que tenga la

variable a la hora de resolver la tarea en cuestión (por consiguiente, la variable más importante será la que tenga el mayor peso y la más insignificante la que tenga menor peso).

Finalmente, para dar paso a la función de entrada global, se agrupan los productos resultantes que se han mencionado en el párrafo anterior siguiendo un determinado patrón dependiendo de la finalidad y del resultado que se pretenda obtener. (Ruiz & Basualdo, 2001)

#### ***4.1.14.2 Función de activación***

La función de activación tiene el objetivo de determinar el estado de activación y la capacidad de procesamiento de una neurona. Para ello se deben distinguir dos tipos de neuronas: aquellas que se pueden encontrar activadas o desactivadas (por lo que sus valores son solo 0 o 1), y aquellas que pueden usar un porcentaje de su capacidad (por lo que los valores que pueden tener se encuentran dentro del intervalo cerrado 0-1). (Ruiz & Basualdo, 2001)

#### ***4.1.14.3 Función de salida***

Esta función es aquella que transmite el resultado global de la tarea, o el resultado de una operación realizada por una neurona a la siguiente. Para ello es vital establecer un valor umbral a la salida con el fin de determinar salidas válidas de aquellas que no lo son.

Los dos tipos de funciones más comunes son: función identidad (en la que el valor umbral a la entrada tiene que ser igual a la salida), y la función binaria (en la que la salida puede tomar valor 1 (activada) o 0 (desactivada) en función de si se supera el valor umbral). (Ruiz & Basualdo, 2001)

#### **4.1.15 Utilidad de redes neuronales en sistemas de reconocimiento facial**

Las redes neuronales tienen una gran cantidad de aplicaciones y se usan en multitud de ambientes diferentes: pasando por la biología hasta llegar al campo militar. Aunque es importante

mencionar que hoy en día estas redes tienen un gran uso en el campo empresarial y manufacturero, sobre todo para clasificar objetos válidos de aquellos que están dañados, por ejemplo:

En lo que al reconocimiento facial se refiere, las redes neuronales ayudan a mejorar el sistema por dos motivos principales. El primero es la capacidad de aprendizaje que tienen estas redes, ya que al introducir un número determinado de imágenes conseguimos que la red aprenda esa nariz del perro y posteriormente puede reconocerla en multitud de casos como pueden ser: al tomar una fotografía o al no saber cuál es el perro real entre dos que sean idénticos.

Además, el otro aspecto que es importante mencionar, es que las redes neuronales dotan al sistema de una rápida capacidad de procesamiento, haciendo que la detección de la nariz sea casi instantánea. Este aspecto es importante ya que la finalidad de nuestro sistema de reconocimiento será introducirlo en zonas con requerimientos de identificación por lo que esta debe ser casi de manera inmediata para evitar que el usuario tenga que esperar un tiempo alto.

## **Deep learning vs Machine learning**

### **4.1.16 Introducción al concepto de inteligencia artificial**

Desde la antigua Grecia el hombre siempre ha soñado con crear máquinas que sean capaces de pensar por sí mismas. Actualmente, el concepto de inteligencia artificial es un campo muy próspero en el que los científicos están realizando inmensidad de investigaciones objeto de grandes avances. Hoy en día, estos avances están consiguiendo que, tareas que a priori eran difíciles de completar por los empleados, sean realizadas por computadores de manera más rápida y eficaz.

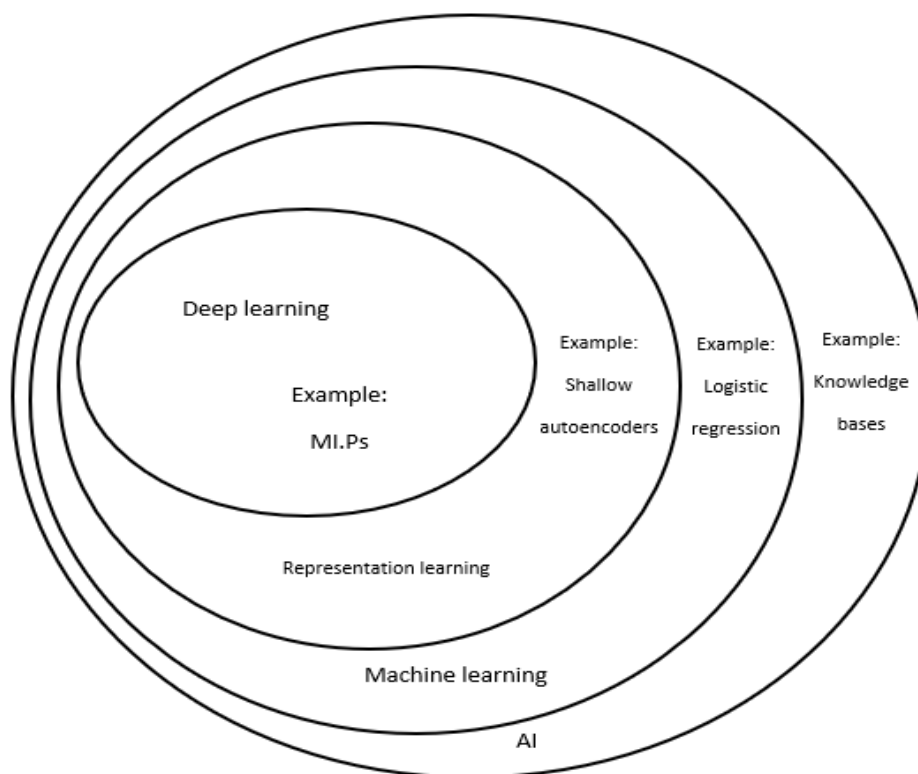
Para ello, la idea consiste en conseguir que los computadores sean capaces de aprender de la experiencia y que sean capaces de entender que el mundo se estructura en base a una jerarquización de conceptos. La jerarquización de conceptos se refiere a que, un concepto que a priori parece

complicado y muy sofisticado sea fácil de entender por una máquina a través de la división de este en conceptos más simples. (Courville, Goodfellow, & Bengio, 2015)

Los campos que abarca la inteligencia artificial son muy numerosos e inexplorados ([Imagen 9](#)). Desde resolución de sencillos problemas y toma de decisiones, hasta ser capaz de derrotar a campeones del mundo en partidas de ajedrez. Esta forma de poder codificar el mundo en un lenguaje informático se conoce como Knowledge base (traducido literalmente como base de conocimiento). Para ello, lo que se realiza es obtener reglas de funcionamiento de un determinado proceso, codificarlo e introducirlo en un computador con el fin de que comience a aprender.

### *Imagen 9*

#### *Alcance de la inteligencia artificial*



*Nota.* Adaptada de Alcance de la inteligencia artificial. (Courville, Goodfellow, & Bengio, 2015)

El proceso de la ilustración anterior es lo que resulta más complicado porque cada proceso es independiente y en numerosas ocasiones resulta imposible de poder matematizarlo a través de cálculos, debido a la gran variedad de variables que hay que tener en cuenta.

#### **4.1.17 Machine learning**

##### ***4.1.17.1 Introducción y definición***

Machine learning (traducido literalmente como aprendizaje de la máquina) consiste en un sistema computacional que es capaz de mejorar por sí mismo a través de la experiencia que va adquiriendo tras la realización de diferentes tareas. Además, se puede considerar como una forma de transformación de datos en software. En definitiva, la ciencia del Machine learning consiste en crear un modelo informático que sea capaz de predecir patrones, tendencias o una determinada salida una vez que ha sido “entrenado” mediante una base de datos bastante extensa; hablando de manera coloquial, se trata de una manera de predecir el futuro con los datos y experiencias que se han obtenido en el pasado. (Barnes, 2015)

Es importante reseñar que la ciencia del Machine learning es efectiva bajo condiciones determinadas, siendo capaz de ir más lejos de los estándares que hasta el momento estaban impuestos por las leyes de la programación y de la informática.

Para poder explicar los principios básicos del Machine learning es necesario compararlo con los paradigmas que se presentan hoy en día en el mundo de la informática. Los modelos tradicionales de programación se basan en obtener una determinada salida a partir de unos datos de entrada que son procesados por un software ([Imagen 11](#)). Por otro lado, el Machine learning persigue el deseo de crear un software que sea capaz de predecir un modelo de comportamiento a



través de unos datos de entrada y un histórico de los resultados que se ha obtenido para unos determinados parámetros de entrada ([Imagen 10](#)). (Barnes, 2015)

*Imagen 10*

*Principio de funcionamiento del Machine learning*

## Machine Learning

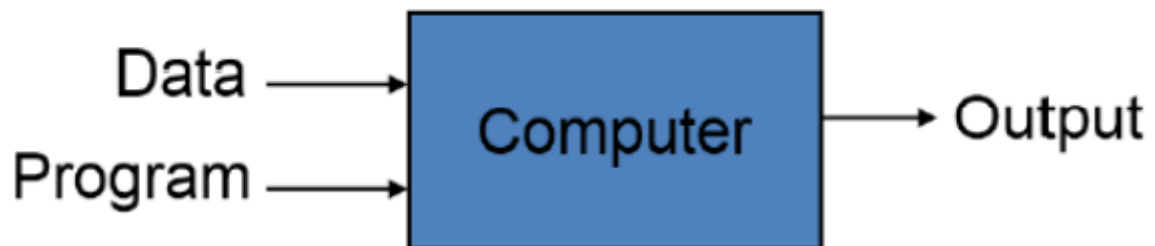


*Nota.* Imagen tomada de Machine learning, (Ruiz & Basualdo, 2001)

*Imagen 11*

*Principio de funcionamiento de la programación tradicional*

## Traditional Programming



*Nota.* Imagen tomada de Machine learning, (Ruiz & Basualdo, 2001)

Los diversos actores en el mundo de la informática actualmente saben de la importancia de este nuevo concepto hasta el punto de que grandes personalidades, como Bill Gates, han afirmado que la creación de un sistema operativo que incluya el principio de Machine learning podría llegar a tener un valor diez veces mayor que el que tiene actualmente el Sistema operativo Microsoft.

La creación de este tipo de programas se produce gracias a ese miedo que tiene todo ser humano ante la posibilidad de conocer algo que los demás si conocen y que puede ser útil para él. Esto motive la creación de lo que se conoce como redes sociales que junto con modelos de análisis predictivo permiten a las compañías realizar estrategias de negocio.

Por ejemplo, la empresa de venta online Amazon ha desarrollado un Sistema capaz de recomendar a los usuarios diferentes productos adecuados para cada uno de ellos en función de las compras que han ido llevando a cabo; dicho Sistema recomienda otros objetos que puedan complementar a otro que se ha comprado con anterioridad. (Barnes, 2015)

Una de las condiciones más importantes para la creación de un algoritmo de este tipo consiste en poseer un histórico de todos los datos de entrada y las correspondientes salidas que se han dado en un determinado entorno. Además, es importante conocer datos de los potenciales usuarios del sistema como, por ejemplo, el sexo, la edad, los estudios, etc. Esto último permitirá introducir una serie de filtros a la hora de generar los patrones de comportamiento, hacienda que sean los más exactos y adecuados posibles.

Adicionalmente, estos programas son capaces de realizar predicciones en un plazo de tiempo corto. Por ejemplo; si un programa detecta que un usuario ha comprado en una página web equipo de futbol y unas zapatillas de deporte, al momento debería poder recomendarle que la próxima adquisición debería ser un balón de futbol. Esto es lo que de verdad supone un reto para los

creadores de algoritmos de Machine learning, ya que en cuestión de segundos el programa debe ser capaz de retener los datos, interpretarlos y generar posibles soluciones que no se alejen mucho de los patrones originales.

En resumen: el Machine learning se podría comparar con el proceso de evolución que sufre un niño durante su crecimiento, en el que gracias al paso del tiempo los niños van ganando experiencias que posteriormente le servirán para elaborar modelos de comportamiento ante situaciones nuevas.

Es importante reseñar que todos los avances en este campo se deben gracias a diversos factores, entre los que se pueden resaltar: el crecimiento exponencial del uso de equipos y tecnologías que almacenan información (sensores, IoT...), el abaratamiento del precio para almacenar información digital, la facilidad de acceso a los datos (uso de las nubes) y el auge de lo que se conoce como Big Data y minería de datos.

De todos los factores anteriormente mencionados, es vital indicar el efecto que ha tenido la aparición del IoT. Este concepto IoT, Internet of things (traducido literalmente como internet de las cosas), se refiere a la capacidad que tienen determinados dispositivos de todo tipo (como los domésticos: lavadores, neveras, etc.) de conectarse a internet para que puedan ser controlados a distancia, ofrecer información en tiempo real de su estado, etc. (es decir, generalizar la conexión a internet a todo tipo de dispositivo, desde la bombilla del salón, al motor de apertura de un garaje). Dichos dispositivos, además de que se puedan activar de manera remota, permiten precisamente elevar toda esa información sensorial hacia internet (usualmente, en alguna nube privada, pública o híbrida). Toda esa información (recopilada en un orden de magnitud exponencialmente superior a como tradicionalmente se venía haciendo), es exportada por medio de novedosas técnicas, incluidas dentro del paradigma del Big Data. Lo anterior, permite que programas basados en el

principio de Machine learning puedan predecir el momento más adecuado para realizar actividades cotidianas. (Barnes, 2015)

Otra principal ventaja del sistema de Machine learning es que al ser un programa informático permite considerar una gran cantidad de variables, muchas más de las que el ser humano es capaz de tener en cuenta. Con esto y teniendo en cuenta que al día de hoy las empresas doblan los datos con los que trabajan cada dieciocho meses, es más fácil que las aplicaciones basadas en Machine learning predigan comportamientos mejor de lo que lo podría hacer un ser humano.

#### ***4.1.17.2 Ejemplo en la historia en la aplicación de Machine Learning***

A lo largo de la historia se ha tenido muchos casos precursores de Machine learning. El primero de estos, data del siglo XIX, en el que los granjeros confeccionaron un libro que se conoce como el almanaque de los granjeros. Este libro consistía en una serie de anotaciones que los granjeros realizaban sobre las fases de la luna, lluvias, épocas de sequía y otros, con el fin de predecir la evolución de sus cosechas en sus años anteriores.

En verdad que este pequeño almanaque de los granjeros se realizaba a mano y las predicciones no eran realizadas por máquinas sino por personas, por lo que no es un ejemplo del todo adecuado en lo que a aplicación del Machine learning se refiere. (Barnes, 2015)

Por el contrario, el servicio de correos de los Estados Unidos de América ha desarrollado un sistema que permite procesar y gestionar el correo basándose en modelos de análisis predictivo. Esto es algo que se entiende como un proceso muy complicado ya que el sistema de correos se compone de postales, cartas y paquetes: tres tipos de elementos muy diferentes entre sí que se deben tratar de manera diferente; además, dentro de cada grupo se pueden realizar subgrupos ateniendo a las diferentes formas, tamaños y pesos que puedan tener cada uno.

Este sistema ha sufrido una evolución notable a lo largo de toda su historia. Su creación en los años 80 cuando, con la evolución de los computadores y de los lenguajes de programación, pudieron crearse los primeros algoritmos que permitían reconocer la escritura de las direcciones de los paquetes y redirigirlos a los centros de distribución. Pero no fue hasta finales de los años 90 cuando este sistema dio un gran salto, e introdujo en los paquetes lo que se hace conocer como los códigos OCR. Con ellos, se podía ir almacenando de datos de todos y cada uno de los paquetes que circulaban por las diferentes oficinas. Esto supuso que cuando un paquete llegaba a un centro de distribución, en función de algunos datos como son su peso y forma, se podía predecir ciudades o estados a los que se debía enviar el paquete. (Barnes, 2015)

Otro ejemplo bastante visual supone la preparación de computadores para que sean capaces de evaluar los movimientos que realiza una persona jugando ajedrez, con el fin de predecir los siguientes y reaccionar ante ellos. Por ejemplo, en 1997, la que se conoce como Deep Blue Chess Machine que fue capaz de derrotar al entonces campeón del mundo Gary Kasparov. Otro ejemplo parecido a éste es del programa Watson, creado en febrero de 2011 por IBM, que era capaz de responder a cualquier tipo de preguntas referentes al lenguaje inglés. Lo más curioso de este programa era que no estaba conectado a internet, por lo que su única fuente de búsqueda eran los datos que tenía almacenados en su interior y las relaciones que existían entre estos.

En definitiva, uno de los aspectos más importantes de Machine learning es que el programa se encuentra siempre en un proceso constante de aprendizaje por el mero hecho de encontrarse en funcionamiento. Siempre que haya un error en las predicciones, el programa lo anota y aprende para no volver a cometerlo en situaciones futuras. En lo referente a este aspecto, se puede afirmar

que un programa basado en Machine learning nunca se puede decir que está terminado, porque este tipo de software se encuentra en un proceso de mejora continua constante.

Esa capacidad de mejora continua es lo que hace a los programas de Machine learning valiosos, ya que tal y como dijo Winston Churchill: “Todos los hombres cometen errores, pero solo aquellos que son los más sabios aprenden de sus errores”. Siguiendo esta afirmación de Churchill, las máquinas basadas en machine learning son unas máquinas sabias que los seres humanos pueden utilizar como una herramienta muy poderosa a la hora de realizar predicciones futuras. (Barnes, 2015)

A modo de resumen, las máquinas de Machine learning están basadas en principios de inteligencia artificial que son capaces de generar, a partir de modelos de análisis predictivo, determinados patrones y actividades que pudieran ocurrir en el futuro. Esto es algo que a lo largo de la historia se ha presentado en numerosas películas de ciencia ficción en las que se mostraba máquinas que eran capaces de pensar por sí mismas (Terminator, Wall-E...), pero que actualmente es algo que se encuentra más lejos de la ficción que de la realidad.

#### ***4.1.17.3 Limitaciones del Machine Learning***

Los algoritmos de Machine learning tienen una gran capacidad en lo que a resolución de problemas se refiere, pero aún son muy limitados en resolver el principal problema por el que se concibió la inteligencia artificial: detección, reconocimiento de objetos y voz.

##### ***4.1.17.3.1 La maldición de la dimensionalidad***

Uno de los principales problemas que presentan los algoritmos de Machine learning, sucede cuando los datos que se le introducen poseen varias dimensiones. Este fenómeno es lo que se conoce como la maldición de la dimensionalidad. Este problema aparece en numerosos ámbitos

científicos. Consiste en que, a medida que las dimensiones de los datos aumentan, las posibles configuraciones a la hora de reordenarlos aumentan de manera exponencial bajo la fórmula:  $N^{\circ de elementos} = N^{\circ de dimensiones} = N^{\circ de regiones}$ . (Courville, Goodfellow, & Bengio, 2015)

Por ejemplo, si se tiene un archivo de datos que se compone de 10 elementos agrupados en dos dimensiones y se tendría un total de  $10^2 = 100$  regiones. De esta manera no puede parecer relevante, pero en cambio sí se utiliza un archivo que se compone de 100 elementos repartidos en las tres dimensiones del espacio tendríamos un total de  $100^3 = 1000000$  regiones. (Courville, Goodfellow, & Bengio, 2015)

Este problema se resume en un problema estadístico, ya que el algoritmo de Machine learning no es capaz de resolver el problema ante tantas combinaciones posibles ([Imagen 12](#)).

### *Imagen 12*

*Demostración de las dificultades de los algoritmos de machine learning a la hora de reordenar objetos*



*Nota.* Imagen tomada de dificultades de los algoritmos de Machine learning, (Courville, Goodfellow, & Bengio, 2015)

En términos matemáticos, el número de configuraciones es mucho mayor que el número de ejemplos para lo cual la red neuronal fue entrenada. Se considera que el espacio es una red y en cada celda se encuentra un elemento que compone el archivo, en pocas palabras al utilizar el algoritmo de Machine learning lo que hará es que en cada celda buscará una configuración que termine teniendo sentido. Es muy difícil que el algoritmo en su memoria al encontrar tantas celdas tenga una solución para cada una de ellas. La solución que encuentra el algoritmo es plantear una distribución parecida a la que tiene en su base de datos entrenada, haciendo que en algunas ocasiones sea erróneo.

#### ***4.1.17.3.2 Constancia local y suavidad de la regularización***

Para que los algoritmos de Machine learning puedan generalizar y obtener predicciones en las salidas, es necesario que estos sean guiados, para que sepan qué funciones deben de aprenderse con el fin de utilizarlas en las predicciones. Es decir, los conocimientos previos influyen directamente sobre la función de una manera directa y sobre los parámetros de manera indirecta, debido a la relación existente entre parámetros de entrada y función. Estos algoritmos introducen unas funciones muy simples para generalizar que general mucho error en la predicción que realizan ante comportamientos determinados.

Hay muchos tipos de inputs (entradas) que se utilizan para que el algoritmo sepa qué funciones debe conocer a la hora de generalizar. Dichos inputs pueden ser de tipo explícito o implícito; pero el principal problema que estos generan consiste en que muchos de ellos generalizan de manera suave y local, sin tener en cuenta muchas variables que introduciéndose generarían un error alto en la predicción.

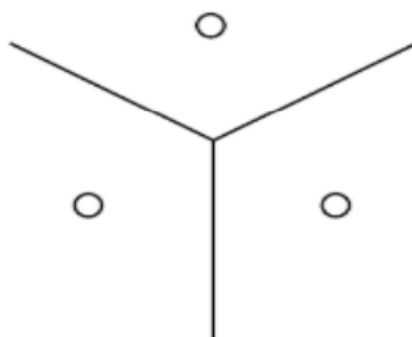
La función de generalización se introduce en el algoritmo durante el proceso de aprendizaje bajo el nombre  $f(x)$ , que satisface la siguiente condición  $f(x) = f(x + E)$ . la función de que



obtiene es la misma, pero con un pequeño cambio (que es lo que permite generalizar ante situaciones similares). Básicamente lo que está ocurriendo es que cuando a la función se le presenta una nueva tarea, busca en la base de datos una solución que haya sido adecuada antes para una tarea similar. En el caso de que haya varias soluciones posibles, lo que ocurre es que la función combina toda para obtener una solución final que se aceptará como válida.

### *Imagen 13*

#### *Ejemplo de generalización de un algoritmo de machine learning*



*Nota.* Imagen tomada de [www.google.com](http://www.google.com)

En la [Imagen 13](#) se puede observar una representación de la generalización de un algoritmo basado en Machine learning. En la imagen, las líneas representan los límites de las diferentes regiones, y los círculos las soluciones a una tarea (las cuales se han introducido en la fase de entrenamiento). Lo que ocurre es que, para una tarea que se encuentra en una región determinada, el algoritmo acepta como válida la solución que se proporcionó en la fase de entrenamiento. En caso de que la distancia a dos soluciones sea la misma, lo que ocurre es que se combinan las soluciones equidistantes entre sí con la finalidad de obtener el modelo final.

En conclusión, es que, en numerosas ocasiones, las soluciones que proporciona el algoritmo no son adecuadas, porque para cada una de las tareas influyen numerosas variables que tienen un valor diferente cada vez.

Esto no solo afecta las funciones, sino que también afecta a los árboles de decisión de los algoritmos. Esto es así porque los árboles de decisión poseen tantas hojas como ejemplos de entrenamiento se tenga en la base de datos. Esto concluye en que en numerosos casos los árboles no poseen los ejemplos necesarios, por lo que la predicción que se proporcionara no tiene el nivel de confianza suficiente.

Para entender este concepto de manera práctica, se puede suponer que la función de entrada es un tablero de ajedrez. Suponga que el número de veces que ha sido entrenada la función es mucho más pequeño que el número de cuadrados que posee el tablero. En este caso, la función operaría correctamente para adivinar el color de un punto que se encuentre situado dentro del tablero, pero, en cambio, no se puede garantizar que se proporcione una solución correcta para puntos que se sitúan fuera de los cuadrados que sí conoce la función. En conclusión, estas funciones se comportan bien cuando el número de dimensiones con el que trabajan es un número bajo, ya que, por el contrario, cuando se tienen varias dimensiones de por medio, se van produciendo errores en las soluciones que se proporcionan para cada una de ellas que sumados dan una predicción no aceptable.

#### **4.1.18 Deep learning**

##### ***4.1.18.1 Introducción y definición del concepto de machine learning***

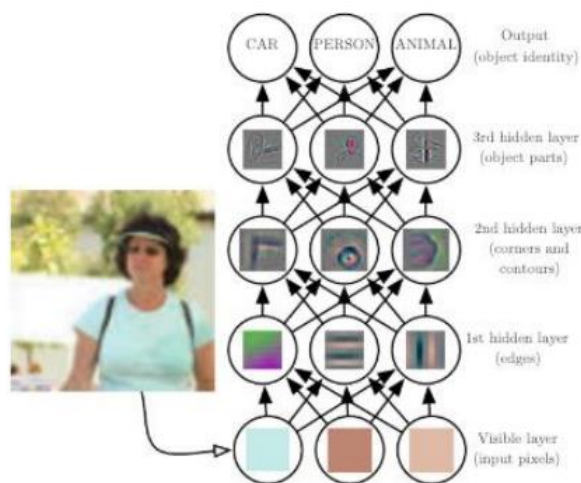
El concepto de Deep learning (traducido literalmente como aprendizaje profundo) es una idea que se encuentra incluida dentro de la tecnología del machine learning. Dicha idea está basada en uno de los principios de la inteligencia artificial: la jerarquización de conceptos. Para ello, los

algoritmos de Deep learning lo que hacen es dividir es dividir un concepto simple en varios más sencillos, que son más fáciles de resolver por un computador. (Courville, Goodfellow, & Bengio, 2015)

Esta concepción de resolución por pasos es lo que generan las Deep networks, también conocidas como redes por capas o redes profundas, en las que a medida que se baja de nivel o de capa se encuentran conceptos más simples que el anterior. En términos informáticos, lo que ocurre es que el algoritmo de resolución está constituido por una sola función, que a su vez se compone de funciones más simples en las que la información de entrada se va dividiendo, y las salidas de cada función se van agrupando a medida que suben de nivel hasta proporcionar la salida final. (Courville, Goodfellow, & Bengio, 2015)

#### *Imagen 14*

#### *Ejemplo del funcionamiento de un algoritmo de deep learning*



*Nota.* Imagen tomada de algoritmo de Deep learning, (Courville, Goodfellow, & Bengio, 2015)

En la [Imagen 14](#) se puede observar un ejemplo del funcionamiento de un algoritmo basado en Deep learning para reconocimiento de imágenes. La imagen que se muestra como problema es muy difícil de interpretar como un solo concepto debido a la cantidad de píxeles que posee. Para ello, el algoritmo, ante la dificultad de resolver el problema inicial, decide dividir el problema en varios que se atacarán por separado. El siguiente paso por seguir es clasificar los problemas en dos tipos principales: aquellos que se encuentran en la capa visible, y aquellos que se encuentran en las capas ocultas. La capa visible recibe este nombre porque es aquella que contiene las variables visibles, o las que son extraídas directamente de la imagen a reconocer. Por otro lado, se tiene una serie de capas ocultas, que son aquellas en las que se analizan otras características abstractas de la fotografía, y que relacionan a las variables visibles entre sí. La primera capa oculta caracteriza el brillo de los diferentes píxeles para que la segunda pueda establecer los límites de la imagen. Con esto, la tercera puede determinar las diferentes formas y objetos que aparecen en la imagen, hasta que finalmente la última capa determina que tipos de objetos se encuentran en la fotografía. (Courville, Goodfellow, & Bengio, 2015)

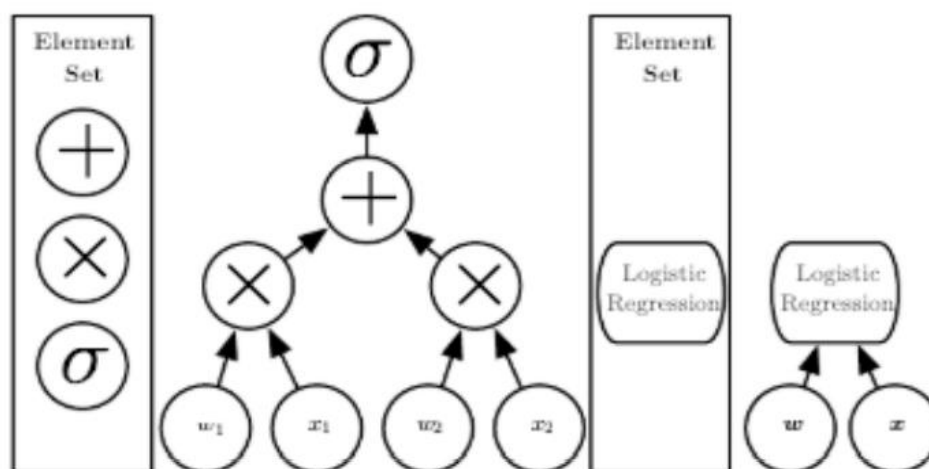
Estos nuevos algoritmos basados en Deep learning generalmente se aplican cuando el archivo a analizar tiene un dominio bastante complicado, como pueden ser archivos de audio y video, en los que es necesario analizar todo el universo que rodea el archivo. Algunos ejemplos de campos de aplicación son: implementación en la CPU para acelerar la velocidad de procesamiento, conexión entre CPU y GPU para mejorar el procesamiento de imágenes, entre otros.

En principio puede parecer que esta nueva concepción presenta solo ventajas, pero eso no es así. El principal problema que tienen las Deep networks es que suele ser muy difícil determinar la profundidad óptima de la red. Un problema se puede dividir infinidad de veces, lo que generaría una red con un número de capas infinitas imposible de diseñar.

Actualmente existen dos formas principales de determinar la profundidad de una red. La primera y más sencilla consiste en contar el número de instrucciones que realiza la red para resolver una tarea; el camino más largo que se sigue desde la base de la pirámide hasta la cúspide. Esta no es una forma muy precisa de calcularlo, ya que el número de instrucciones que se realizan en una red depende de muchos factores. Por ejemplo, en función de la tarea y el tipo de lenguaje que se ha utilizado en la programación, un determinado algoritmo puede llevar a cabo más o menos instrucciones.

### Imagen 15

Ejemplo del cálculo de la profundidad de una red



*Nota.* Imagen tomada de profundidad de una red, (Courville, Goodfellow, & Bengio, 2015)

En la [Imagen 15](#) se muestra una red de Deep learning cuya función es predecir resultados a través de un modelo de regresión logística. Un modelo de regresión logística es un tipo de análisis matemático basado en la regresión, cuyo fin consiste en predecir el valor que puede tomar una

variable que es capaz de adoptar un número limitado de categorías (también conocido como variable categórica), en base a unas variables independientes.

El problema que se presenta en esta imagen 15 es el criterio que sigue para calcular la profundidad de la red. Si se atiende a cada una de las operaciones que realiza el modelo, se podrá afirmar que la profundidad de la red es de dos capas; en cambio, si se supone que el modelo es una operación (el cual se encuentra englobado dentro de una red de mayor tamaño), se afirmaría que la profundidad del modelo es uno.

Ante esta ambigüedad en el cálculo de la profundidad de la red surgió un segundo modelo que cambiaba totalmente esta dinámica. Este nuevo enfoque afirma que la profundidad de una red no se podía calcular a través del gráfico de la red. El gráfico de la red solo proporciona información de las relaciones que existían entre los conceptos de nivel inferior con los de nivel superior. En este caso, la profundidad se determina a través de la profundidad del diagrama de flujo.

Como ejemplo del cálculo de la profundidad, usted puede suponer que a una red se le presenta una imagen en la que un ojo aparece cubierto. En este caso, las operaciones que realiza la red solo son dos: por una parte, detectar el ojo, y por otra posteriormente identificar la cara. Pero esto no es una afirmación real, ya que el sistema puede inferir que a pesar de que no detecte el otro ojo, éste se encuentra presente, por lo que está realizando más operaciones. Concluyendo: para este modelo de cálculo, la profundidad tiene un valor de  $2n$  capas, siendo  $n$  el número de veces que la red analiza una determinada característica. (Courville, Goodfellow, & Bengio, 2015)

Hoy en día, no se ha llegado a un acuerdo acerca de qué método es el más correcta para calcular la profundidad de las redes. Esto provoca que no se pueda dar un valor exacto de profundidad para una determinada arquitectita de red; pero en lo que sí se está de acuerdo es que el modelo de Deep

learning requiere que durante el entrenamiento previo se almacenen en la base de datos de la red neuronal una cantidad de funciones y/o conceptos mayores que los que se requerirán para el aprendizaje automático normal. (Courville, Goodfellow, & Bengio, 2015)

A modo de resumen, las redes de Deep learning constituyen un primer intento de aproximación a los objetivos que se pretendían cuando se concibió el concepto de inteligencia artificial. Es un tipo específico de algoritmo de Machine learning que permite al sistema mejorar gracias a la experiencia y a los datos. Además, esta es la única forma de operar en ambientes que son realmente complicados, gracias a que es una herramienta poderosa y flexible. Por último, su funcionamiento está basado en la jerarquización de conceptos. Un concepto global se puede explicar a través de relaciones que mantiene entre sí conceptos más simples.

#### ***4.1.18.2 Deep learning en reconocimiento facial (deep residual learning)***

Como se ha mencionado anteriormente, las redes neuronales basadas en Deep learning son capaces de evaluar características de bajo/medio/alto nivel e la hora de resolver determinados problemas y tareas. Pero hay una pregunta que nos queda pendiente en responder, y consiste en saber si hay una relación directamente proporcional entre la capacidad de aprendizaje y el número de capas.

Este proceso se puede dividir en tres fases principales: a medida que se va aumentando las capas mejora la capacidad de aprendizaje de la red; posteriormente se entra en una fase en la cual la capacidad de precisión se satura, para llegar finalmente a una fase en la que el aumento de la profundidad hace que la precisión de nuestra red decrezca rápidamente.

Esta degradación de la precisión demuestra que no todos los sistemas pueden ser optimizados con facilidad. Es decir: comparando dos redes, una poco profunda y otra con un número de capas

alto, puede darse el caso que la red con más profundidad genere más errores durante la fase de entrenamiento.

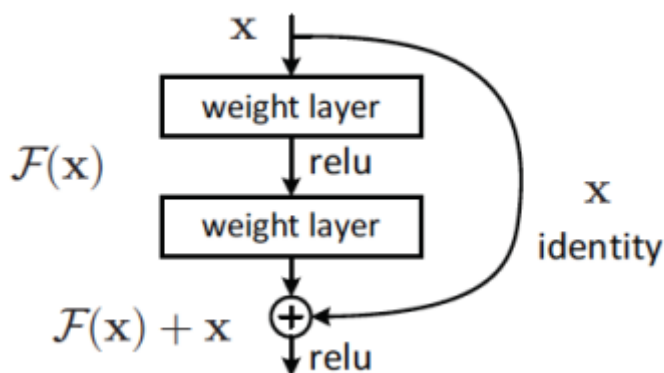
Para ello, en el campo del reconocimiento de imágenes se está buscando un método alternativo a la aplicación directa de los principios de Deep learning. En este momento es cuando entra a escena el Deep residual learning para reconocimiento de imágenes, cuya finalidad es atacar esta degradación que sufren las redes cuando su número de capas es muy alto. (Kaiming , Xiangyu, Shaoqing, & Sun, 2015)

El principio sobre el que se basa esta idea consiste en que, en lugar de esperar a que las capas de ajusten por sí solas al mapeo deseado, se obliga a que se ajusten automáticamente para que realicen un mapeo residual. Matemáticamente lo que ocurre es lo siguiente: cambiar la función inicial de mapeo, inicialmente llamada  $y = H(x)$ , por otra parte, que se puede expresar como  $y = F(x) = H(x) - X$ , para finalmente fundir el mapeo original en  $y = F(x) + x$  añadiendo un atajo a la red de valor  $x$ . Esta introducción de la identidad es lo que genera el mapeo residual buscado que a priori parece más sencillo de optimizar, en comparación con el mapeo inicial ([Imagen 16](#)). Es importante reseñar que todas las consideraciones se han realizado partiendo de la base de que las salidas ( $y$ ) y las entradas ( $x$ ) tienen las mismas dimensiones. (Kaiming , Xiangyu, Shaoqing, & Sun, 2015)



*Imagen 16*

*Funcionamiento de un bloque de una red basada en deep residual learning*



*Nota.* Imagen tomada de Deep residual learning, (Cabello Pardos, 2004)

En palabras menos técnicas, lo que ocurre es que la función que mapea toda la entrada se le resta el valor de entrada para que analice una zona más pequeña y obtener una mejor información. Una vez completada la tarea, este valor de salida se suma al mapeo total de la imagen.

La arquitectura de este tipo de redes se basa en la misma que tienen las redes planas, pero además se les añade el atajo que permite sumar a la función de mapeo el valor de identidad de la entrada. Como se ha mencionado anteriormente, este tipo de atajos solo se puede utilizar cuando las dimensiones de entrada y salida son las mismas. En caso de que la dimensión de salida aumente con respecto a la entrada, se pueden llevar a cabo de procesos diferentes. El primero consiste en que el atajo sigue funcionando de la misma manera, pero además se le añade un cero que multiplica al valor de la nueva dimensión que aparece con el fin de anularla. En segundo lugar, se añade un nuevo atajo, conocido como atajo de proyección, que lo que hace es unir las dimensiones entre sí.

## **Aplicaciones móviles**

### **4.1.19 Introducción al concepto de aplicaciones móviles**

Si se habla de una definición básica, una aplicación móvil es un programa diseñado para ser ejecutado en un smartphone, tablet y otros dispositivos móviles, que le permiten al usuario realizar actividades cotidianas, acceder a servicios, mantenerse informado, vida social, entre otros. (servisoftcorp, 2020)

Son infinitas las posibilidades que tiene un usuario al interactuar con su dispositivo móvil, hoy en día estos son casi indispensables en nuestro diario vivir, esto conlleva a que el desarrollo de software para este tipo de dispositivos sea vital para entrar en el mercado y generar consumo.

### **4.1.20 Origen de las aplicaciones móviles**

Históricamente las primeras aplicaciones se dieron a conocer a finales de los años 90. Esto no hace referencia a aplicaciones (conocidas como Apps) para teléfonos inteligentes, sino para analógicos, como por ejemplo la agenda, Tetris, culebrita, editores de tonos de llamada, etc. Estas cumplían funciones muy básicas comparadas con las que se tienen actualmente, sin embargo, en ese entonces significaron un avance abismal en la forma como se tenía concebida a los teléfonos celulares antiguos, esto generó que se abriera un mercado gigantesco, cuya competencia es, y sigue siendo voraz; lo cual nos ha permitido disfrutar de herramientas cada vez más prácticas, útiles e increíbles. (servisoftcorp, 2020)

### **4.1.21 Sistemas operativos móviles**

Entre los más importantes se pueden mencionar los siguientes:

- Android
- IOS
- Windows Phone

- Blackberry
- Symbian
- Firefox O.S
- Ubuntu Touch

De los sistemas operativos anteriores, Android lidera las estadísticas mundiales, ocupando más del 80% del mercado y el en segundo lugar se encuentra IOS con más del 10% del mercado. Esto permite ver con claridad que donde más impacto se puede generar al desarrollar una aplicación móvil es desarrollando para los sistemas operativos Android. (Tabor & Vrdoljak, 2016)

En ese orden de ideas, ya se conocen los sistemas operativos y cual es más adecuado para usar, pero ¿en qué plataformas se publican estas aplicaciones?, para ello se mencionarán las tiendas de aplicaciones más relevantes:

- Google Play de Google Inc.
- App Store de Apple
- Windows Phone Store de Microsoft

#### **4.1.22 Desarrollo nativo (iOS)**

Swift es el lenguaje es el seleccionado por el grande tecnológico de Apple para el desarrollo de sus aplicaciones móviles, es un lenguaje robusto lo que hace que sufra menos errores que otras plataformas, además de interactuar en con el teléfono de una forma más optima, y que la velocidad es mucho más rápida que en otras plataformas.

#### **4.1.23 Android**

La plataforma Android fue desarrollada por la Open Handset Alliance, liderada por Google, y ha estado a disposición del público desde noviembre del 2007. Su uso por parte de la mayoría de

los fabricantes de hardware la ha convertido en el sistema operativo de smartphones con el mayor crecimiento de todos los tiempos y el que domina el mercado hoy. (Tabor & Vrdoljak, 2016)

Android es un sistema operativo que contiene una colección de aplicaciones preinstaladas y una plataforma de desarrollo de aplicaciones soportada por un amplio paquete de herramientas. La plataforma sigue evolucionando rápidamente con la agregación regular de nuevas características, aproximadamente cada 6 meses y las actualizaciones siempre buscan mejorar temas de interfaz, seguridad, gestión, entre otros.

Uno de los problemas más debatidos cuando se desarrolla para Android es la fragmentación del sistema: la diversidad de dispositivos diferentes de diversos fabricantes y el rápido progreso de la plataforma en sí mismo, generan incertidumbre sobre si la aplicación Android podrá ejecutarse o no en cualquier dispositivo. Además, la adopción de la última versión del sistema operativo está siendo más lenta que en otras plataformas. A continuación, en la [Imagen 17](#) se evidenciará la distribución de las versiones del sistema operativo en los dispositivos del mundo.

### **Imagen 17**

*Distribución de Android según versión*

VERSIÓN DE ANDROID	PORCENTAJE
ICE CREAM SANDWICH (4.0)	0,2%
JELLY BEAN (4.1 - 4.3)	1,7%
KITKAT (4.4)	4%
LOLLIPOP (5.0 - 5.1)	9,2%
MARSHMALLOW (6.0)	11,2%
NOUGAT (7.0 - 7.1)	12,9%
OREO (8.0 - 8.1)	21,3%
PIE (9.0)	31,3%
ANDROID 10 (10.0)	8,2%

*Nota: (Datos de 2020), Imagen tomada de: [www.xatakandroid.com](http://www.xatakandroid.com)*

Por ende, para desarrollar una aplicación enfocada a Android hay que tener en cuenta varios factores, entre ellos, a que publico va dirigida, que versión soporta, publicación, etc. Esto lleva a la conclusión de que se debe realizar un análisis detallado para analizar, diseñar, desarrollar e implementar una aplicación móvil enfocada a dispositivos cuyo sistema operativo es Android.

### **Lenguajes de programación**

Mucho se habla sobre cuáles son los mejores lenguajes de programación para IA y Machine learning, los cuales tienen grandes utilidades, paquetes y algoritmos que ayudan para dicho fin, es por ello, por lo que se debe escoger el más eficiente para dicho fin y por ello, se hace en la siguiente tabla, una relación minuciosa sobre dichos lenguajes, que permiten escoger el más conveniente para desarrollar el proyecto.

*Tabla 1*

*Lenguajes de programación. Creación propia*

Lenguaje de programación	Paquetes u algoritmos	Descripción
--------------------------	-----------------------	-------------

---

Python	<ul style="list-style-type: none"> <li>▪ Matplotlib</li> <li>▪ NumPy</li> <li>▪ <b>Numba</b></li> <li>▪ scikit-learn</li> <li>▪ <b>TensorFlow</b></li> <li>▪ <b>Keras</b></li> <li>▪ <b>PyTorch</b></li> <li>▪ SHAP</li> <li>▪ OpenCV</li> </ul>	<p>Python es un lenguaje de programación relativamente nuevo, diseñado en 1991, es muy útil y de código de fácil aprendizaje para programadores nuevos, cuenta con bastantes algoritmos diseñados para desarrollo de IA</p>
R	<ul style="list-style-type: none"> <li>▪ Mice</li> <li>▪ Rpart</li> <li>▪ Party</li> <li>▪ Caret</li> <li>▪ randomForest</li> <li>▪ nnet</li> </ul>	<p>Aunque R cuenta con bastantes paquetes también para la programación, es un lenguaje más que todo usado para minería de datos, estadística, etc. Donde por medio de la información ingresada, predice una tendencia por medio de graficas.</p>
Java	<ul style="list-style-type: none"> <li>▪ Spark+MLlib</li> <li>▪ Mahout</li> <li>▪ Deeplearning4j</li> <li>▪ Weka</li> <li>▪ MALLET</li> <li>▪ ELKI</li> <li>▪ MOA</li> </ul>	<p>Java es uno de los lenguajes más utilizados para Machine learning debido a que es orientado a objetos lo que lo hace bastante consistente aparte de contar con bastantes librerías.</p>
C++	<ul style="list-style-type: none"> <li>▪ LibSVM</li> <li>▪ Shark</li> <li>▪ Mlpack</li> </ul>	<p>Es un lenguaje poco común para desarrollo de Machine learning e IA, aunque cuenta con librerías y herramientas para ello, lo que quiere decir, no es tan robusto para el desarrollo de lo ya mencionado.</p>

---

---

JavaScript	<ul style="list-style-type: none"> <li>▪ Brain</li> <li>▪ Synaptic</li> <li>▪ TensorFlow.js</li> <li>▪ Neataptic</li> <li>▪ Deep playground</li> <li>▪ DeepForge</li> <li>▪ ml.js</li> </ul>	<p>JavaScript es un lenguaje que cuenta con múltiples librerías y paquetes que hacen que su uso sea más eficaz, pero conlleva una serie de algoritmos para desarrollo de Machine learning.</p>
------------	--	--

---

*Nota:* Elaboración propia.

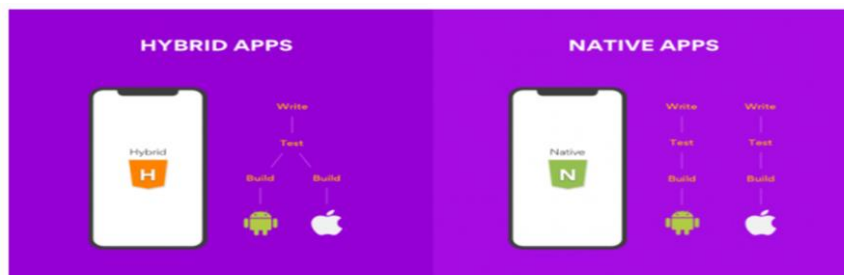
De la tabla anterior se deduce que el lenguaje más usado en los últimos tiempos y que cuenta con buen repertorio de librerías y bibliotecas para el desarrollo de inteligencia artificial, redes neuronales, etc., es Python, el cual será usado para la creación y materialización del proyecto.

### **Plataformas**

En el transcurso de los años desde que se tiene la percepción de APP (aplicaciones móviles) es bastante el cuestionamiento tanto desde su funcionamiento óptimo hasta la implementación en la plataforma más utilizada, hoy en día son dos, Android e IOS, pero si se podría implementar en ambas plataformas sin ningún problema y su funcionamiento no cambiaría y fuese igual de óptimo, esta posibilidad se le conoce como desarrollo híbrido.

*Imagen 18*

### *App nativas o híbridas*



*Nota.* Imagen tomada de tipos de aplicaciones, (Tabor & Vrdoljak, 2016)

#### **4.1.24 Plataformas nativas**

Las aplicaciones nativas son aquellas que están desarrollada solo para una plataforma, para el desarrollo del presente proyecto se escogió Android que se desarrolla sobre el lenguaje java, una aplicación nativa, quiere decir que es solo para esa plataforma lo que hace que en algún momento que se quiera implementar alguna APP en alguna otra plataforma como IOS no se podría ya que está creada exclusivamente para Android y su compilación está situada en ella, cuenta con ventajas y desventajas:

#### **Ventajas:**

- Se trabaja con las funciones SDK de la plataforma, lo que significa que no hay límites en cuanto al desarrollo
- Utiliza del entorno gráfico de la plataforma para mayor interacción con el usuario
- Se puede acceder a los componentes del hardware donde está incluida la multiplataforma, en este caso los teléfonos móviles con Android
- Proporciona la utilidad de todo tipo de funciones del dispositivo.



**Desventajas:**

- Un mayor costo
- Doble desarrollo para implementar la APP en alguna otra plataforma
- Tener tantos códigos fuente para el desarrollo de la App eficazmente

**4.1.25 Plataformas híbridas**

Las aplicaciones híbridas son multiplataforma, esto quiere decir que funcionan tanto en Android como en IOS, un ejemplo de ello es la aplicación de mensajería WhatsApp la cual es óptima en ambas plataformas lo que quiere decir que se obtiene la mayor funcionalidad y optimización en las dos plataformas, es por ello que tiene más ventajas que desventajas:

**Ventajas:**

- No necesita de bases de datos externas complejas
- No necesita rendimiento 100% nativo
- Desarrollo de código único, con lo que no se debe realizar otro código fuente para otra plataforma.
- Menor inversión ya que se realiza una sola aplicación

*Tabla 2***APP móviles**

	<b>Aplicaciones híbridas</b>	<b>Aplicaciones nativas</b>
<b>Coste de desarrollo y mantenimiento</b>	Coste menor al ser un único desarrollo para múltiples plataformas	Coste elevado debido a la programación por tipo de dispositivo con lenguajes más complejos
<b>Rendimiento</b>	Bastante bueno, excepto en aplicaciones que exigen de mucho hardware, como juegos	Óptimo

<b>Acceso a características del dispositivo</b>	Prácticamente completo	Completo
<b>Tiempo de desarrollo</b>	Mucho menor debido a que es un lenguaje mucho más simple y un único desarrollo por plataforma	Duplica como mínimo el tiempo a la híbrida debido a que serán dos desarrollos por separado

*Nota.* tomada de: Casans.A (04/06/2020) Las mejores herramientas de desarrollo de apps móviles, Next Tech, Hiberus blog, <https://www.hiberus.com/crecemos-contigo/mejores-herramientas-de-desarrollo-de-apps-movil/>

En una perspectiva de demanda del usuario se puede observar que la mayoría de la población que obtiene un teléfono móvil contiene Android, es por ello que una de las opciones para el desarrollo App y viable es desarrollo nativo, pero ¿porqué no puede ser híbrida? la respuesta es que si puede ser híbrida y es la mejor opción que se encuentra en el mercado, ya que otro porcentaje de la población tiene un teléfono móvil con IOS lo que genera que la posibilidad de utilidad de la App desarrollada incremente significativamente.

Hoy en día, la mayoría de las personas cuenta con un dispositivo Android que lo hace como el sistema operativo para móviles más usado, razón por la cual es que se ha escogido el sistema Android para desarrollar la App, precisamente, por su gran demanda en el mercado.

### **Bases de datos**

En el mundo de las bases de datos se encuentran múltiples motores que hacen que se pueda desarrollar una de ellas, pero que sucedería si existen tanto bases de datos SQL como NO SQL cuál de las dos tiene más ventajas o desventajas.

#### *Tabla 3*

##### *Bases de datos más utilizadas*

Base de datos	Motores	Descripción
SQL	<ul style="list-style-type: none"> <li>• MySQL</li> <li>• Oracle</li> <li>• Access</li> <li>• PostgreSQL</li> <li>• MariaDB</li> <li>• Microsoft SQL server</li> </ul>	Es el lenguaje estándar de gestión de datos relacionales que tiene la necesidad de organizar los datos de forma ordenada y que permita seleccionar, insertar, actualizar, eliminar y crear nuevos datos en las tablas,
NO SQL	<ul style="list-style-type: none"> <li>• Apache Cassandra</li> <li>• Redis</li> <li>• MongoDB</li> <li>• HBase</li> <li>• Neo4j</li> </ul>	Su fuerte frente a las bases de datos relacionales como SQL es que permiten almacenar información evitando errores que tiene las SQL como lo son escalabilidad y rendimiento donde en grandes cantidades de información.

*Nota:* Esta tabla muestra las bases de datos más utilizadas hoy.

#### 4.1.26 SQL o NoSQL

SQL (Structured Query Language) o NoSQL. Pregunta sobre cuál de estos dos, un Lenguaje de Consulta Estructurado o el sistema de gestión de bases de datos no relacionales de alto desempeño. Luego de realizar el análisis correspondiente, se concertó que para el desarrollo del proyecto, utilizar en algunos puntos NoSQL y ligarlo con SQL:

- Cuando los volúmenes de datos incrementan
- Cuando el nivel de escalabilidad es mayor que un modelo relacional,
- Cuando se generan muchos picos de uso del Sistema por parte de los usuarios
- Cuando el esquema de datos no es homogéneo, por decirlo así que al insertar información se tenga que almacenar en campos distintos

#### 4.1.27 Bases de datos NoSQL

Las bases de datos NoSQL en pocas palabras, son más eficientes de las bases de datos SQL ya que es un sistema de almacenamiento de datos que no cumple con el esquema entidad relación directamente no hacen uso de tablas para guardar información si no por medio de clave valor, grafos, mapeos de columnas, cuenta con múltiples ventajas para:

**Ventajas:** (Telefonica, 2014)

- Se ejecutan en máquinas con pocos recursos
- Escalabilidad horizontal
- Maneja gran cantidad de datos
- No genera cuellos de botella

Cuenta con múltiples diferencias con las bases de datos SQL y entre ellas se destacan las más importantes y relevantes: (Telefonica, 2014)

- No utiliza SQL como lenguaje de consultas
- No utilizan estructuras fijas como tablas para el almacenamiento de los datos
- No suelen permitir operaciones JOIN
- Arquitectura distribuida

Dependiendo de la forma en la que se vayan a almacenar los datos se puede encontrar con varios tipos de bases de datos, los más utilizados son los siguientes: (Telefonica, 2014)

- Bases de datos clave – valor
- Bases de datos documentales
- Bases de datos en grafo

#### 4.1.28 SQL

Es el lenguaje estándar de bases de datos con una interfaz principal utilizada para comunicarse con bases de datos relacionales, una de sus mayores utilidades es el orden de los datos siendo almacenados en tablas directamente enlazadas con las demás que estén creadas dentro de un sistema a eso se le llama bases de datos relacionales. Cuenta con varias relaciones como son:

- Uno a uno
- Uno a muchos y muchos a uno
- Muchos a muchos
- Relaciones de autorreferencia

Es un lenguaje basado en álgebra y cálculos lo que otorga que de una forma muy sencilla se puedan consultar, insertar, eliminar y saber con exactitud donde está la información guardada en ella contiene múltiples ventajas.

##### **Ventajas:**

- Almacenamiento y operaciones entre los datos
- Uso de tablas con un uso sencillo
- Seguridad de los datos y gran acceso a los mismos
- Múltiples transacciones y gestionar una gran cantidad de datos

Las bases de datos son un asunto importante a la hora de seleccionar cual es la mejor opción para almacenar los datos. Y cada una de ellas cuenta con sus ventajas lo que hace que sean robustas,

es por ello por lo que el proyecto se basará en bases de datos NoSQL ya que se pueden manejar más datos que en una base de datos SQL.

### **Algoritmos de redes neuronales**

Cada uno de los lenguajes de programación conocidos por el mundo se encuentran sujetos a muchos algoritmos que funcionan para diferentes programas, es por ello, que uno de los lenguajes predilectos y más funcionales para la programación de IA y Machine Learning es Python, ya que cuenta con diferentes algoritmos paquetes o librerías que ayudan a dicho fin, entre ellos destacan los siguientes:

#### **4.1.29 Numba**

No necesariamente es un algoritmo de redes neuronales, pero es un algoritmo muy fácil de usar y optimiza el código de una forma excelente, está enfocado en funciones numéricas y realiza la ejecución de una forma más rápida por medio de la GPU, cabe resaltar que solo optimiza funciones matemáticas. Se encuentra principal mente para las unidades de procesamiento grafico como:

- Intel
- Nvidia
- AMD

#### **4.1.30 TensorFlow**

TensorFlow es una librería de aprendizaje automático de código abierto, con la cual se pueden realizar múltiples tareas en inteligencia artificial, maneja redes neuronales con la capacidad de clasificar imágenes, reconocer patrones escritos por medio de símbolos o dibujos hechos a mano, genera soluciones a problemas de IA.

- Compilación sencilla para entrenar modelos con facilidad mediante API intuitivas con lo que permite iteraciones con modelos realizados en él.

Se encuentra disponible para Windows, Linux, MacOS y plataformas móviles incluidas Android y iOS lo que hace su mayor interacción y facilidad con las plataformas más utilizadas hoy en día, aparte de ello puede ser ejecutada tanto en la CPU como en la GPU.

En pocas palabras está diseñada para crear redes neuronales con la capacidad para detectar y descifrar patrones y correlacionarlos entre sí, es uno de los algoritmos más utilizados para la creación de redes neuronales.

#### **4.1.31 Keras**

Es una biblioteca de redes neuronales desarrollada en Python, tiene la capacidad de ejecutarse sobre TensorFlow (Team, 2021), se centra en el uso amigable con el usuario, esta librería implementa bloques de redes neuronales como son, layers, funciones objetivo, funciones de activación, optimizadores matemáticos, por lo que lo hace uno de los algoritmos más seleccionados para redes neuronales.

Esta Librería ofrece soporte para redes neuronales Convolucionadas y Redes neuronales recurrentes, también tiene la función de entrenamiento distribuido de modelos de aprendizaje profundo utilizando clusters de la la GPU de un computador donde se está compilando.

#### **4.1.32 OpenCv**

Es una biblioteca de visión artificial desarrollada por Intel, lo que significa Open Computer Vision (Visión Artificial Abierta). Fue desarrollada en 1999, su impacto para visión artificial ha sido tanta que a la fecha se menciona como la biblioteca más popular y efectiva, entre sus usos más usados son: (Team, 2021)

- Detección de movimiento
- Reconocimiento de objetos

- Reconstrucción de imágenes
- Biblioteca publica
- Multiplataforma (GNU/Linux, Windows, MacOS, Android, iOS)
- Documentación con mucha explicación

La biblioteca es de uso libre tanto para propósitos comerciales como de investigación, es por ello que es una de las más nombradas, el área de aplicación es bastante y algunas de ellas son: (Team, 2021)

- Características 2D y 3D
- Estimación de pose de cámara
- Reconocimiento facial
- Reconocimiento de gestos
- Comprensión de movimientos
- Reconocimiento de objetos

Esta biblioteca está desarrollada en el lenguaje de programación C++ orientada a objetos, pero tiene extensiones o conectores para otros lenguajes además cuenta con documentación y tutoriales:

- Python
- Java
- C++



### 4.1.33 Pytorch

Existen varias Librerías relacionadas con el aprendizaje automático y redes neuronales; TensorFlow fue desarrollada por Google y es la más extendida y seleccionada actualmente, esta librería está creciendo rápidamente por varios factores, por la facilidad de uso y ejecución, siendo una de sus principales ventajas de Pytorch, el uso de la GPU, lo que permite optimizar y entrenar procesos que, por lo general, son demasiado lentos. (Team, 2021)

Una red neuronal es un sistema de nodos interconectados de forma ordenada, su función es obtener una señal, se distribuye entre sus nodos y produce una salida haciendo que procese esa información y vaya aprendiendo de la misma.

Creada por el equipo de investigación de Facebook de código abierto, cuenta con una interfaz mucho más fácil y pulida en Python, haciéndola una de las librerías fuertes para la creación de modelos de redes neuronales; Pytorch presenta dos funciones de alto nivel:

- Computación tensorial con la Librería Numpy lo que hace una aceleración del código por medio de la GPU
- Creación de redes neuronales profundas, con un sistema de diferenciación basado en cintas

El proyecto aquí presentado tiene la posibilidad de usar múltiples algoritmos que ayuden a materializarlo, es por ello por lo que los algoritmos usados sobre el lenguaje Python serán OpenCV y TensorFlow, para realizar las redes neuronales.

## 5 Marco Conceptual

El Proyecto consta de múltiples partes importantes para su desarrollo, es por ello por lo que se resaltarán las dos más importantes para su implementación.

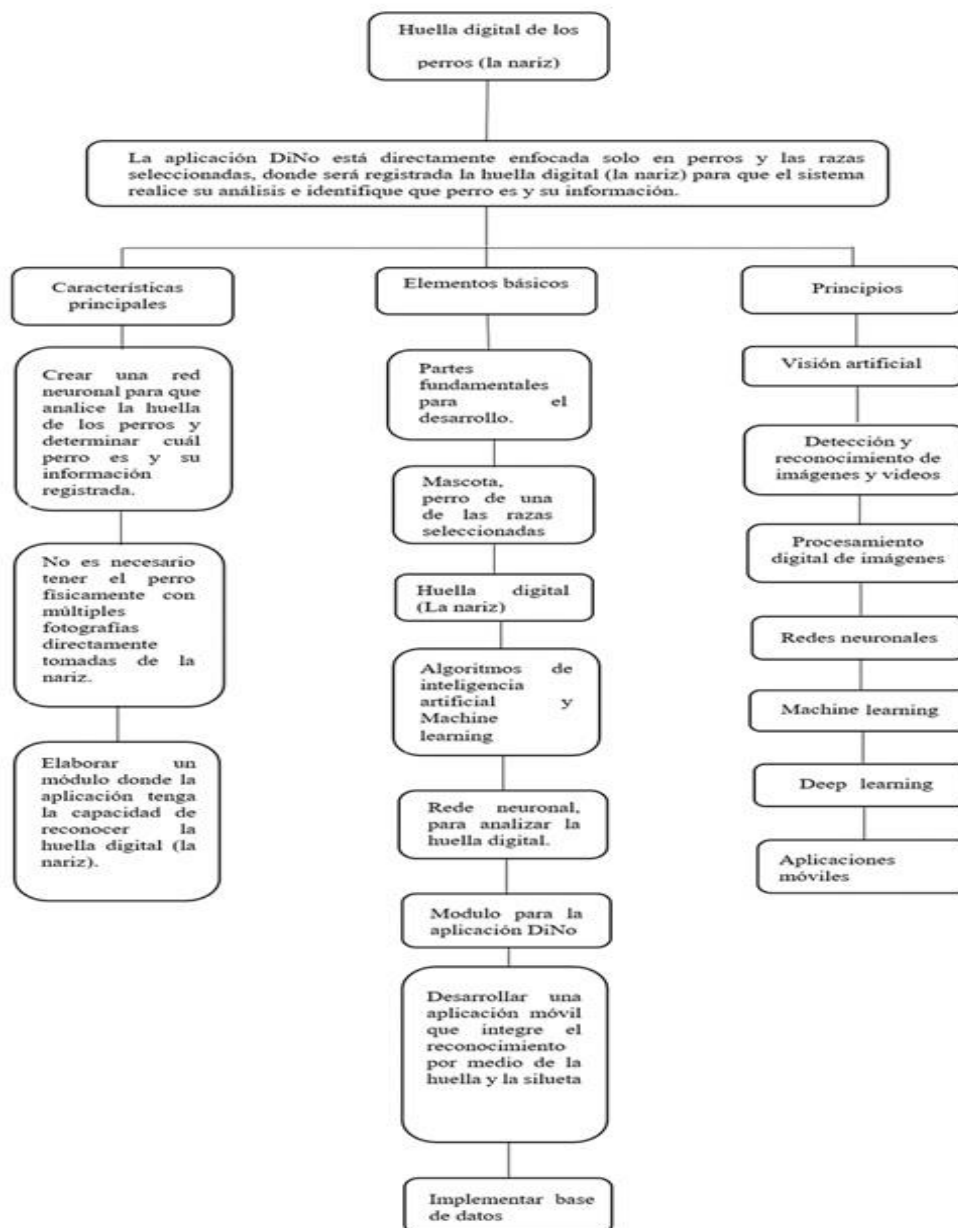
**Huella digital del perro (La nariz)**

La huella digital de un perro no esta ubicada en su pata, esta se encuentra ubicada en su nariz. No hay dos perros con la misma nariz, asi como a los humanos nos identifican por nuestra huella dactilar, lo perros pueden ser identificados por su nariz, con este metodo se pretende desarrollar un sistema inteligente que permita el registro de los perros para que posteriormente se pueda identificar este y determinar su dueño.

Dino tiene la capacidad de reconocer la huella digital del perro por medio de una fotografia de la nariz del peludo:

## Imagen19

### Marco conceptual huella digital del perro



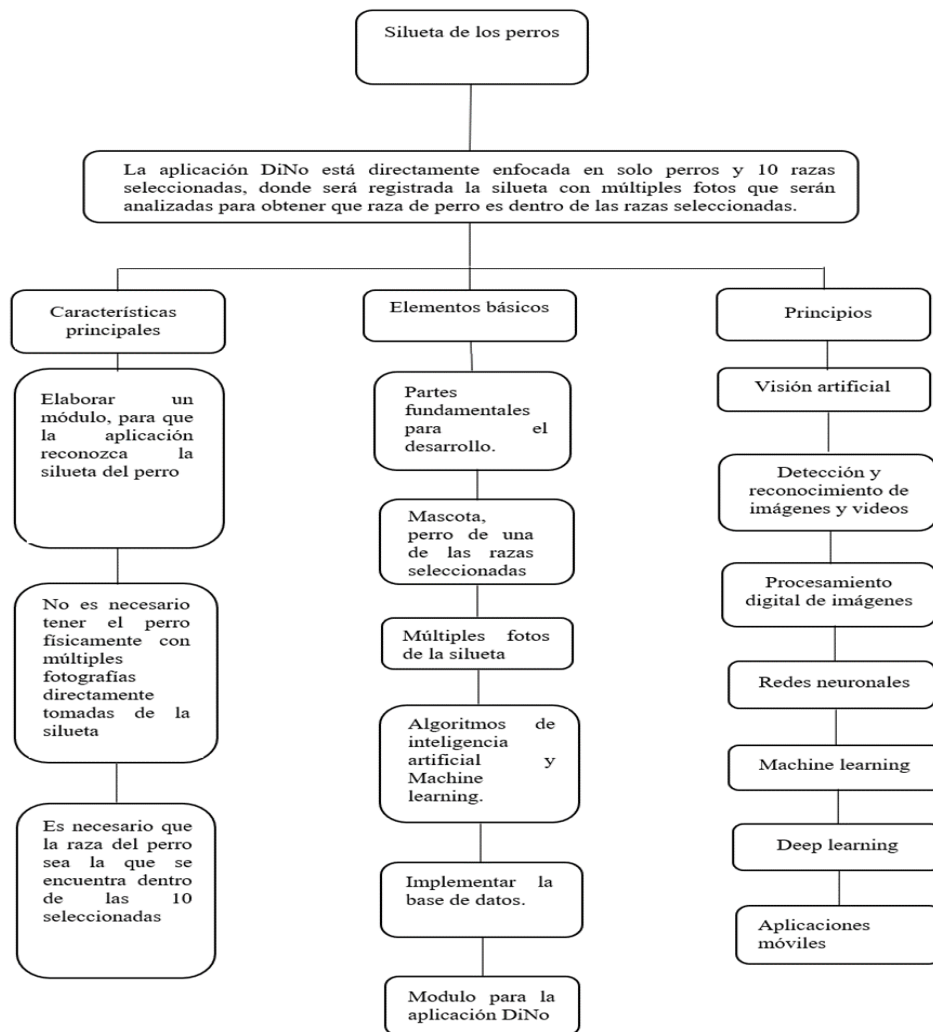
*Nota.* La imagen representa los conceptos sobre huella digital utilizados dentro del texto.

Creación propia, 2021

## Silueta del perro

### Imagen 20

#### Marco conceptual silueta del perro



*Nota.* La imagen representa los conceptos sobre silueta del perro utilizados dentro del texto.

Creación propia, 2021

## **6 Marco legal**

### **Protección de datos personales**

Hoy en día se habla mucho de las leyes que regulan el uso de datos personales, las implicaciones que conllevan contenerlos, la seguridad y los riesgos de no usar los datos de una forma ética, es por ello que el gobierno y a nivel mundial se han creado distintos decretos y leyes que regulan el uso de datos personales, en este caso se resaltarán los artículos más importantes sobre la ley estatutaria 1581 de 18 octubre 2012 donde el Congreso de la República de Colombia en sus funciones, dictan disposición para la protección de datos personales .

#### **Título 1, Objeto, ámbito de aplicación y definiciones. (Republica, 2012)**

La presente ley tiene como objeto desarrollar el derecho que tienen todas las personas a conocer, actualizar y rectificar la información que se haya recogido sobre ellas, en bases de datos. Ámbito de aplicación, la presente ley aplicara para el tratamiento de datos personales, efectuados en territorio nacional o cuando el responsable del tratamiento no este establecido en el territorio nacional le sea aplicable la legislación colombiana en virtud de las normas internacionales. Los principios sobre protección de datos serán aplicables en todas las bases de datos que tienen características de estar amparadas por reserva legal. Definiciones, se entiende por conceptos previamente establecidos, como lo son autorización , consentimiento del titular para llevar a cabo el tratamiento de sus datos, base de datos, conjunto organizado de datos, Dato personal, información vinculada o que pueda asociare a una o varias personas, encargado del tratamiento, persona natural o jurídica que por sí misma o en asociación realice tratamiento de los datos, responsable del tratamiento es aquel persona natural o jurídica que decide sobre el tratamiento de los datos y de las bases de datos, titular, persona natural cuyos datos sean objeto de tratamiento,

Tratamiento, cualquier operación tales como recolección, almacenamiento, uso, circulación o supresión.

## **Título 2. Principios rectores**

Principios para el tratamiento de datos, en funciones de la presente ley se requiera e implementara de manera consecutiva y de forma que se requiera los siguientes principios: (Republica, 2012)

La presente ley tiene los siguientes principios los cuales, constan del tratamiento y mantenimiento de la integridad de los datos de cada uno de los titulares, tales como, finalidad que se entiende como por qué se obtienen los datos, libertad, solo puede ejercerse con un consentimiento previo y expreso al titular de los datos, veraz o de calidad donde el tratamiento de los datos debe garantizar que la información este completa y actualizada, transparencia, debe garantizarse el derecho del titular a obtener del responsable del tratamiento información acerca de la existencia de datos del mismo.

Acceso y circulación restringida, el tratamiento este sujeto a los límites que se derivan de la naturaleza de los datos personales, por medio de un consentimiento, salvo que la información no podrá estar disponible en internet u otros medios de divulgación o comunicación masiva.

Principio de seguridad y confidencialidad, la información sujeta a tratamiento de datos por el responsable del tratamiento, se deberá manejar con las medidas técnicas, humanas y administrativas para prestar seguridad, perdida, consulta, adulteración o uso fraudulento, todas las personas que intervengan en el tratamiento de datos personales que no tengan naturaleza de públicos están obligadas a garantizar la reserva de la información, inclusive después de finalizada su relación con alguna de las labores que comprenden el tratamiento.

### **Título 3, Categorías especiales de datos**

Datos sensibles, El propósito de la siguiente ley se entiende por datos sensibles aquellos que afectan la intimidad del titular como lo son, datos étnicos, datos que generen discriminación, datos sobre la vida sexual, datos relativos a la salud, convicciones religiosas, orientación política, organizaciones sociales y datos biométricos o de identificación como lo es el caso de la cédula de ciudadanía.

Tratamiento de datos sensibles. Se prohíbe el tratamiento de datos sensibles excepto cuando:  
(Republica, 2012)

- El Titular haya dado su autorización explícita a dicho Tratamiento.
- El Tratamiento sea necesario para salvaguardar el interés vital del Titular.
- El Tratamiento sea efectuado en el curso de las actividades legítimas y con las debidas garantías por parte de una fundación, ONG, asociación o cualquier otro organismo sin ánimo de lucro.
- El Tratamiento se refiera a datos que sean necesarios para el reconocimiento, ejercicio o defensa de un derecho en un proceso judicial;
- El Tratamiento tenga una finalidad histórica, estadística o científica. En este evento deberán adoptarse las medidas conducentes a la supresión de identidad de los Titulares.

Derechos de los niños niñas y adolescentes, el tratamiento de datos se asegurará del respeto de los derechos de los niños, niñas y adolescentes, lo que quiere decir que se debe cumplir y reglamentar que la información solicitada a niños, niñas y adolescentes debe ser protegida de forma más segura excepto de datos que sean de naturaleza pública.

#### **Título 4, derechos y condiciones de legalidad para el tratamiento de datos**

Derechos de los titulares, se entiende por aquellos derechos que tienen los titulares al suministrar información en una base de datos archivos y demás documentos que tengan que ver con registro de datos personales: (Republica, 2012)

- Conocer, actualizar y rectificar sus datos personales frente a los responsables del Tratamiento o Encargados del Tratamiento.
- Solicitar prueba de la autorización otorgada al responsable del Tratamiento
- Ser informado por el responsable del Tratamiento o el Encargado del Tratamiento, previa solicitud, respecto del uso que les ha dado a sus datos personales;
- Presentar ante la Superintendencia de Industria y Comercio quejas por infracciones a lo dispuesto en la presente ley y las demás normas que la modifiquen, adicionen o complementen;
- Revocar la autorización y/o solicitar la supresión del dato cuando en el Tratamiento no se respeten los principios, derechos y garantías constitucionales y legales.
- Acceder en forma gratuita a sus datos personales que hayan sido objeto de Tratamiento.

Autorización del titular, se entiende por aquel aviso al titular del registro, recolección y utilización de datos personales, por medio de cualquier medio que sea solicitado, debe ser informado claramente y autorizado únicamente del titular.

Casos en los que no es necesaria la autorización del titular, la autorización del titular no será necesaria en los siguientes ítems dispuestos en la presente ley: (Republica, 2012)



- Información requerida por una entidad pública o administrativa en ejercicio de sus funciones legales o por orden judicial;
- Datos de naturaleza pública;
- Casos de urgencia médica o sanitaria;
- Tratamiento de información autorizado por la ley para fines históricos, estadísticos o científicos;
- Datos relacionados con el Registro Civil de las Personas.
- A los Titulares, sus causahabientes o sus representantes legales;
- A las entidades públicas o administrativas en ejercicio de sus funciones legales o por orden judicial;
- A los terceros autorizados por el Titular o por la ley.

### **Titulo 5, Procedimientos**

Los Titulares o sus causahabientes podrán consultar la información personal del Titular que repose en cualquier base de datos, sea esta del sector público o privado. El responsable del Tratamiento o Encargado del Tratamiento deberán suministrar a estos toda la información contenida en el registro individual o que esté vinculada con la identificación del Titular. (Republica, 2012)

Reclamos, se entiende por aquellos reclamos realizados por el titular donde se vulneren sus derechos como titular o encuentre actividad sospechosa sobre el tratamiento de sus datos personales, por medio de una solicitud dirigida al responsable de los tratamientos de datos o el encargado del tratamiento además de la identificación del titular.

## **Título 6, Deberes de los responsables del tratamiento y encargados del tratamiento**

Deberes de los responsables del tratamiento de datos, los responsables del tratamiento de datos están sujetos a cumplir y hacer valer sin perjuicio las siguientes disposiciones dictadas a continuación: (Republica, 2012)

- Garantizar al Titular, en todo tiempo, el pleno y efectivo ejercicio del derecho de hábeas data;
- Solicitar y conservar, en las condiciones previstas en la presente ley, copia de la respectiva autorización otorgada por el Titular;
- Informar debidamente al Titular sobre la finalidad de la recolección y los derechos que le asisten por virtud de la autorización otorgada;
- Conservar la información bajo las condiciones de seguridad necesarias para impedir su adulteración, pérdida, consulta, uso o acceso no autorizado o fraudulento;
- Garantizar que la información que se suministre al Encargado del Tratamiento sea veraz, completa, exacta, actualizada, comprobable y comprensible;
- Actualizar la información, comunicando de forma oportuna al Encargado del Tratamiento, todas las novedades respecto de los datos que previamente le haya suministrado y adoptar las demás medidas necesarias para que la información suministrada a este se mantenga actualizada;
- Rectificar la información cuando sea incorrecta y comunicar lo pertinente al Encargado del Tratamiento;

- Suministrar al Encargado del Tratamiento, según el caso, únicamente datos cuyo Tratamiento esté previamente autorizado de conformidad con lo previsto en la presente ley;
- Exigir al Encargado del Tratamiento en todo momento, el respeto a las condiciones de seguridad y privacidad de la información del Titular;
- Tramitar las consultas y reclamos formulados en los términos señalados en la presente ley;
- Adoptar un manual interno de políticas y procedimientos para garantizar el adecuado cumplimiento de la presente ley y en especial, para la atención de consultas y reclamos;
- Informar al Encargado del Tratamiento cuando determinada información se encuentra en discusión por parte del Titular, una vez se haya presentado la reclamación y no haya finalizado el trámite respectivo;
- Informar a solicitud del Titular sobre el uso dado a sus datos;
- Informar a la autoridad de protección de datos cuando se presenten violaciones a los códigos de seguridad y existan riesgos en la administración de la información de los Titulares.
- Cumplir las instrucciones y requerimientos que imparta la Superintendencia de Industria y Comercio.

Deberes de los encargados del tratamiento, los encargados del tratamiento de datos están sujetos a cumplir y hacer valer sin perjuicio las siguientes disposiciones dictadas a continuación:  
(Republica, 2012)

- Garantizar al Titular, en todo tiempo, el pleno y efectivo ejercicio del derecho de hábeas data;
- Conservar la información bajo las condiciones de seguridad necesarias para impedir su adulteración, pérdida, consulta, uso o acceso no autorizado o fraudulento;
- Realizar oportunamente la actualización, rectificación o supresión de los datos en los términos de la presente ley;
- Actualizar la información reportada por los responsables del Tratamiento dentro de los cinco (5) días hábiles contados a partir de su recibo;
- Tramitar las consultas y los reclamos formulados por los Titulares en los términos señalados en la presente ley;
- Adoptar un manual interno de políticas y procedimientos para garantizar el adecuado cumplimiento de la presente ley y, en especial, para la atención de consultas y reclamos por parte de los Titulares;
- Registrar en la base de datos las leyendas “reclamo en trámite” en la forma en que se regula en la presente ley;
- Insertar en la base de datos la leyenda “información en discusión judicial” una vez notificado por parte de la autoridad competente sobre procesos judiciales relacionados con la calidad del dato personal;
- Abstenerse de circular información que esté siendo controvertida por el Titular y cuyo bloqueo haya sido ordenado por la Superintendencia de Industria y Comercio;
- Permitir el acceso a la información únicamente a las personas que pueden tener acceso a ella;

- Informar a la Superintendencia de Industria y Comercio cuando se presenten violaciones a los códigos de seguridad y existan riesgos en la administración de la información de los Titulares;
- Cumplir las instrucciones y requerimientos que imparta la Superintendencia de Industria y Comercio.

### **Derechos de los animales**

En Colombia en las últimas décadas se han comenzado a fortalecer los derechos de los animales y la protección de estos, dando así castigos o condenas a los que abusan y maltratan a los animales, es por ello por lo que se destaca la Ley 1774 de enero de 2016, la Ley 84 de 19989 y el código de procedimiento penal abriendo así la posibilidad de que cualquier especie esté sujeta a derechos.

Las leyes expuestas anteriormente decretaron que los animales no son cosas ni objetos y que por el contrario son animales sintientes por lo que se instauraron sanciones para aquellos que maltraten vulneren y abusen de los animales en diferentes contextos, además de que los animales en territorio colombiano tienen derecho a su protección, bienestar y solidaridad.

Declaración universal de los derechos de los animales dictados en los siguientes artículos:

**Artículo No 1**, Absolutamente todos los animales nacen iguales ante la vida y tienen los mismos derechos a la existencia.

**Artículo No 2**, principios básicos: (Colombia, 2016)

- Todo animal tiene derecho al respeto.

- El hombre, en tanto que especie animal, no puede atribuirse el derecho de exterminar a los otros animales o de explotarlos, violando ese derecho. Tiene la obligación de poner sus conocimientos al servicio de los animales.
- Todos los animales tienen derecho a la atención, a los cuidados y a la protección del hombre.

**Artículo No 3**, Actos crueles, ningún animal será sometido a actos crueles ni vejámenes que vulneren su integridad, si es necesaria la muerte del animal será instantánea, indolora, no generadora de angustia ni mucho menos sin los permisos establecidos ni las justificaciones de dicho acto.

**Artículo No 4**, Libertad de los animales, Todo animal pertenece a naturaleza salvaje, tiene derecho a vivir en su propio entorno ya sea, terrestre, aéreo o acuático, además de tener la posibilidad de reproducirse.

**Artículo No 5**, todo animal de entorno perteneciente al hombre tiene el derecho de vivir y crecer a ritmo en las condiciones en las que vive, lo que quiere decir que donde se realice una modificación en su condición de vida es un fin contradictorio a dicho derecho.

**Artículo No 6**, Todo animal que haya sido escogido por un hombre para ser su compañero tiene derecho a vivir conforme a su longevidad, un animal que sea abandonado es un acto cruel y degradante que conlleva a causas penales.

**Artículo No 7**, Todo animal que sea sometido a trabajo o algún acto que gente esfuerzo debe tener y sin justificación, alimento, tratamiento en caso de enfermedad y descansos, además de no ser forzado a trabajar más tiempo de lo normal.

**Artículo No 8**, experimentar con animales que implique un sufrimiento físico o psicológico en el entorno experimental de médicos, científicos y comerciales, es una forma contradictoria de esta ley y se debe recurrir a técnicas alternativas utilizadas y desarrolladas.

**Artículo No 9**, Todo animal que fue criado para la alimentación, debe ser y tiene el derecho de ser nutrido, cuidado, y transportado, así como el sacrificio indoloro y que no represente un motivo de ansiedad y sacrificio indebido.

**Artículo No 10**, Ningún animal debe ser explotado para el esparcimiento del ser humano, además las exhibiciones que dispongan de animales son contradicciones a la presente ley y representa daño a la dignidad de los animales.

**Artículo No 11**, Toda acción que represente con la muerte de un animal es un biocidio, es decir que implica y atenta con la vida del animal, sin necesidad ni justa causa.

**Artículo No 12**, toda acción que implique con la muerte de un gran número de animales se le conoce como genocidio, además de la contaminación y destrucción del ambiente del animal, aquellos vejámenes perpetrados tendrán consecuencias penales.

**Artículo No 13**, un animal sin vida merece respeto, no pueden ser expuestas dichas escenas de violencia, ni en películas filmaciones, cine y televisión, exceptuando si el fin es dar a entender las acciones que atentan con la vida del animal.

**Artículo No 14**, Los organismos de protección de los animales debe salvaguardar los derechos y ser representados a nivel gubernamental de igual forma que ha sido con los derechos del hombre.  
(Colombia, 2016)

## **7 Diseño Metodológico**

### **7.1 Tipo de investigación**

El desarrollo de este proyecto se enfocará en seguir los lineamientos de la investigación aplicada. (Muntané Relat, 2010) la define como “*Una investigación que se caracteriza porque busca la aplicación o utilización de los conocimientos que se adquieren*”, adicionalmente cabe agregar que este tipo de investigación requiere también que se use la investigación básica ya que se necesita un marco teórico como base, aunque el objetivo principal son las consecuencias prácticas.

Por otro lado, la clase de medios que se van a usar para obtener la información es la documental, ya que esta se basa en la búsqueda de información presente tanto en bases de datos como en documentos elaborados por otros autores. Por otro lado, se utiliza la investigación aplicada, debido a que el resultado de esta investigación es un software destinado al uso comunitario, que permita a las personas tener un registro de sus mascotas y que se sientan más tranquilas de que su mascota está plenamente registrada y que es caso de un hecho adverso pueden identificarla.

### **7.2 Fases del Proyecto**

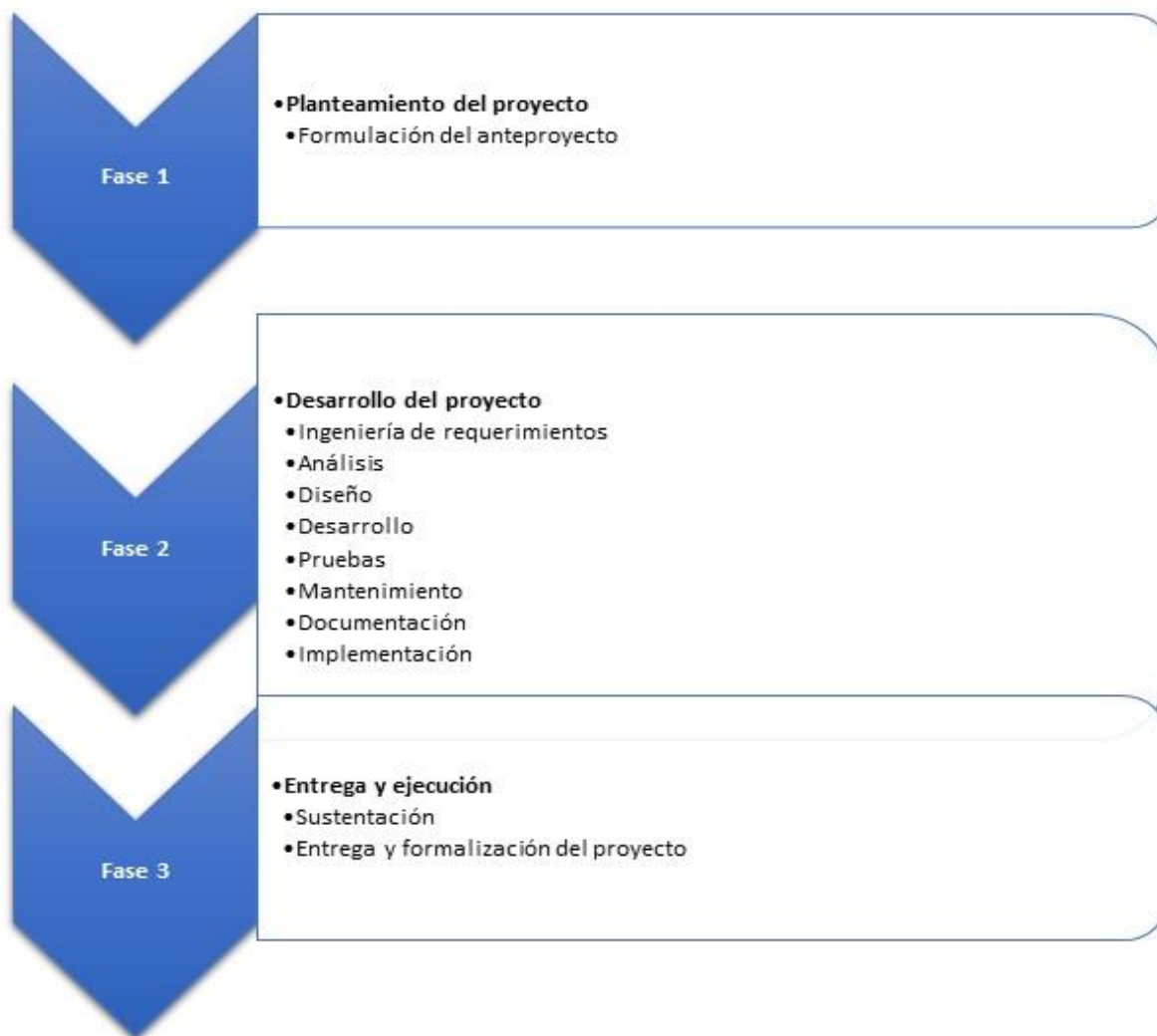
El desarrollo del proyecto se definió en 3 fases. La fase 1 contiene la parte inicial de la investigación, búsqueda de fuentes, referencias, análisis del mercado, etc. En la fase 2 en base a las fuentes y documentación obtenida se procede al desarrollo del proyecto evaluado en cada una de sus etapas. Por último, en la fase 3, última fase, se realiza la sustentación y entrega formal del proyecto ante la universidad representada por los jurados escogidos.



El Proyecto cuenta con 3 fases importantes como se muestra en la siguiente imagen.

### Imagen21

#### *Fases del proyecto*



*Nota.* La imagen representa las fases definidas para el proyecto. Creación propia, 2021.

### 7.3 Procedimientos, herramientas y técnicas para recolectar la información

Para la recolección de información, se realizó una encuesta masiva elaborando un formulario utilizando la herramienta de Google Forms, esta encuesta se estructuró con 12 preguntas con temas sobre el manejo, uso y problemas frecuentes que se presentan con los perros, las preguntas se

dirigieron a buscar información relevante sobre cómo piensa una persona respecto al cuidado y seguridad de su mascota, el objetivo fue que mínimo 30 personas contestaran la encuesta. El enlace de la encuesta fue enviado por la aplicación WhatsApp con el consentimiento de la persona que la deseaba diligenciar, la encuesta se dirigió a las personas que contaran con una mascota (perro).

La herramienta forms se utilizó elaborar el formulario porque brinda información útil acerca de las respuestas de la encuesta, como son: la generación de estadísticas, gráficos, tendencias, además brinda la posibilidad de exportar la información a un archivo Excel. Esta información fue de gran importancia para la toma y levantamiento de requerimientos y necesidades del usuario.

## **8 Desarrollo del proyecto**

En el presente Proyecto se desarrolló un sistema que permite reconocer e identificar la huella de un perro y su silueta por medio de una imagen y la metodología ágil que se utilizó para desarrollar el proyecto fue SCRUM, vista esta metodología como un esfuerzo de colaboración para crear un nuevo producto, servicio u otro resultado. Los proyectos se ven afectados por limitaciones de tiempo, costos, alcance, calidad, recursos, capacidades organizacionales y demás limitaciones que dificultan su planificación, ejecución, administración y, por último, su éxito. Sin embargo, la exitosa implementación de los resultados de un proyecto terminado ofrece ventajas económicas considerables a una organización.

SCRUM es uno de los métodos ágiles más populares. Es un framework adaptable, iterativo, rápido, flexible y eficaz, diseñado para ofrecer un valor considerable en forma rápida a lo largo del proyecto. SCRUM garantiza transparencia en la comunicación y crea un ambiente de responsabilidad colectiva y de mejora continua.

Una fortaleza clave de SCRUM radica en el uso de equipos interfuncionales, auto organizados y empoderados que dividen su trabajo en ciclos de trabajos cortos y concentrados llamados *sprints*. En la siguiente figura se muestra la visión general de flujo de un proyecto SCRUM.

## Imagen22

### *Flujo de SCRUM para un sprint*



*Nota.* Imagen tomada de flujo en SCRUM. (SCRUMstudy, 2017)

A continuación, se detallan cada una de las fases que se desarrollaron a través de la metodología ágil SCRUM.

### **Inicio Fase 1**

La fase 1 del Proyecto se basa en el planteamiento y ataque de la problemática planteada inicialmente, es por ello, que se empieza con la metodología scrum sobre cada una de las épicas.

#### **8.1.1 Visión del proyecto**

Se plantea ser la primera aplicación móvil colombiana que cuente con estas características únicas y diferenciadoras, aportando así al mercado algo novedoso. Se desea llegar a la gran mayoría de

usuarios que poseen una mascota en su hogar para aportar un granito de arena, y como último y no menos importante lograr monetizar el proyecto.

### 8.1.2 Desarrollo de Épicas

Las épicas son funcionalidades de alto nivel o descripción del producto realizada por el usuario.

#### 8.1.2.1 Épicas

*Tabla 4*

#### *Historia de usuario*

---

##### **Historia de usuario 1**

---

##### **Definición:**

Como usuario, quiero tener la posibilidad de registrarme en la aplicación para tener mi información registrada y ingresar entrar posteriormente.

##### **Criterios de aceptación:**

- Permitir registro del usuario
  - Tiempo de respuesta rápido
  - Posibilidad de recuperar su cuenta en caso de olvidarla
- 

*Tabla 5*

#### *Historia de usuario*

---

##### **Historia de usuario 2**

---

##### **Definición:**

Como usuario, quiero ingresar la información de mi perro para tener un registro

##### **Criterios de aceptación:**

---

- 
- Formulario fácil de entender
  - Formulario fácil de diligenciar
  - Tiempo de respuesta rápido
  - Permitir el registro exitosamente
  - Permitir actualizar los datos previamente registrados
- 

*Tabla 6*

*Historia de usuario*

---

**Historia de usuario 3**

---

**Definición:**

Como usuario, quiero registrar la huella de mi perro para reconocerlo en caso de perdida

**Criterios de aceptación:**

- Reconocimiento de la huella eficiente y acertado
  - Interfaz de registro de la huella fácil de entender e intuitivo
  - Tiempo de respuesta rápido
- 

*Tabla 7*

*Historia de usuario*

---

**Historia de usuario 4**

---

**Definición:**

Como usuario, quiero que la aplicación me brinde la posibilidad de reconocer la raza de los perros para divertirme y aprender sobre razas

**Criterios de aceptación:**

- Reconocimiento de la raza eficiente y acertado
  - Interfaz de reconocimiento de la raza fácil de entender e intuitivo
  - Tiempo de respuesta rápido
-

Tabla 8

*Historio de usuario*


---

**Historia de usuario 5**

---

**Definición:**

Como usuario, quiero aprender a usar las funcionalidades de la aplicación para sacarle su mayor provecho

**Criterios de aceptación:**

- Instructivo o consejos para que el usuario aprenda a usar la aplicación
- Interfaces amigables con el usuario (fáciles de entender, intuitivas)
- Tiempo de respuesta rápido
- Tener soporte con el grupo de desarrollo si tiene preguntas

---

**8.1.3 Creación del backlog priorizado del producto****8.1.3.1 Backlog priorizado del producto**

El Backlog Priorizado del Producto contiene una lista priorizada de los requerimientos y de las funcionalidades descritos en las épicas, que son historias de usuario de alto nivel. El Backlog Priorizado del Producto se basa en tres factores principales: valor, riesgo o incertidumbre y dependencias. Este se definió de la siguiente manera:

Tabla 9

*Backlog priorizado del producto*

<b>Priorización</b>	<b>Valor</b>	<b>Riesgo</b>	<b>Dependencia</b>
Historia de usuario 3	Alto	Alto	Medio

---

Historia de usuario 4	Alto	Alto	Medio
Historia de usuario 1	Alto	Medio	Bajo
Historia de usuario 2	Alto	Medio	Bajo
Historia de usuario 5	Medio	Bajo	Bajo

### 8.1.3.2 Criterios de terminado

Tabla 10

#### *Historia de usuario*

---

#### **Historia de usuario 1**

---

##### **Definición:**

Como usuario, quiero tener la posibilidad de registrarme en la aplicación para tener mi información registrada y poder ingresar posteriormente.

##### **Criterios de terminado:**

- Las expectativas de usabilidad son las esperadas
  - Los tiempos de respuesta son aceptables
  - El diseño es aprobado por su facilidad de diligenciamiento
- 

Tabla 11

#### *Historia de usuario*

---

#### **Historia de usuario 2**

---

##### **Definición:**

Como usuario, quiero ingresar la información de mi perro para tener un registro

---

---

**Criterios de terminado:**

- Las expectativas de usabilidad son las esperadas
  - Los tiempos de respuesta son aceptables
  - El diseño es aprobado por su facilidad de diligenciamiento
- 

*Tabla 12**Historia de usuario*

---

**Historia de usuario 3****Definición:**

Como usuario, quiero registrar la huella de mi perro para reconocerlo en caso de perdida

**Criterios de terminado:**

- El algoritmo de reconocimiento de la huella es aprobado debido a su porcentaje de éxito en la identificación y reconocimiento de la huella
  - Tiempo de respuesta dentro de los parámetros normales
- 

*Tabla 13**Historia de usuario*

---

**Historia de usuario 4****Definición:**

Como usuario, quiero que la aplicación me brinde la posibilidad de reconocer la raza de los perros para divertirme y aprender sobre razas

**Criterios de terminado:**

- El algoritmo de reconocimiento de la raza es aprobado debido a su porcentaje de éxito en la identificación y reconocimiento de la raza
  - Tiempo de respuesta dentro de los parámetros normales
-



*Tabla 14****Historia de usuario***

---

**Historia de usuario 5**

---

**Definición:**

Como usuario, quiero aprender a usar las funcionalidades de la aplicación para sacarle su mayor provecho

**Criterios de terminado:**

- El diseño de los consejos o instrucciones para usar la aplicación son claros y entendibles
  - Permitirle al usuario conocer cada funcionalidad de la aplicación de manera dinámica
- 

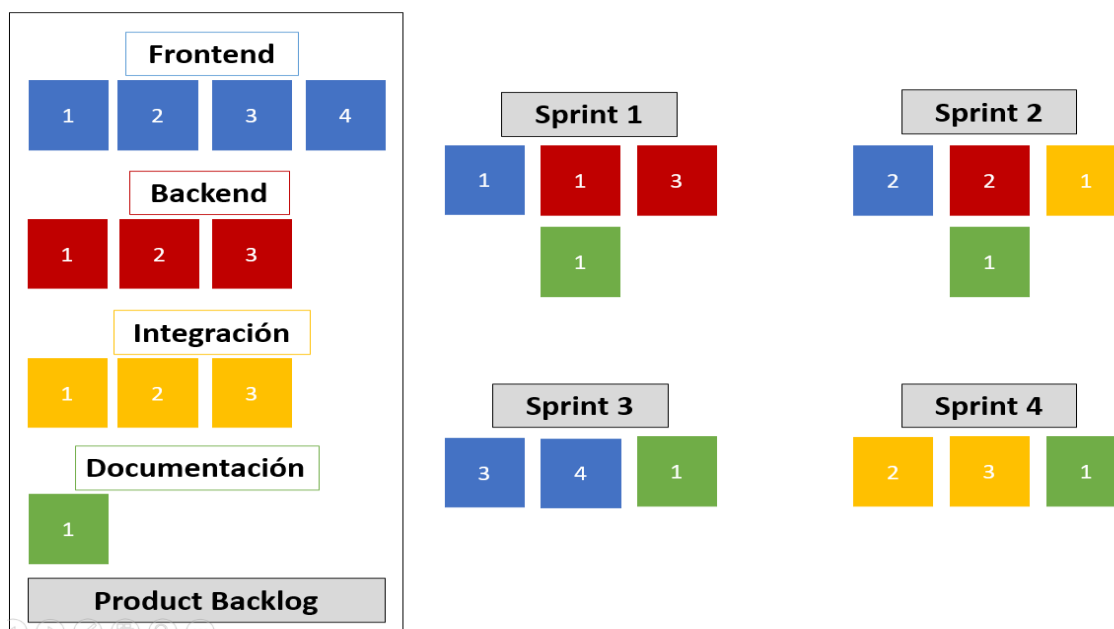
**8.1.4 Planificación del lanzamiento**

El objetivo de la planificación del lanzamiento es desarrollar un plan, este plan define cuándo las distintas funcionalidades útiles serán entregadas al cliente.

**8.1.4.1 Cronograma de planificación del lanzamiento**

### Imagen 23

#### Cronograma de planificación de lanzamiento



*Nota.* Imagen que ilustra el cronograma. Creación propia, 2021

#### 8.1.4.2 Duración del sprint

Con base en las diversas entradas, incluyendo los requerimientos del Proyecto y el cronograma de planificación de lanzamiento, se definieron sprints con dos semanas de duración.

### Planificación y estimación

#### 8.1.5 Identificación y lista de tareas

Para el proyecto de identificaron las siguientes tareas según el cronograma de planificación de lanzamiento, los números anexados al final entre “()” se refieren al identificador de la actividad según el cronograma:

##### 1. Frontend (Desarrollo y funcionalidad):

- Pantalla (interfaz) principal, registro y inicio de sesión. (1)
- Pantalla (interfaz) mascotas. (2)
- Pantalla (interfaz) perfil. (3)
- Pantalla (interfaz) cámara. (4)

**2. Backend (Desarrollo y funcionalidad):**

- Base de datos. (1)
- API Python. (2)
- API .NET. (3)

**3. Integración:**

- Base de datos con las APIS. (1)
- Frontend con la API .NET. (2)
- Frontend con la API Python. (3)

**4. Documentación:**

- Documentar cada una de las actividades realizadas. (1)

A nivel general, se identificaron 11 tareas para el desarrollo y culminación del proyecto.

**8.1.6 Estimación de tareas**

Según el cronograma de planificación de lanzamiento se tiene un total de 8 semanas para culminar, teniendo en cuenta esto, se presentan los siguientes sprint:

**1. Sprint 1:**

- Frontend: Pantalla (interfaz) principal. (1)
- Backend: Base de datos (1) y API .NET. (3)
- Documentación: Documentar cada una de las actividades realizadas. (1)

**2. Sprint 2:**

- Frontend: Pantalla (interfaz) mascotas. (2)
- Backend: API Python. (2)
- Integración: Base de datos con las APIS. (1)
- Documentación: Documentar cada una de las actividades realizadas. (1)

**3. Sprint 3:**

- Frontend: Pantalla (interfaz) perfil. (3) y Pantalla (interfaz) cámara. (4)
- Documentación: Documentar cada una de las actividades realizadas. (1)

**4. Sprint 4:**

- Integración: Frontend con la API .NET. (2) y Frontend con la API Python. (3)

### **8.1.7 Crear entregables**

Crear entregables es el proceso en el cual el Equipo Scrum trabaja en las tareas del Sprint Backlog para crear entregables del sprint.

#### **8.1.7.1 Entregables del sprint**

Los entregables del sprint son los incrementos del producto o los entregables que se terminan al final de cada sprint. En el caso concreto del proyecto se definieron 4 sprints, lo que se traduce en 4 entregables.

### **8.1.8 Envío de entregables**

#### **8.1.8.1 Lanzamientos del producto**

A la hora de modelar sistemas, una de las herramientas más usadas es el lenguaje Unificado de Modelado (UML, del inglés *Unified Modeling Language*) (Booch , Rumbaugh , & Jacobson, 1998). Este estándar es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. Provee un vocabulario y reglas que permiten realizar modelos conceptuales de varios aspectos del sistema, con el objetivo de facilitar la comunicación entre el equipo de desarrollo y el usuario final, quienes son los que especifican las funciones y el comportamiento deseado del mismo.

Con el objetivo de realizar un análisis y documentación del sistema, a continuación, se explicarán las funcionalidades necesarias por medio de UML. Se especifican los requerimientos identificados, y se realizan los diagramas de casos de uso, clases y secuencia respectivamente.

### **Análisis de requerimientos**

Un requerimiento puede ser definido como:

- Una condición o capacidad que un usuario necesita para resolver un problema o lograr un objetivo.
- Una condición o capacidad que debe tener un sistema para satisfacer un contrato, norma, especificación, u otros documentos formales.
- Una representación documentada de una condición o capacidad como las mencionadas anteriormente.

A su vez, estos requerimientos pueden ser diferenciados en *requerimientos funcionales* y *requerimientos no funcionales*. Los primeros, son declaraciones de las funciones del sistema que debe ser capaz de realizar, de cómo debe responder ante entradas particulares; mientras que los segundos, son restricciones del sistema, tales como disponibilidad, mantenimiento, seguridad, capacidad de los dispositivos de entrada/salida, rendimiento (por ejemplo, velocidad y tiempo de respuesta), etc.

De acuerdo las definiciones anteriores, se realiza un análisis de requerimientos con el objetivo de identificar tanto las funcionalidades que se esperan del software como sus limitaciones.

### **8.1.9 Requerimientos funcionales**

El sistema debe ser capaz de:

1. Permitir el registro individual de cada usuario.
2. Contar con autenticación al momento de entrar en la APP.
3. Permitir registrar el perro (s).
4. Consultar la información del propietario del perro.
5. Consultar la información del perro.
6. Permitir la actualización de datos, tanto del perro como del usuario.
7. Disponer de un botón de alerta para notificar extravió del perro.
8. Analizar e identificar la huella digital del perro (nariz) mediante una foto.

9. Analizar e identificar la raza del perro mediante una foto.
10. Permitir consultar información limitada del perro y usuario cuando este sea identificado.

#### **8.1.10 Requerimientos no funcionales**

Como requerimientos no funcionales se especifican los siguientes:

1. El sistema debe tener la capacidad de soportar la conexión de varios usuarios al mismo tiempo.
2. La funcionalidad del sistema debe responder al usuario en menos de 10 segundos.
3. El sistema debe ejecutarse en tiempo real, lo que implica que las operaciones computacionales deben procesarse en un tiempo aceptable
4. Los permisos de acceso al sistema de datos deben ser únicamente otorgados por el administrador.
5. La actualización del sistema y configuraciones solo debe ser gestionado por el administrador del sistema.
6. El sistema debe tener un respaldo o backup de la información tanto de los usuarios como de las mascotas ya registradas.
7. El sistema no mostrará información sensible como datos personales de los usuarios a terceros como lo son, numero de cedula y dirección de residencia.
8. El tiempo de aprendizaje del sistema debe realizarse una vez empieza el registro de las fotografías (silueta y nariz de la mascota).
9. El sistema debe ser de uso fácil y en dado caso con una descripción de lo que puede realizar el usuario ante el sistema.
10. El sistema debe contar con interfaces gráficas robustas.
11. El sistema debe contar con mensaje de error en el caso que la respuesta del usuario no sea correcta.
12. La aplicación debe contar un diseño responsive como técnica de diseño web para la correcta visualización de la información en diferentes dispositivos.
13. El sistema estará desarrollado para el sistema operativo móvil Android.
14. En esta versión, el sistema debe contar con idioma español latino.

15. El sistema debe contar con una disponibilidad para el usuario del 98.9%.
16. El tiempo de indisponibilidad del servicio no debe ser superior a 15 minutos.
17. El inicio y reinicio del sistema no puede ser superior a 10 minutos.

### **Diagramas de casos de uso**

Un caso de uso captura el comportamiento deseado de un sistema en desarrollo, sin tener que detallar como se implementara éste. Es una descripción de un conjunto de secuencias de acciones que el sistema lleva a cabo para producir un resultado de interés para un usuario, reflejado en como deberían interactuar ambas partes.

Un diagrama de casos de uso muestra la relación entre los usuarios que interactúan con el sistema y los casos de uso, modelando los aspectos dinámicos del mismo. A continuación, se describen los casos de uso correspondientes al diagrama de la [Ilustración2](#).

*Tabla 15*

#### ***Caso de uso 1***

---

**Caso de uso:** Registro en la aplicación

---

**Actor:** Usuario no registrado

**Descripción:** Registrar usuario en la aplicación

**Curso normal:**

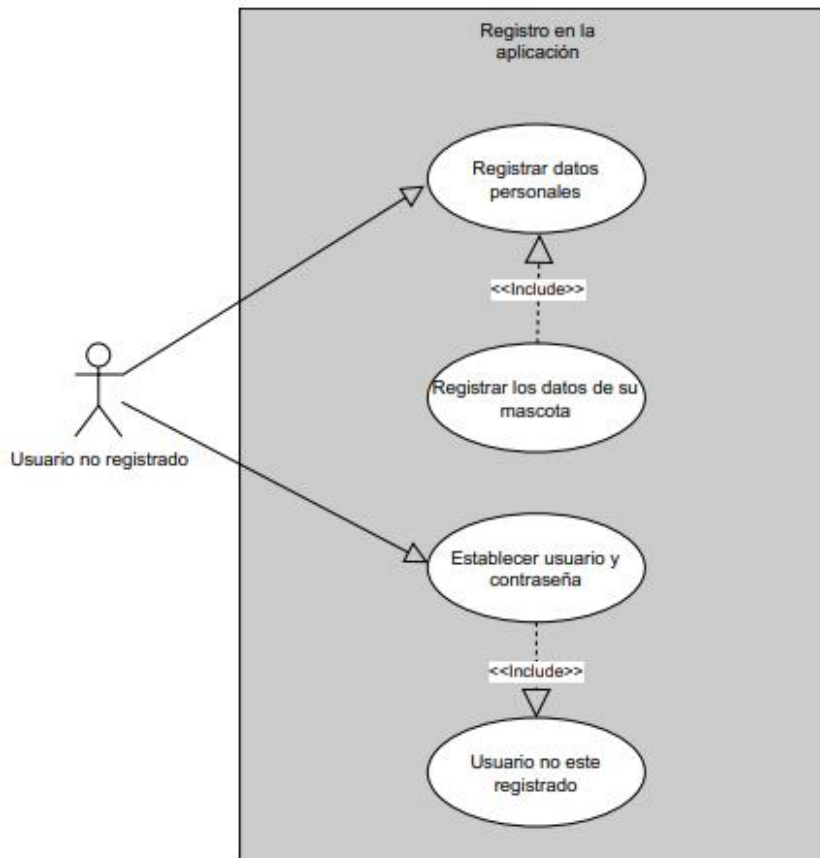
- 1) Inicia cuando el usuario no registrado quiere registrarse, presiona el botón de registrarse
- 2) Diligencia sus datos personales y los de su mascota (perro)
- 3) Establece usuario y contraseña

**Cursos alternativos:**

---

Imagen 24

Diagrama de casos de uso de alto nivel



*Nota.* La imagen ilustra el caso de uso del usuario no registrado. Creación propia, 2021

Tabla 16

**Caso de uso 2**

---

**Caso de uso:** Autenticación en la aplicación

---

**Actor:** usuario registrado o usuario no registrado

---



---

**Descripción:** Proceso de autenticación e inicio en la aplicación

**Curso normal:**

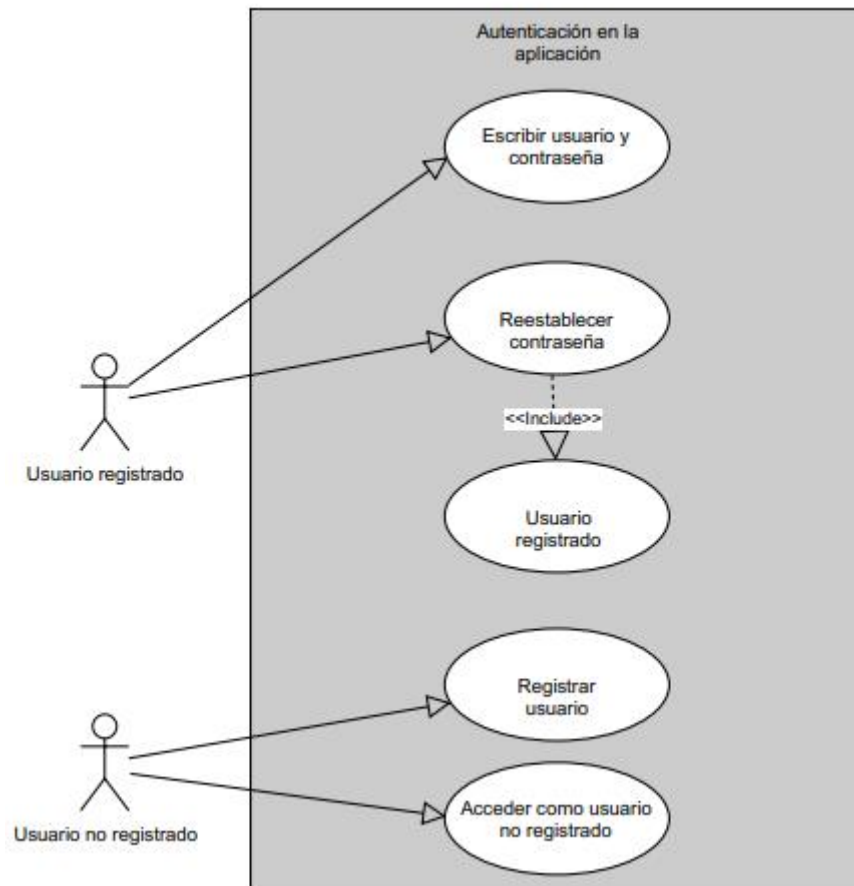
- 1) Inicia cuando el usuario dependiendo del rol que tenga presiona un botón u otro
- 2) Si es un usuario registrado debe escribir su usuario y contraseña en los campos establecidos
- 3) Si es un usuario no registrado debe presionar el botón que dice “acceder como usuario no registrado”

**Cursos alternativos:**

- 1) Si es un usuario no registrado y desea registrarse debe presionar el botón de “registrarse”
-

Imagen 25

Diagrama de casos de uso de alto nivel



*Nota.* La imagen ilustra el caso de uso del usuario registro y no registrado. Creación propia, 2021.

*Tabla 17***Caso de uso 3**

---

**Caso de uso:** Roles en la aplicación

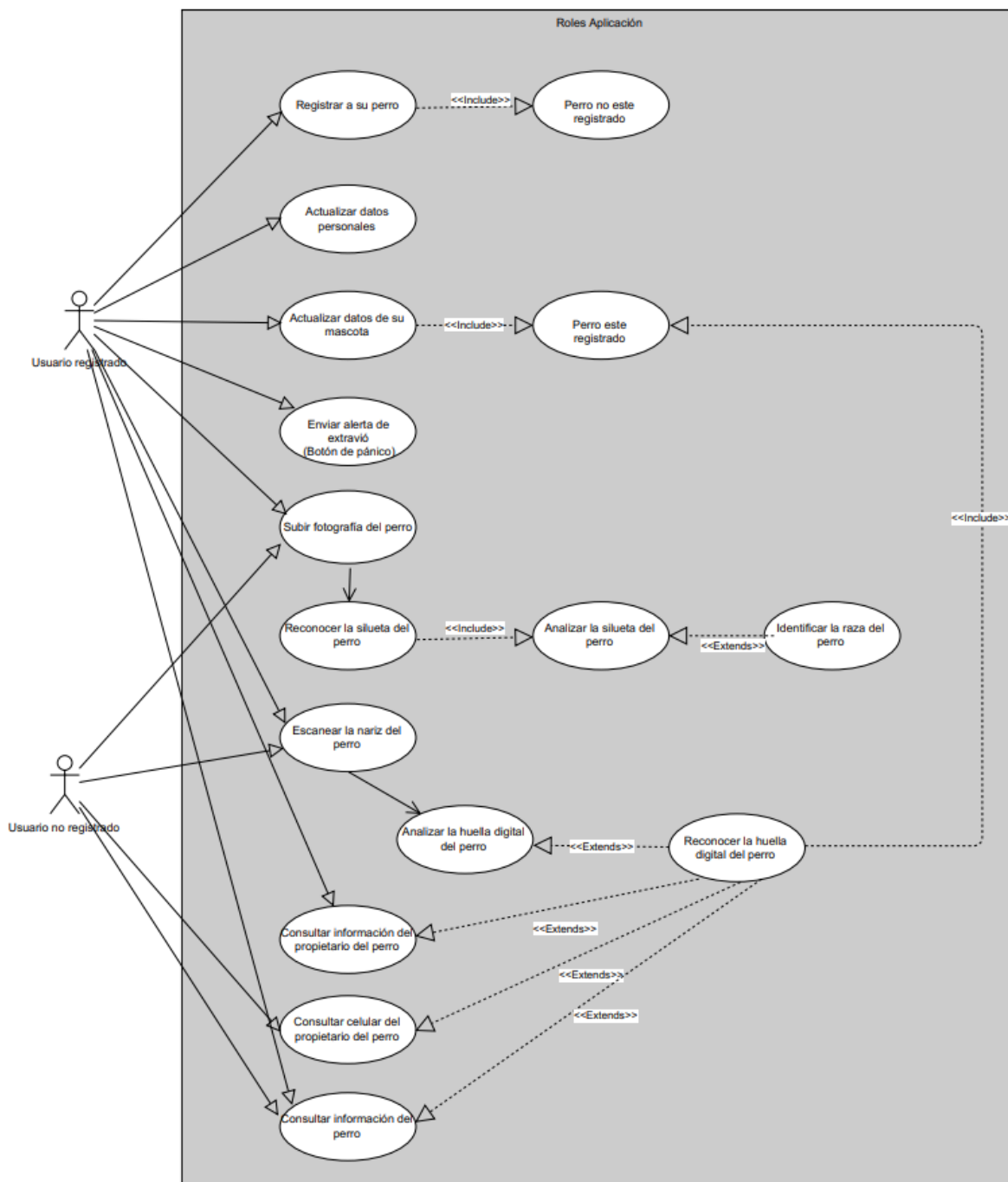
---

**Actor:** Usuario registrado o usuario registrado**Descripción:** Que rol tiene cada perfil y que privilegios o permisos dispone cada uno**Curso normal:****Cursos alternativos:**

---

Imagen 26

Diagrama de casos de uso detallado



*Nota.* Imagen que ilustra el caso de uso detallado del usuario registrado y no registrado. Creación propia, 2021

### **Diagrama de clases**

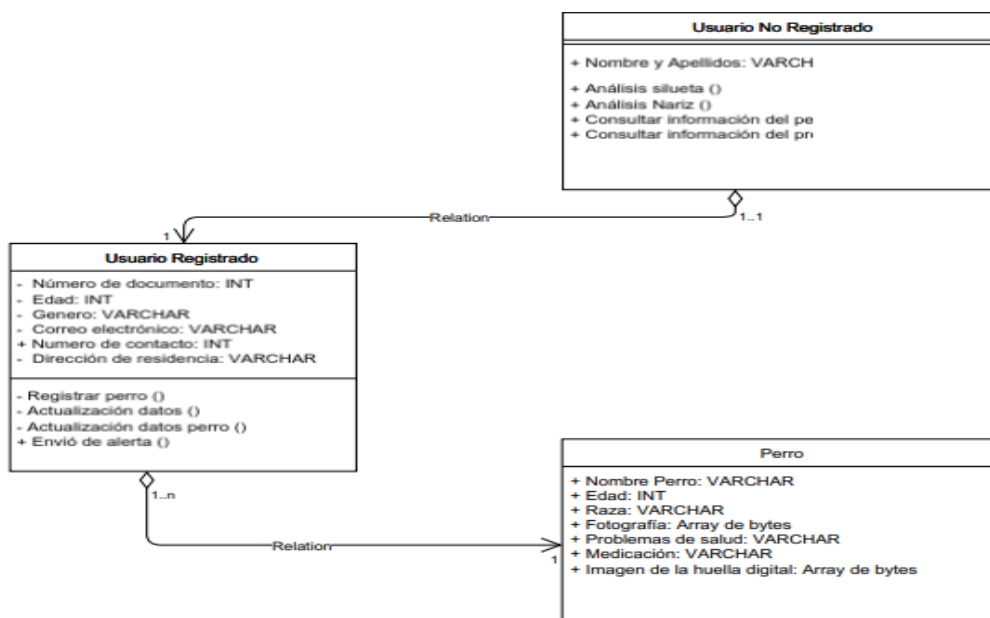
Los diagramas de clases son los más usados en el modelado de sistemas orientados a objetos, los cuales muestran un conjunto de clases y sus relaciones. Se utilizan para modelar la vista del diseño estático de un sistema, lo que implica modelar el vocabulario de este, modelar relaciones o modelar esquemas.

Así como son importantes para visualizar, especificar y documentar modelos estructurales, también lo son para construir sistemas ejecutables, aplicando ingeniería directa e inversa.

En la [Ilustración 5](#) se puede observar el diagrama correspondiente al sistema inteligente reconocedor de perros. Como se puede notar, se definieron distintas clases que permiten realizar tanto la captura de imágenes como el reconocimiento e identificación de un perro.

Imagen 27

### Diagrama de clases



*Nota.* La imagen ilustra el diagrama de clases establecido para el software. Creación propia,

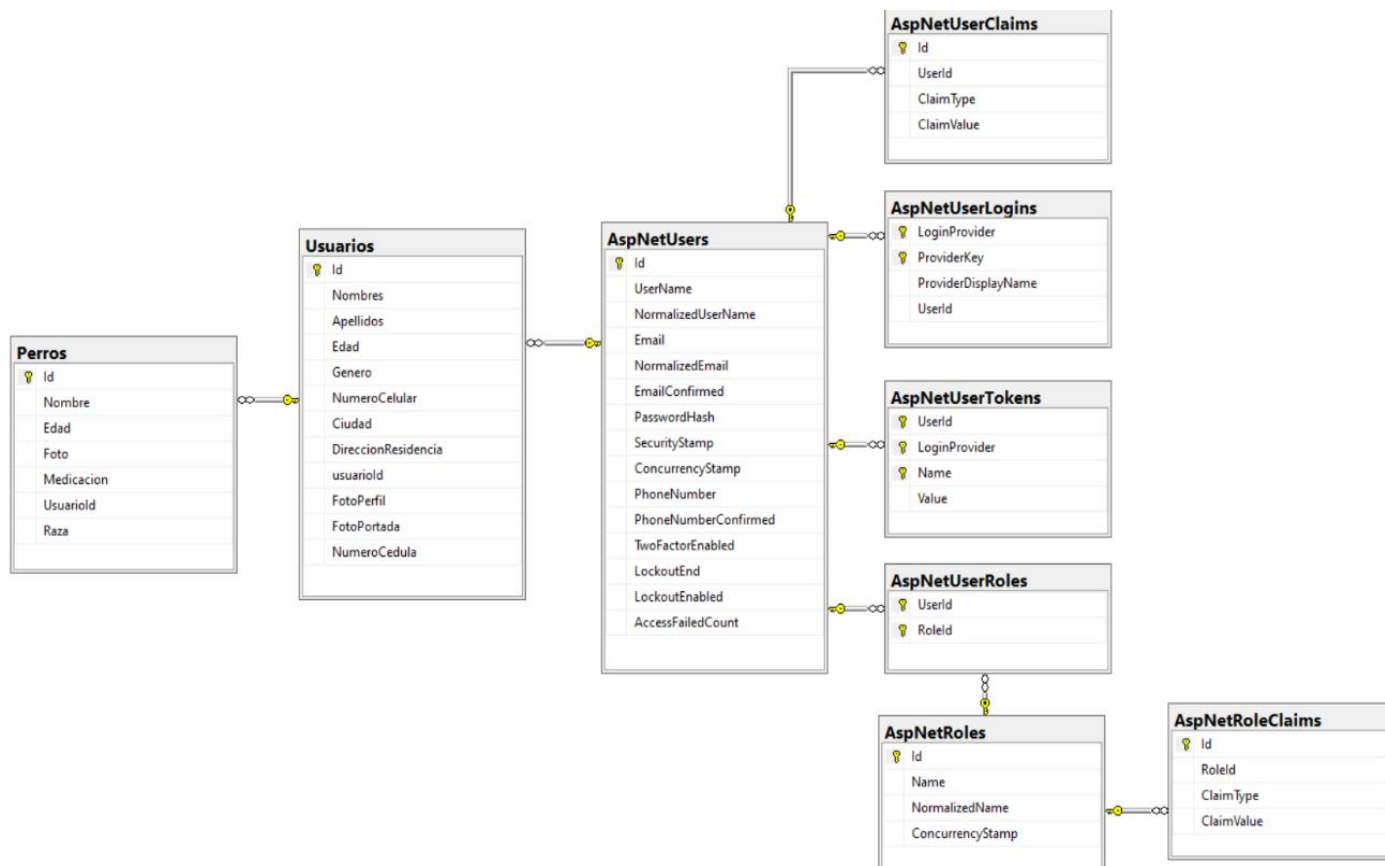
2021

### Diagrama relacional

El diagrama relacional ilustra como las entidades, como personas, objetos o conceptos, se relacionan entre sí dentro de un sistema.

## Imagen 28

### Diagrama relacional



*Nota.* La imagen ilustra el diagrama relacional establecido para el software. Creación propia,

2022

### Diagramas de secuencia

Los diagramas de secuencia son usados para modelar los aspectos dinámicos del sistema. Ilustran las interacciones entre los distintos objetos participantes y los mensajes que intercambian. Estos diagramas se forman colocando los objetos y los actores que interactúan en el eje horizontal, y en el eje vertical se colocan los mensajes intercambiados en orden temporal, desde arriba hacia

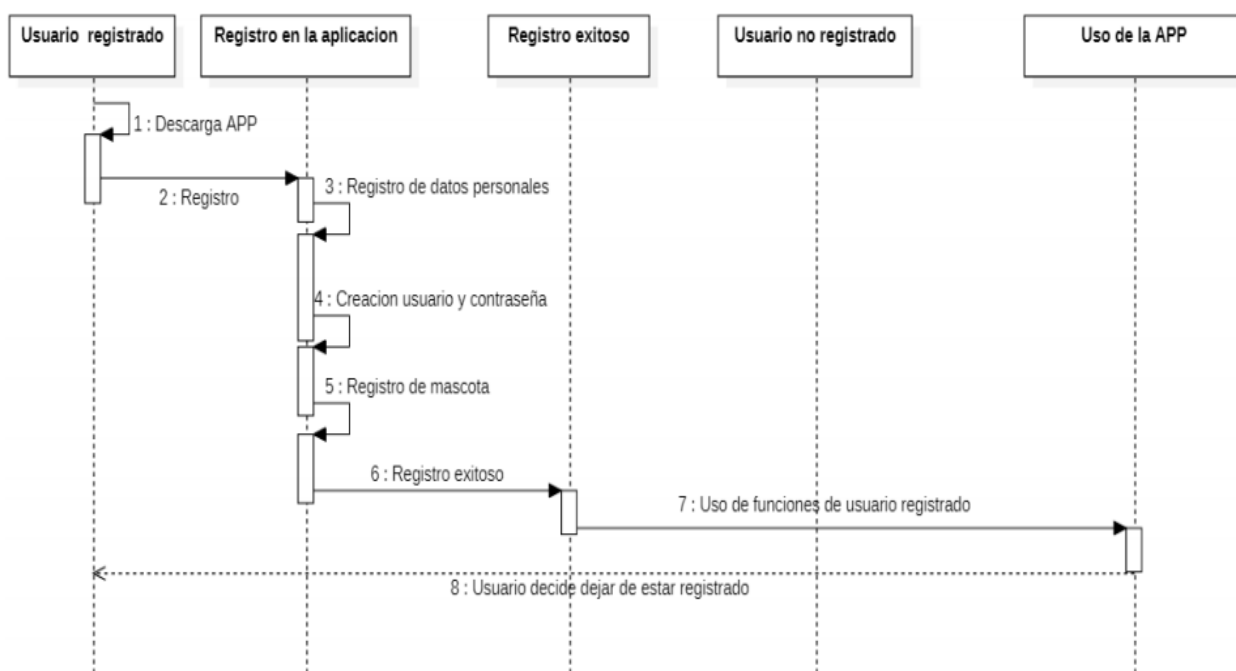
abajo. El objeto que inicia la interacción suele ubicarse a la izquierda y los objetos subordinados a la derecha. A continuación, se explicarán cada uno de los diagramas de secuencia definidos:

## 1. Registro:

- Usuario registrado:

*Imagen 29*

*Diagrama de secuencia registro – usuario registrado*



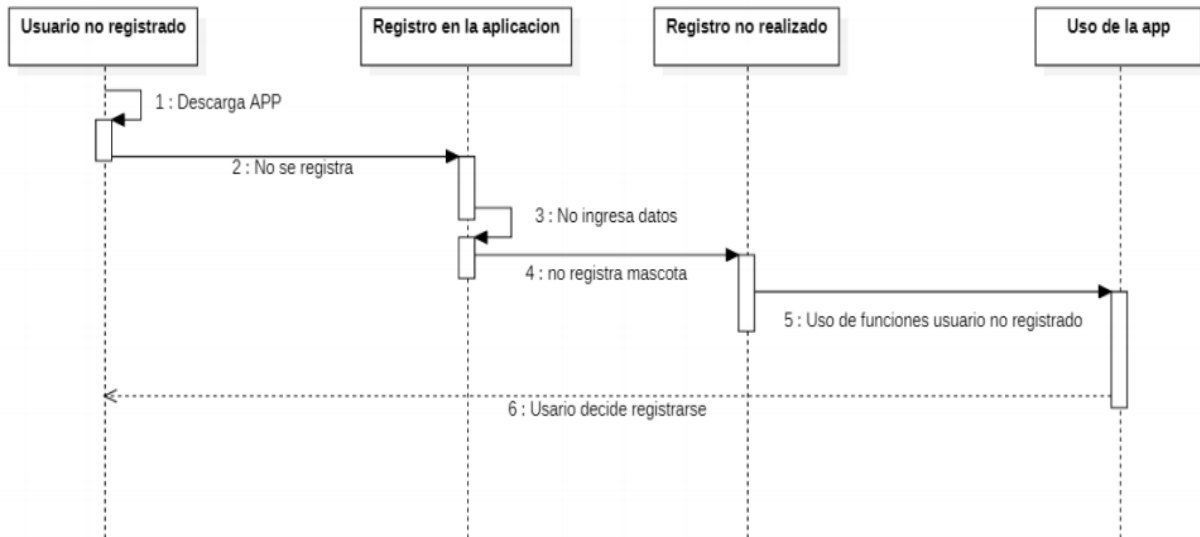
*Nota.* Imagen que ilustra el diagrama de secuencia del usuario registrado. Creación propia, 2021



- Usuario no registrado:

*Imagen 30*

*Diagrama de secuencia registro – usuario no registrado*



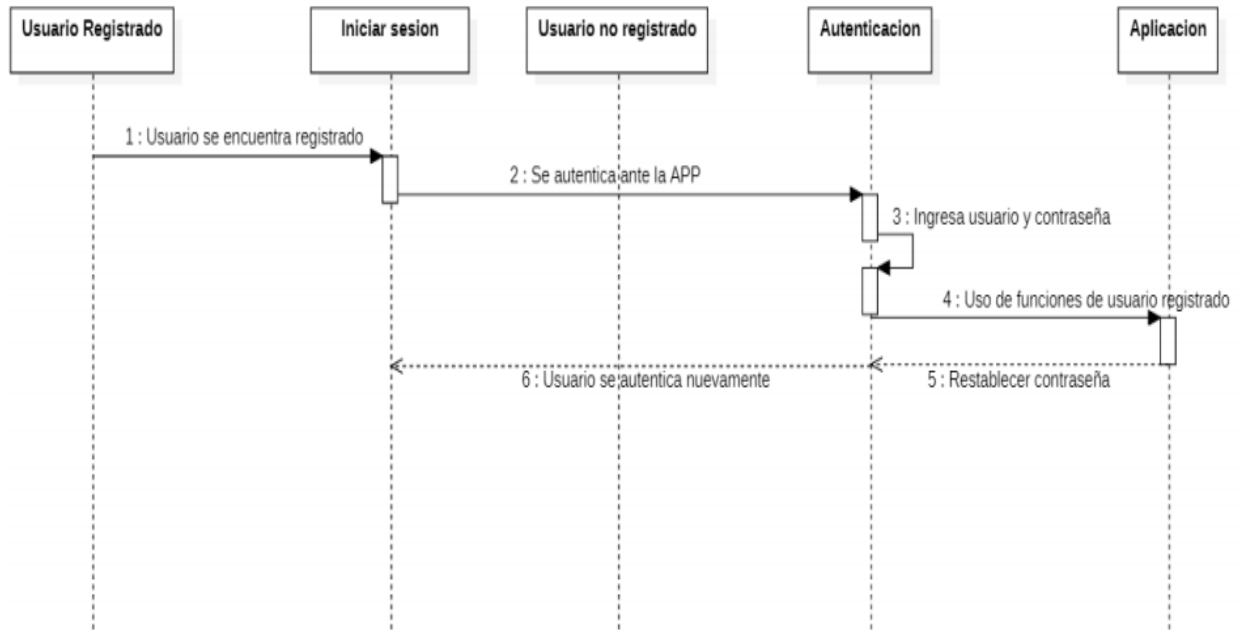
*Nota.* Imagen que ilustra el diagrama de secuencia del usuario no registrado. Creación propia, 2021

## 2. Autenticación:

- Usuario registrado:

*Imagen 31*

*Diagrama de secuencia autenticación – usuario registrado*



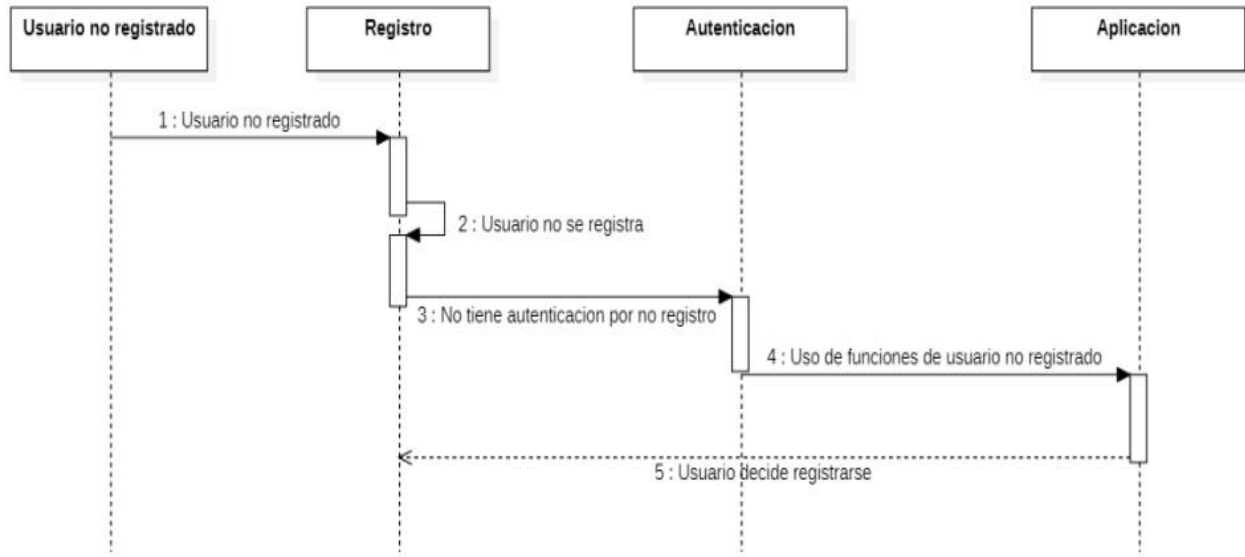
*Nota.* Imagen que ilustra el diagrama de secuencia de autenticación del usuario registrado.

Creación propia, 2021

- Usuario no registrado:

*Imagen 32*

*Diagrama de secuencia autenticación – usuario registrado*



*Nota.* Imagen que ilustra el diagrama de secuencia de autenticación del usuario no registrado.

Creación propia, 2021

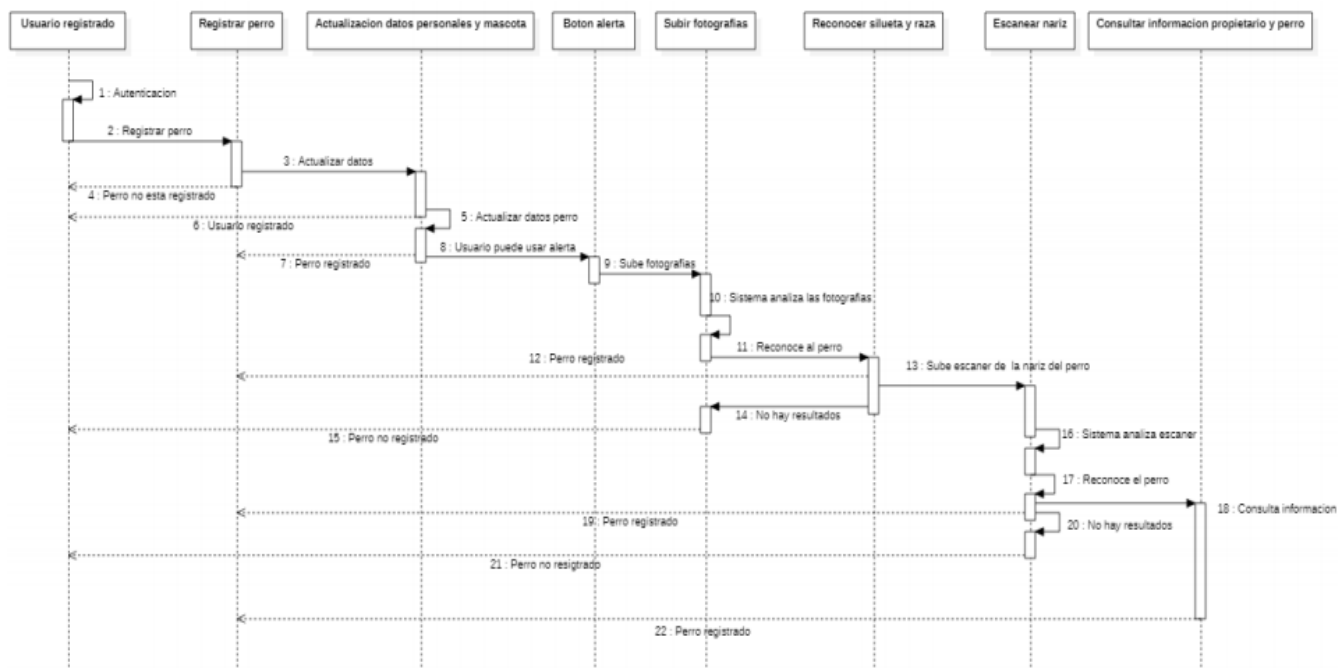
*Ilustración 1. Diagrama de secuencia autenticación – usuario registrado*

**3. Aplicación DiNo (APP):**

- Usuario registrado:

### Imagen 33

#### Diagrama de secuencia aplicación – usuario registrado



*Nota.* Imagen que ilustra el diagrama de secuencia de aplicación del usuario registrado. Creación propia, 2021

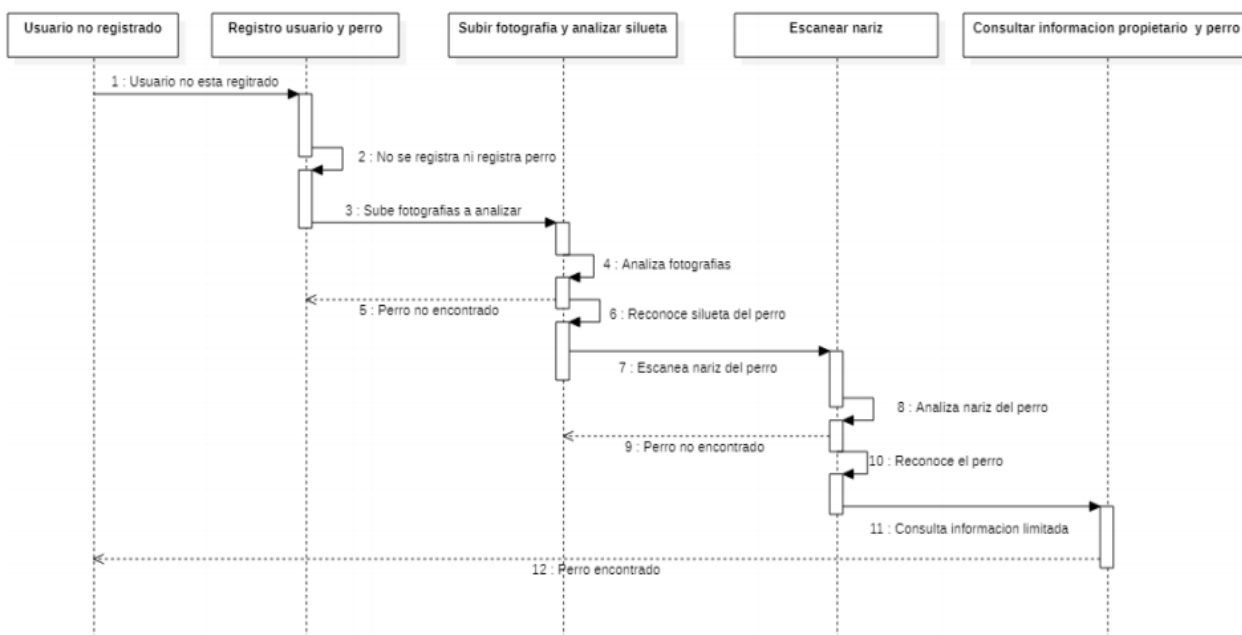
### Ilustración 2.

#### Diagrama de secuencia aplicación – usuario registrado

- Usuario no registrado:

### Imagen 34

#### Diagrama de secuencia aplicación – usuario no registrado



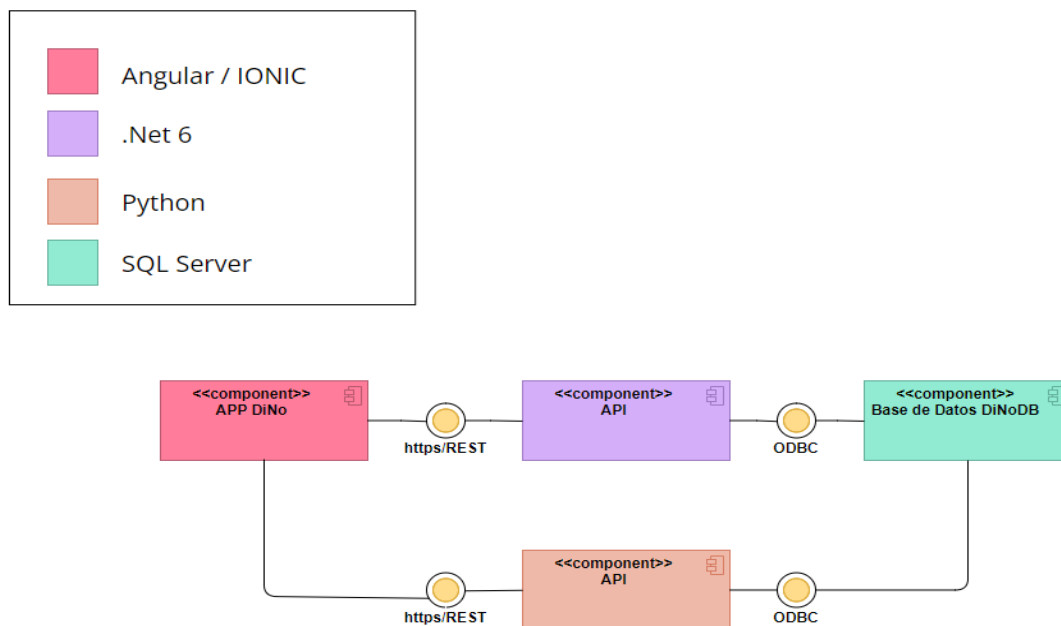
*Nota.* Imagen que ilustra el diagrama de secuencia de aplicación del usuario no registrado.

Creación propia, 2021

## Diagrama componentes

### Imagen 35

#### Diagrama de componentes

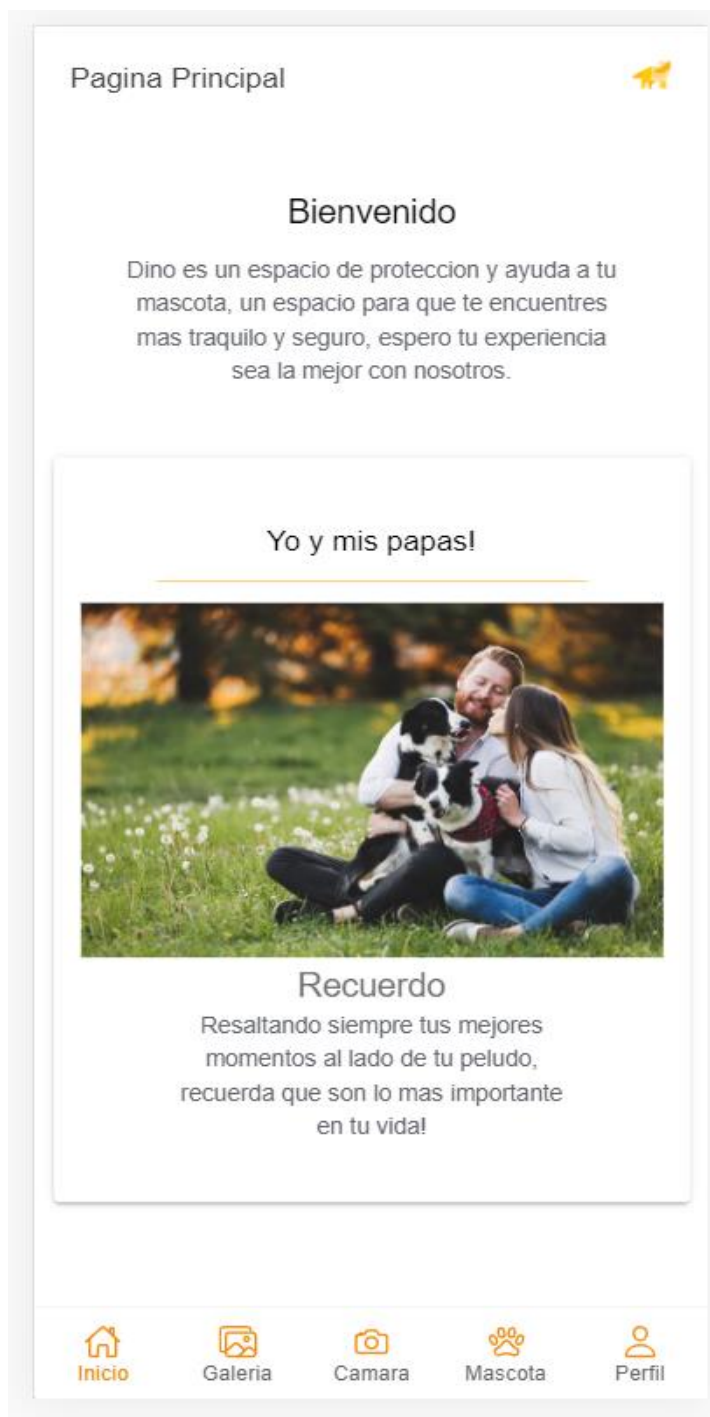


*Nota.* La imagen ilustra los componentes que componen el software. Creación propia, 2022

## Interfaz de usuario

La interfaz de usuario se estructuró con el objetivo de que la experiencia de usuario sea sencilla y eficiente.

A continuación, se presenta la pantalla principal de la aplicación,

**Imagen 36****Pantalla principal APP**

*Nota.* La imagen ilustra la pantalla principal de la APP. Creación propia, 2022

Esta se compone de 5 botones y cada uno de estos tiene la siguiente función:

- Inicio: Pantalla principal, se compone de mensajes de bienvenida y una imagen que el usuario coloca como portada.
- Galería: Pantalla donde el usuario puede seleccionar una foto de la galería.
- Cámara: Pantalla donde la aplicación abre la cámara del smartphone y puede capturar una foto.
- Mascota: Pantalla donde se encuentra la información de las mascotas del usuario.
- Perfil: Pantalla donde se encuentra la información del usuario.

En el proceso de desarrollo se ilustrará cada una de las pantallas que componen cada una de las interfaces del usuario y sus funcionalidades.

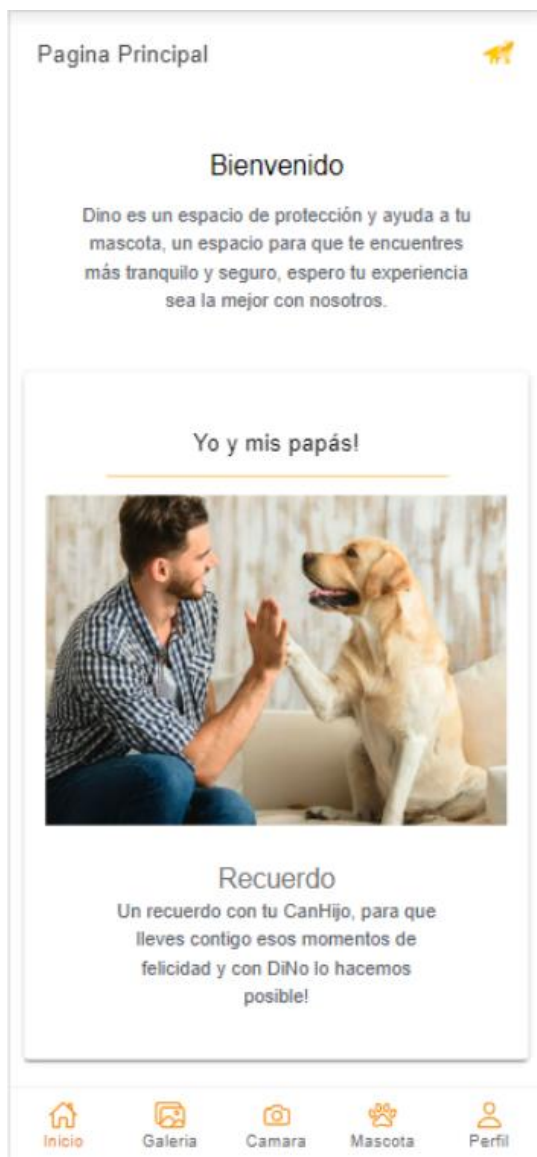
## **9 Desarrollo del software**

A continuación, se describirá el desarrollo del software dividido en sus entregables por cada uno de los sprint definidos.

### **Sprint 1:**

#### **9.1.1 Frontend: Pantalla (interfaz) principal, registro e inicio de sesión.**




**Imagen 37****Pantalla principal APP**

*Nota.* La imagen ilustra la pantalla principal de la APP. Creación propia, 2022


En la pantalla principal de la aplicación el usuario va a encontrar un mensaje de bienvenida, y puede agregar una foto de portada la cual se visualizará al estar en esta pantalla. En la parte inferior dispone de los botones para moverse entre los diferentes menús.


**Imagen 38****Pantalla registro APP**


← Registrarse 

### Registrar cuenta

Complete todos los datos y  
presione continuar

Email 

Contraseña 

Confirmar contraseña 

**CONTINUAR**

*Nota.* La imagen ilustra la pantalla de registro de la APP. Creación propia, 2022

Para registrarse el usuario debe proporcionar su email y una contraseña, seguido a esto debe completar su perfil llenando los siguientes campos.

**Imagen 39**

*Pantalla para completar el perfil posterior al registro*

← Completar Perfil

Complete su perfil

Complete toda la información para continuar

Nombres

Apellidos

Cédula

Edad

Género

Teléfono

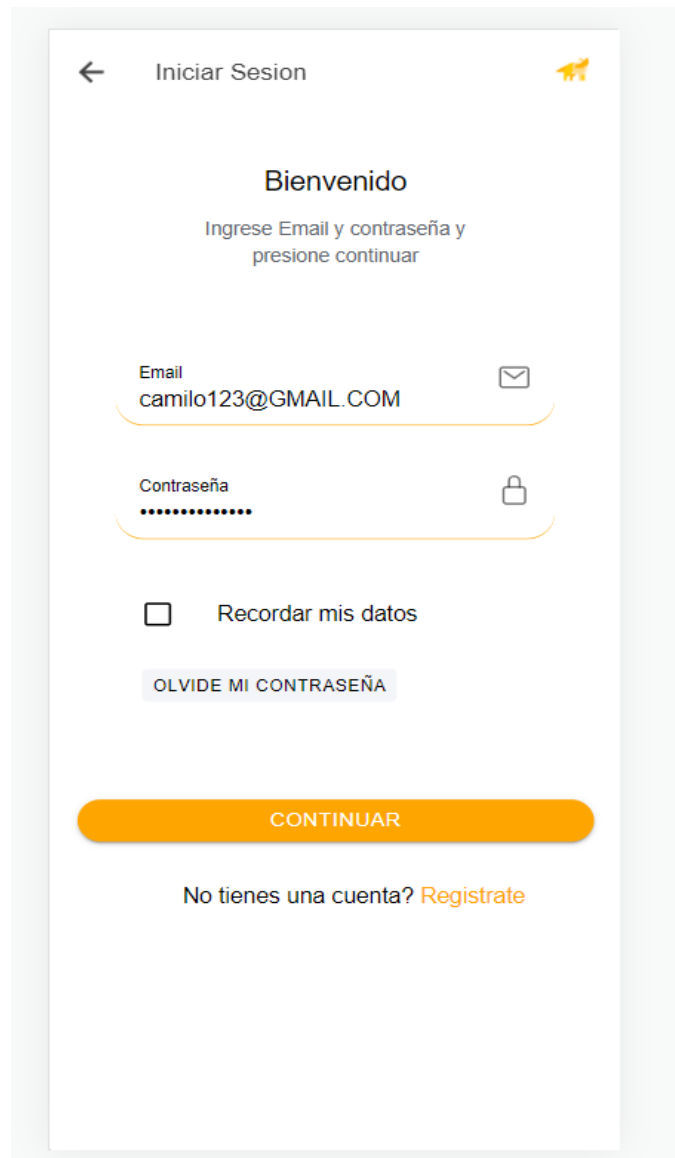
Ciudad

*Nota.* La imagen ilustra la pantalla para completar el perfil. Creación propia, 2022

Para el inicio de sesión el usuario debe proporcionar el email y contraseña registrado

### **Imagen 40**

#### *Pantalla de inicio de sesión*



The image shows a mobile application login screen. At the top, there is a back arrow on the left and the text 'Iniciar Sesión' in the center, with a small orange fox icon on the right. Below this, the word 'Bienvenido' is centered, followed by the instruction 'Ingrese Email y contraseña y presione continuar'. There are two input fields: 'Email' with the value 'camilo123@GMAIL.COM' and an envelope icon, and 'Contraseña' with a lock icon and a masked password. Below the password field is a checkbox labeled 'Recordar mis datos'. A button labeled 'OLVIDE MI CONTRASEÑA' is positioned below the checkbox. A large orange button labeled 'CONTINUAR' is at the bottom. At the very bottom, there is a link: 'No tienes una cuenta? [Regístrate](#)'.

*Nota.* La imagen ilustra la pantalla de inicio de sesión. Creación propia, 2022

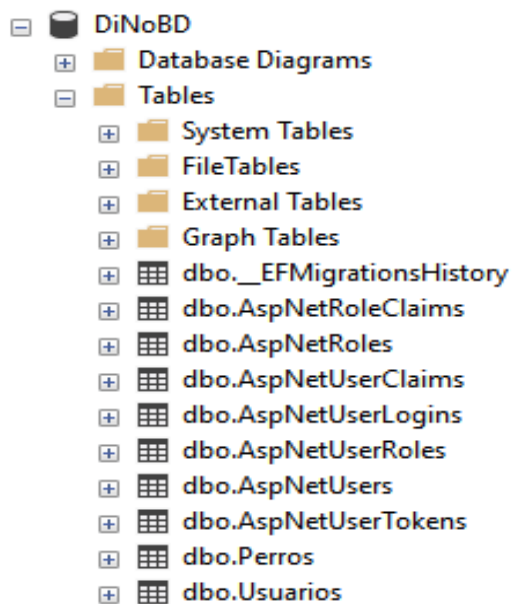
### 9.1.2 Backend: Base de datos y API .NET

Como se mencionó anteriormente, el motor de base de datos definido para el proyecto fue SQL Server, allí se diseñó e implementó todo el modelo relacional para satisfacer los requerimientos de la aplicación.

A continuación, se describirán cada una de las tablas creadas en la base de datos y su función:

#### *Imagen 41*

#### *Composición tablas de base de datos APP*



*Nota.* La imagen ilustra la composición de las tablas en el motor de base de datos SQL. Creación propia, 2022

**Tablas AspNet:** Estas tablas se crean implícitamente al trabajar con entity framework desde .NET, estas se usan para el registro y autenticación de usuario en la aplicación.

**Tabla Usuarios:** Esta tabla contiene la información de los usuarios que se registraron en la aplicación (nombres, apellidos, número de cédula, edad, genero, número de celular, ciudad, dirección de residencia, foto perfil, foto portada).

**Tabla Perros:** Esta table contiene toda la información de los perros registrados en la aplicación (nombre, edad, foto, raza, medicación).

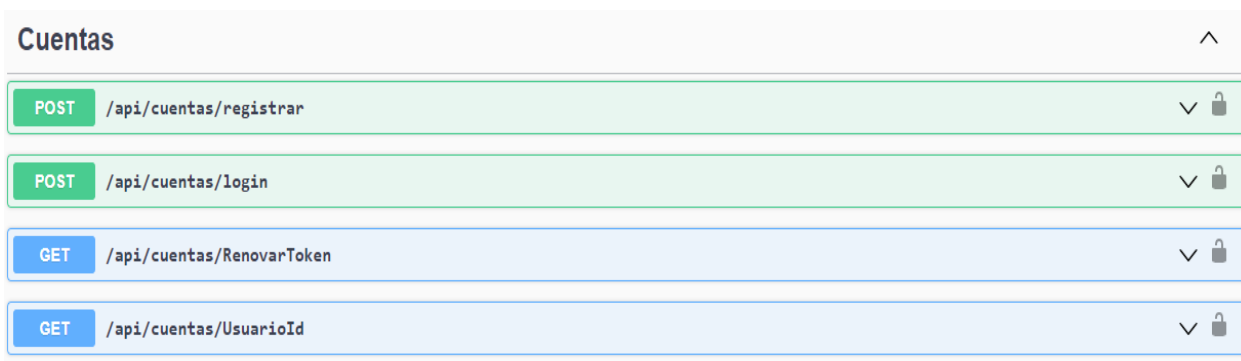
Posteriormente, se desarrollo la primera API que compone la aplicación, para esto, se utilizó .Net 6 con entity framework, para creación de entidades y tablas directamente en la base de datos. En esta API se encuentran expuestos todos los recursos para el manejo del usuario de la aplicación. Es importante resaltar que el manejo con la API se basa en operaciones CRUD. En la siguiente imagen ([Imagen41](#)) se encuentra la interfaz que se dispone para su visualización y manejo de los recursos expuestos.

**Autenticación:** Se desarrollo el controlador de **Cuentas**, este esta encargado de todo lo que se refiere a la autenticación y seguridad dentro de la aplicación.



## Imagen 42

Controlador cuentas, API .NET, swagger



The image shows a Swagger UI interface for an API. The title is 'Cuentas'. There are four endpoints listed:

Method	Endpoint	Visibility	Security
POST	/api/cuentas/registrar	Visible	Protected
POST	/api/cuentas/login	Visible	Protected
GET	/api/cuentas/RenovarToken	Visible	Protected
GET	/api/cuentas/UsuarioId	Visible	Protected

*Nota.* La imagen ilustra el controlador de cuentas. Creación propia, 2022

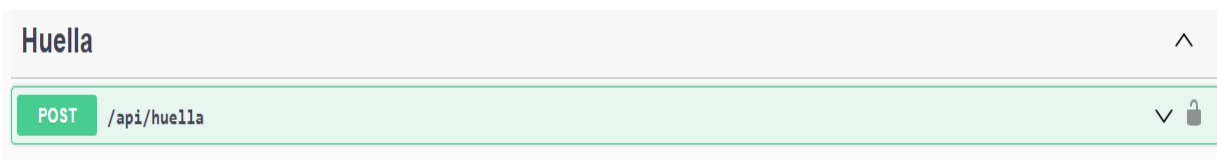
El controlador de cuentas de la Imagen 42 permite:

- **/api/cuentas/registrar:** Metodo POST para registrar un usuario.
- **/api/cuentas/login:** Metodo POST para autenticarse en la aplicación (inicio de session).
- **/api/cuentas/RenovarToken:** Metodo GET para renovar el token, el metodo de autenticación y autorización usado en la API es basado en el estandar JWT (JSON WEB TOKEN) cada ves que el usuario inicio session se le asigna un token con una vigencia de 30 minutos, si el usuario permanece en la aplicación mas de 30 minutos automaticamente este token se renueva para no afectar la experiencia de usuario.
- **/api/cuentas/UsuarioId:** Metodo GET que devuelve como respuesta el ID del usuario, necesario para operaciones específicas.

**Huella:** Se desarrolló el controlador de **Huella**, este se encarga de recibir la imagen del perro y procesarla en el algoritmo de IA para identificar al perro.

### Imagen 43

Controlador huella, API .Net, swagger



Nota. La imagen ilustra el controlador de huella. Creación propia, 2022

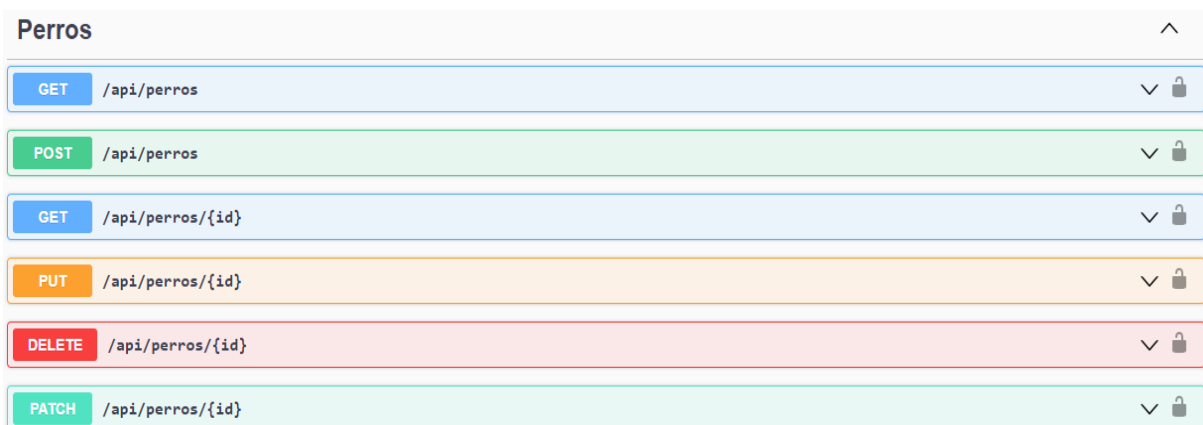
El controlador de huella permite:

- **/api/huella:** Metodo POST donde recibe la imagen en BASE64, la procesa en el algoritmo de IA y devuelve un objeto tipo JSON con la información del perro.

**Perros:** Se desarrollo el controlador de **Perros** y se encarga de todas las operaciones con la entidad perro.

### Imagen 44

Controlador perros, API .NET, swagger



Nota. La imagen ilustra el controlador de perros. Creación propia, 2022

Este controlador de perros permite:

- **/api/perros:** Metodo GET para consultar todos los perros registrados.
- **/api/perros:** Metodo POST para crear un perro.
- **/api/perros/{id}:** Metodo GET para consultar un perro por su ID.
- **/api/perros/{id}:** Metodo PUT para actualizar un perro por su ID.
- **/api/perros/{id}:** Metodo DELETE para eliminar un perro por su ID.
- **/api/perros/{id}:** Metodo PATCH para actualizar solo un campo del perro por su ID.

Por ejemplo la raza, el nombre, la medicación, etc.

**Usuarios:** Se desarrollo el controlador **Usuarios**, este se encarga de todas las operaciones con la entidad usuario.

### *Imagen 45*

*Controlador de usuarios, API .NET, swagger*

Usuarios	
GET	/api/usuarios
POST	/api/usuarios
GET	/api/usuarios/{id}
PUT	/api/usuarios/{id}
DELETE	/api/usuarios/{id}
PATCH	/api/usuarios/{id}

*Nota.* La imagen ilustra el controlador de usuarios. Creación propia 2022

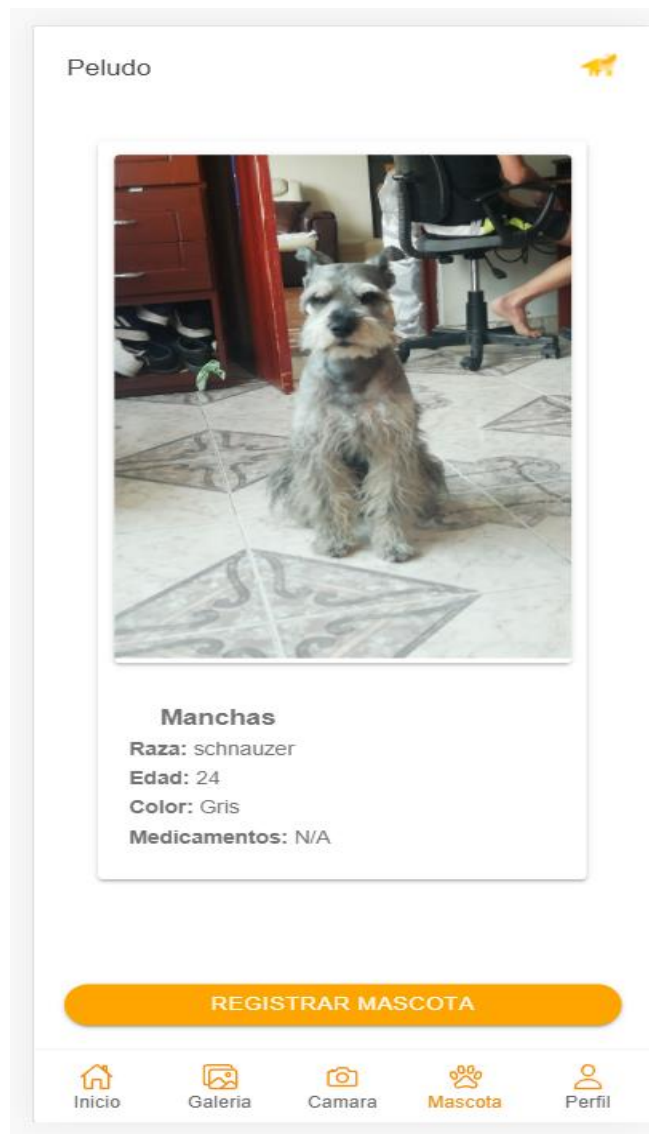
Este controlador de usuarios permite:

- **/api/usuarios:** Metodo GET para consultar todos los usuarios registrados.
- **/api/usuarios:** Metodo POST para crear un usuario.
- **/api/usuarios /{id}:** Metodo GET para consultar un usuario por su ID.
- **/api/usuarios /{id}:** Metodo PUT para actualizar un usuario por su ID.
- **/api/usuarios /{id}:** Metodo DELETE para eliminar un usuario por su ID.
- **/api/usuarios /{id}:** Metodo PATCH para actualizar solo un campo del usuario por su ID. Por ejemplo el nombre, la edad, la dirección de residencia, etc.

## **Sprint 2:**

### **9.1.3 Frontend: Pantalla (interfaz) mascotas.**

En esta pantalla se encuentra toda la información de las mascotas que el usuario registro.

**Imagen 46***Pantalla de mascotas*

*Nota.* La imagen ilustra la pantalla de mascotas de la APP. Creación propia, 2022

**9.1.4 Backend: API Python.**

Posteriormente, se desarrolló la segunda API que compone la aplicación, para hacerla se utilizó el lenguaje python con el framework de Flask. En esta API se encuentra expuesto el recurso para

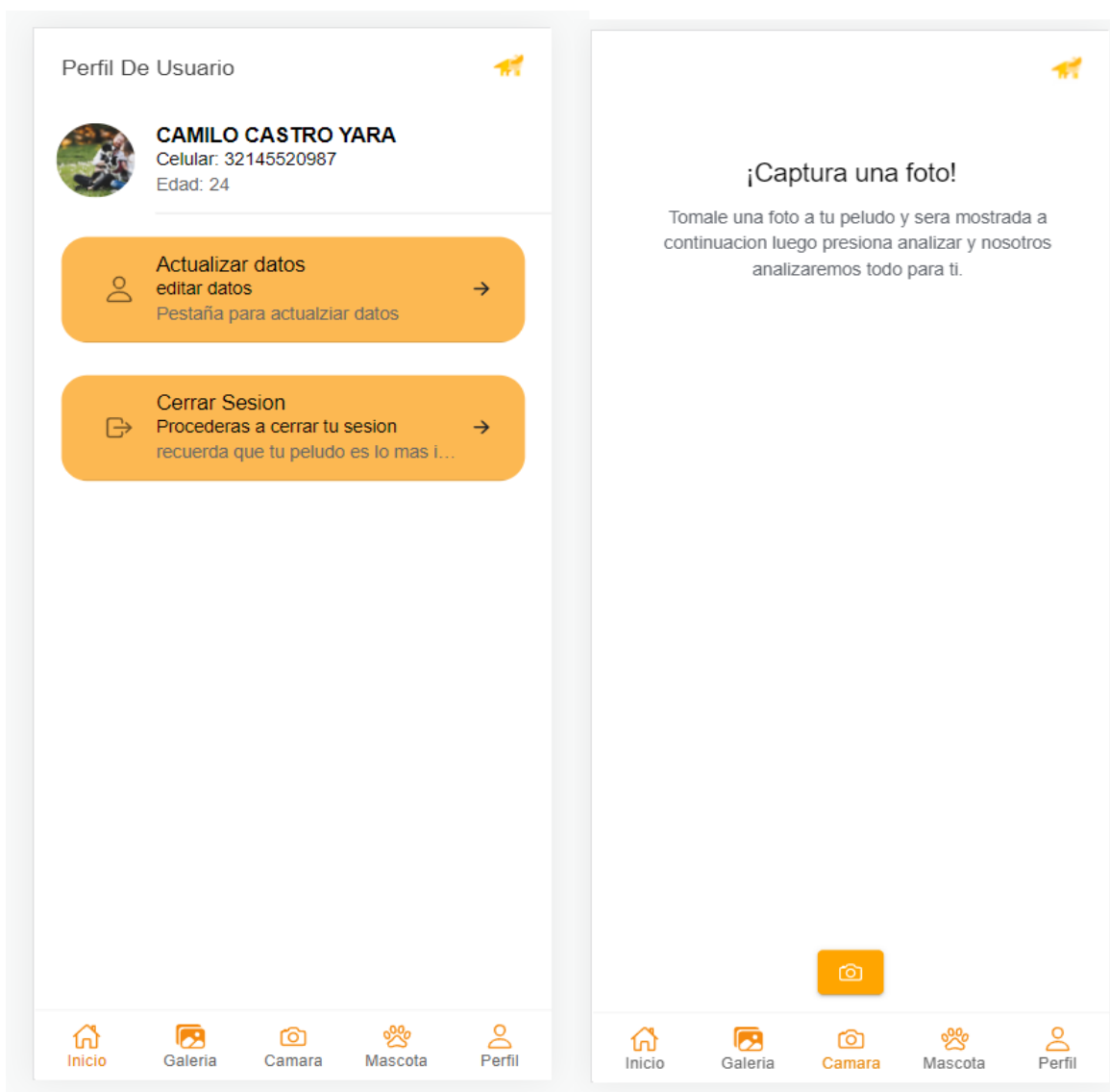
el reconocimiento de la raza del perro. Es importante resaltar que el manejo con la API se basa en operaciones CRUD, en este caso, solo tiene un endpoint de tipo POST.

### Sprint 3

#### 9.1.5 Frontend: Pantalla (interfaz) perfil. (3) y Pantalla (interfaz) cámara

##### Imagen47

*Pantalla Interfaz perfil e interfaz cámara*



*Nota.* La imagen ilustra la pantalla de perfil e interfaz cámara de la APP. Creación propia, 2022

## 10 Pruebas del proyecto

Para probar el funcionamiento del software se realizó el siguiente set de pruebas.

### Set de pruebas

Este set fue construido de forma tal que se evaluarán todos los componentes de la aplicación, experiencia, interacción y dinamicidad del usuario.

### *Imagen 48*

#### *Set de pruebas APP DiNo*

No.	PUNTOS DE VERIFICACION	PASA LA PRUEBA (Seleccione con una X)		OBSERVACIONES
		SI	NO	
1	Registro de usuario	X		Se usaran 3 usuarios de prueba
2	Autenticación usuario registrado	X		se usaran los 3 usuarios de prueba creados
3	Registro de perro	X		Se registrara minimo un perro por cada usuario de prueba creado
4	Reconocimiento huella digital perro	X		Se analizara la huella de 3 perros registrados
5	Reconocimiento silueta perro	x		Se analizara la silueta de 5 perros
6	Actualización datos usuario	X		Se actualizaran los datos de 1 usuario
7	Actualización datos perro	X		Se actualizaran los datos de 1 perro
<b>OBSERVACIONES COMPLEMENTARIAS</b>				
<b>CONCLUSIONES</b>				
<i>Pruebas satisfactorias, se aprueba para paso a producción. Se debe ajustar el modelo de reconocimiento de la huella digital.</i>				

*Nota.* La imagen ilustra todo el set de pruebas realizado. Creación propia, 2022

## Evidencias set de pruebas

A continuación, se va a ilustrar cada uno de los puntos de verificación probados en la aplicación en ambiente de pruebas.

### 10.1.1 Punto de verificación 1 – registro usuario

En este punto se verificó que permitiera diligenciar los formularios, validar campos vacíos, estructura de los campos, generara errores si el usuario digitaba mal el correo o la contraseña mostrando un mensaje de error, interacción entre la pantalla de registro y la de completar perfil. Posteriormente al completar el registro se verificó que toda la información del usuario quedara registrada en la base de datos.

#### ***Tabla 18***

##### *Registro usuario 1*

Pantalla registro	Pantalla completar perfil	Pantalla completar perfil
-------------------	---------------------------	---------------------------



The image displays three sequential mobile app screens for user registration and profile completion. The first screen, titled 'Registrarse', prompts the user to 'Registrar cuenta' and provides input fields for 'Email' (juan.castro@gmail.com), 'Contraseña', and 'Confirmar contraseña', with a 'CONTINUAR' button. The second screen, 'Completar Perfil', asks to 'Complete su perfil' and includes fields for 'Nombres' (Juan Camilo), 'Apellidos' (Castro Yara), 'Cedula' (12345678921), 'Edad' (28), 'Genero' (Masculino), 'Telefono' (3133992800), and 'Ciudad' (Bogota D.C). The third screen, also 'Completar Perfil', shows fields for 'Cedula' (12345678921), 'Edad' (28), 'Genero' (Masculino), 'Telefono' (3133992800), 'Ciudad' (Bogota D.C), and 'Direccion' (calle 59 # 60 -98 norte), along with a 'SELECCIONA TU FOTOGRAFIA' button and a 'CONTINUAR' button. A disclaimer at the bottom of the third screen states 'Al continuar aceptara terminos y condiciones'.

**Tabla 19****Registro usuario 2**

Pantalla registro

Pantalla completar perfil

Pantalla completar perfil

The image displays three sequential screenshots of a mobile application's registration and profile completion process. Each screen features a back arrow on the top left and a title at the top center.

- Screen 1: Registrarse**
  - Title: Registrar cuenta
  - Instruction: Complete todos los datos y presione continuar
  - Fields: Email (diana.cortes@gmail.com), Contraseña (masked), Confirmar contraseña (masked)
  - Button: CONTINUAR
- Screen 2: Completar Perfil**
  - Title: Complete su perfil
  - Instruction: Complete toda la informacion para continuar
  - Fields: Nombres (Diana), Apellidos (Cortes), Cedula (798456312), Edad (25), Genero (Femenino), Telefono (2165498), Ciudad (Bogota D.C)
  - Button: SELECCIONA TU FOTOGRAFIA
- Screen 3: Completar Perfil**
  - Title: Completar Perfil
  - Fields: Cedula (798456312), Edad (25), Genero (Femenino), Telefono (2165498), Ciudad (Bogota D.C), Direccion (Avenida 1 de mayo # 89 -45)
  - Button: CONTINUAR
  - Text: Al continuar aceptara terminos y condiciones

**Tabla 20***Registro usuario 3*

Pantalla registro

Pantalla completar perfil

Pantalla completar perfil

The image displays three sequential screens from a mobile application:

- Registrarse (Register):** The screen is titled "Registrar cuenta" (Create account). It prompts the user to "Complete todos los datos y presione continuar" (Complete all data and press continue). It features three input fields: "Email" (edisonflorezz89@gmail.com), "Contraseña" (password), and "Confirmar contraseña" (confirm password). A large orange "CONTINUAR" button is at the bottom.
- Completar Perfil (Complete Profile):** The screen is titled "Complete su perfil" (Complete your profile). It prompts the user to "Complete toda la información para continuar" (Complete all information to continue). It lists personal details: "Nombres" (Edison), "Apellidos" (Flores), "Cedula" (9632588741), "Edad" (58), "Genero" (Masculino), "Telefono" (23165498), and "Ciudad" (Cali). Each field has a corresponding icon (person, ID card, hourglass, gender, phone, location pin).
- Completar Perfil (Complete Profile):** This screen continues the profile completion. It shows "Cedula" (9632588741), "Edad" (58), "Genero" (Masculino), "Telefono" (23165498), and "Ciudad" (Cali). It also includes a "Direccion" field with the value "Calle 23 # 45 - 15". At the bottom, there is an orange button labeled "SELECCIONA TU FOTOGRAFIA" (Select your photo) with a camera icon, and another orange "CONTINUAR" button. A small note at the bottom states: "Al continuar aceptara terminos y condiciones" (By continuing, you accept terms and conditions).

### 10.1.2 Punto de verificación 2 – autenticación usuarios registrados

En este punto se verifico que la autenticación de los usuarios previamente registrados fuera exitosa. Si el usuario no esta registrado o digita mal su email o contraseña, la aplicación generara un mensaje de error indicándole que alguno de los campos es incorrecto.

Tabla 21

## Autenticación usuario 1


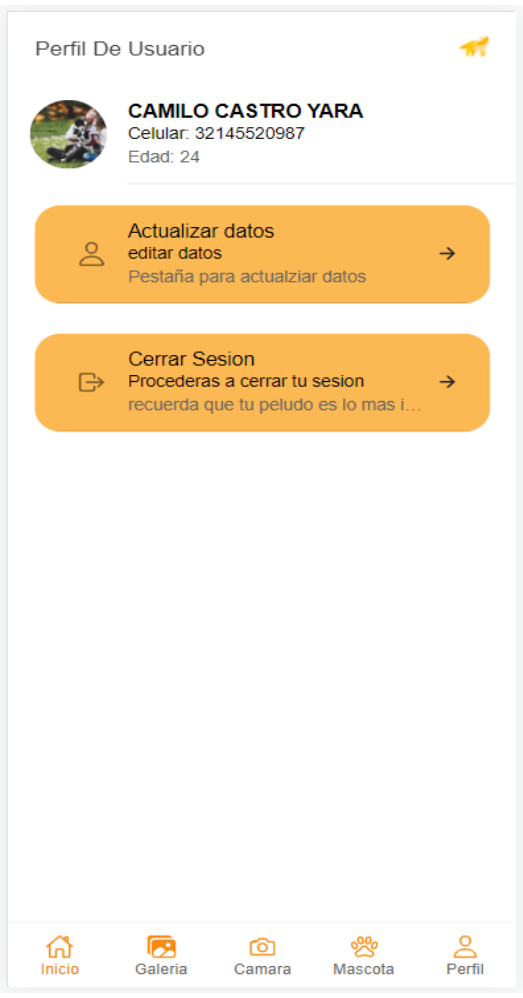
Pantalla inicio de sesión	Pantalla perfil
	

Tabla 22

## Autenticación usuario 2

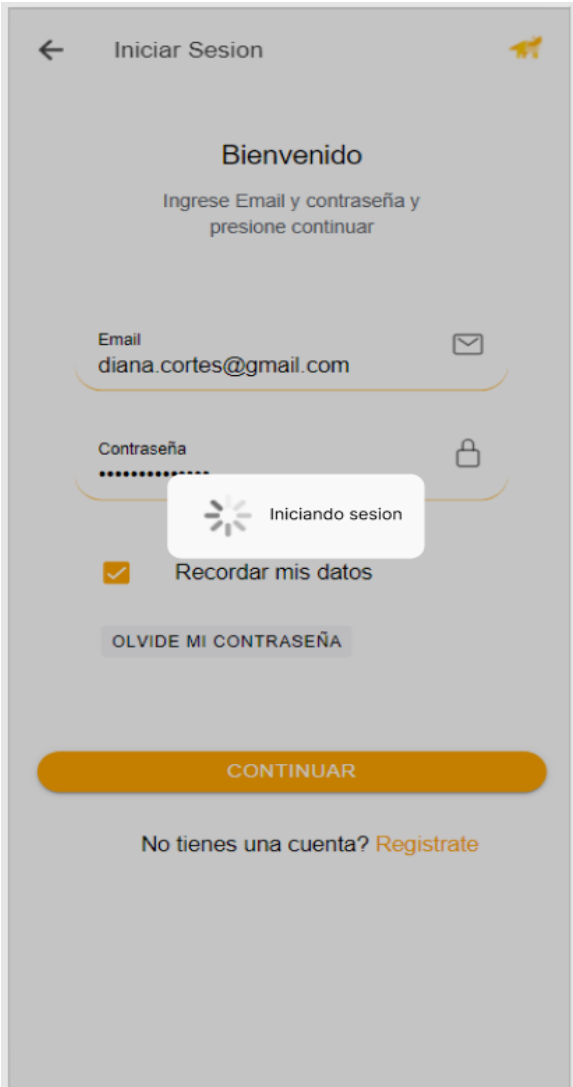
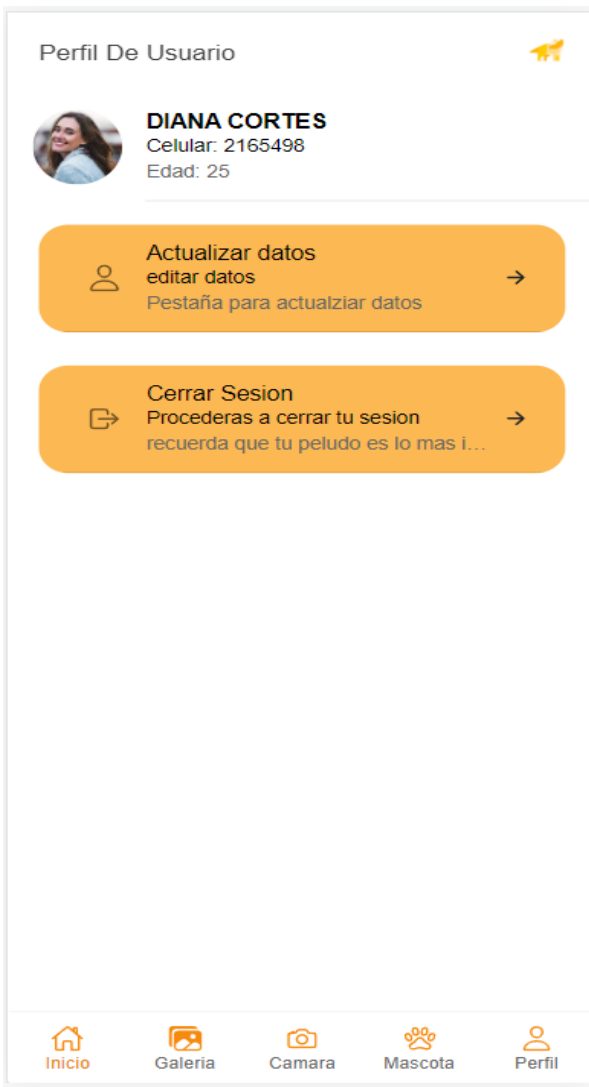

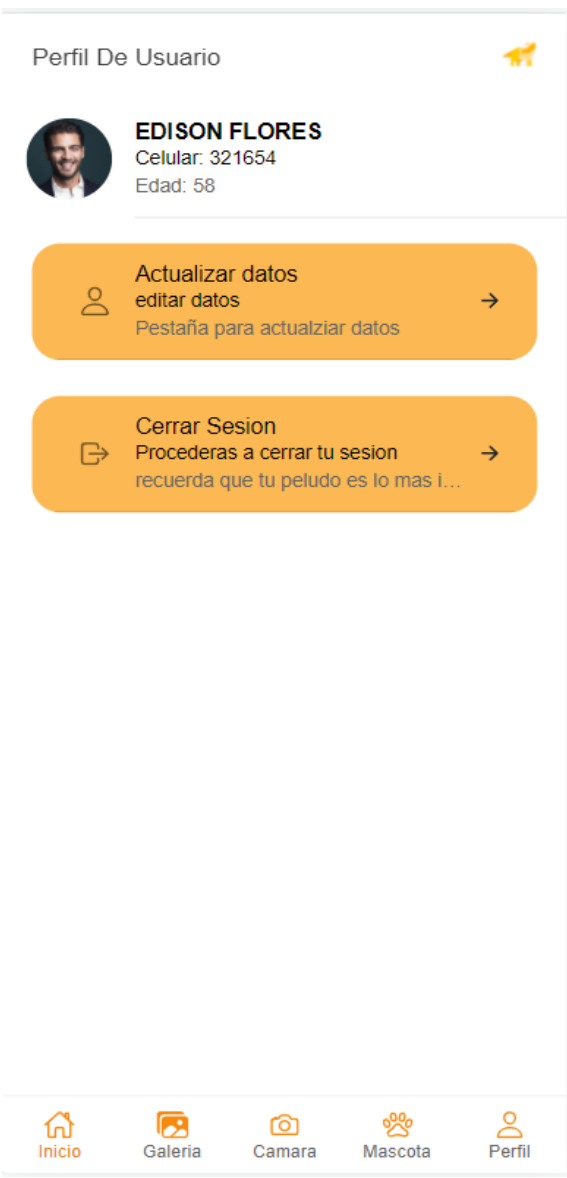
Pantalla inicio de sesión	Pantalla perfil
 <p>The screenshot shows the login screen with the title "Iniciar Sesión". It features a "Bienvenido" message and instructions to enter email and password. The email field contains "diana.cortes@gmail.com" and the password field is masked with dots. A "Recordar mis datos" checkbox is checked. A loading spinner with the text "Iniciando sesion" is overlaid on the password field. Below the fields is a "CONTINUAR" button and a link to "Registrate".</p>	 <p>The screenshot shows the user profile screen with the title "Perfil De Usuario". It displays the user's name "DIANA CORTES", phone number "Celular: 2165498", and age "Edad: 25". There are two main action buttons: "Actualizar datos" (with subtext "editar datos" and "Pestaña para actualziar datos") and "Cerrar Sesion" (with subtext "Procederas a cerrar tu sesion" and "recuerda que tu peludo es lo mas i..."). A bottom navigation bar includes icons for Inicio, Galeria, Camara, Mascota, and Perfil.</p>

Tabla 23

## Autenticación usuario 3

Pantalla inicio de sesión	Pantalla perfil
 <p>Iniciar Sesión</p> <p>Bienvenido</p> <p>Ingrese Email y contraseña y presione continuar</p> <p>Email edisonflorezz89@gmail.com</p> <p>Contraseña</p> <p>Iniciando sesion</p> <p><input checked="" type="checkbox"/> Recordar mis datos</p> <p>OLVIDE MI CONTRASEÑA</p> <p>CONTINUAR</p> <p>No tienes una cuenta? <a href="#">Regístrate</a></p>	 <p>Perfil De Usuario</p> <p>EDISON FLORES</p> <p>Celular: 321654</p> <p>Edad: 58</p> <p>Actualizar datos editar datos</p> <p>Cerrar Sesion Procederas a cerrar tu sesion</p> <p>Inicio   Galeria   Camara   Mascota   Perfil</p>

### 10.1.3 Punto de verificación 3 – registro perro

En este punto se verificó que cada uno de los usuarios lograra registrar uno o varios perros.

**Tabla 24**

*Registro perro 1, usuario 1*


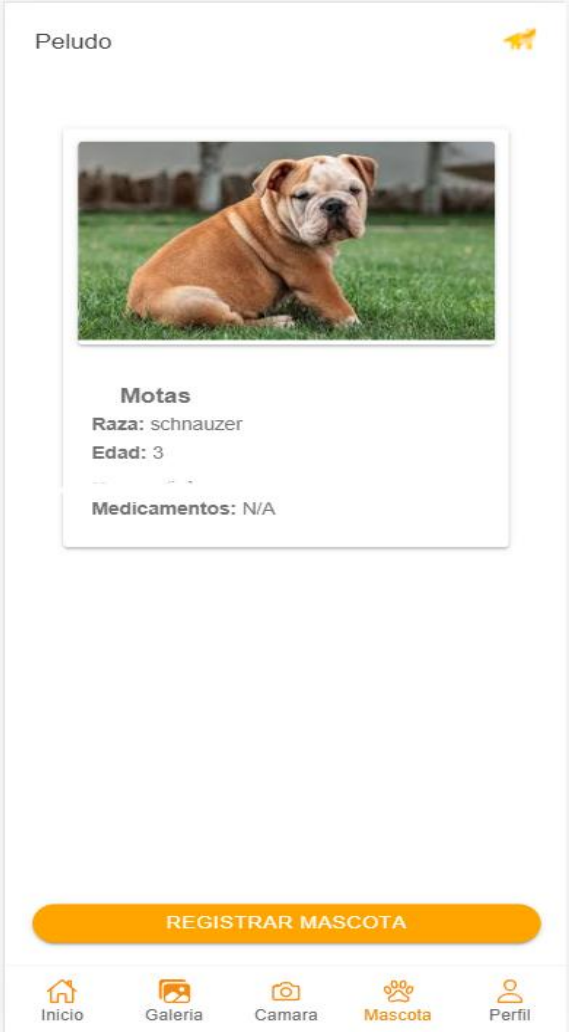
Pantalla registro perro	Pantalla mascota
 <p>Registro De Peludo</p> <p>Perfil del peludo Completa toda la informacion para continuar</p> <p>Nombre Motas</p> <p>Edad 3</p> <p>Medicacion N/A</p> <p>SELECCIONA LA FOTO DE TU PELUDO</p> <p>CONTINUAR</p> <p>Al continuar aceptara terminos y condiciones</p>	 <p>Peludo</p> <p>Motas</p> <p>Raza: schnauzer Edad: 3 Medicamentos: N/A</p> <p>REGISTRAR MASCOTA</p> <p>Inicio Galeria Camara Mascota Perfil</p>

Tabla 25


Registro perro 2, usuario 1


Pantalla registro perro


Pantalla mascota


← Registro De Peludo

**Perfil del peludo**  
Completa toda la información para continuar

Nombre  
Zeus 


Edad  
20 


Medicacion  
N/A 

SELECCIONA LA FOTO DE TU PELUDO 


CONTINUAR

Al continuar aceptara terminos y condiciones

Peludo 



**Zeus**  
Raza: schnauzer  
Edad: 20  
Medicamentos: N/A



REGISTRAR MASCOTA






Inicio  Galeria  Camara  Mascota  Perfil 



Tabla 26

Registro perro 1, usuario 2


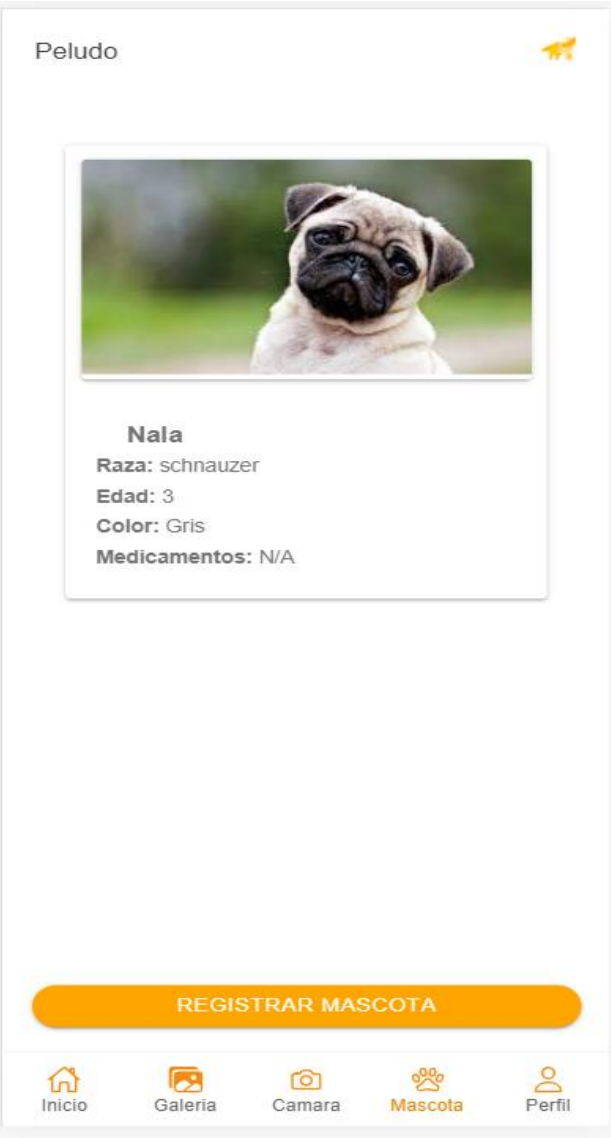
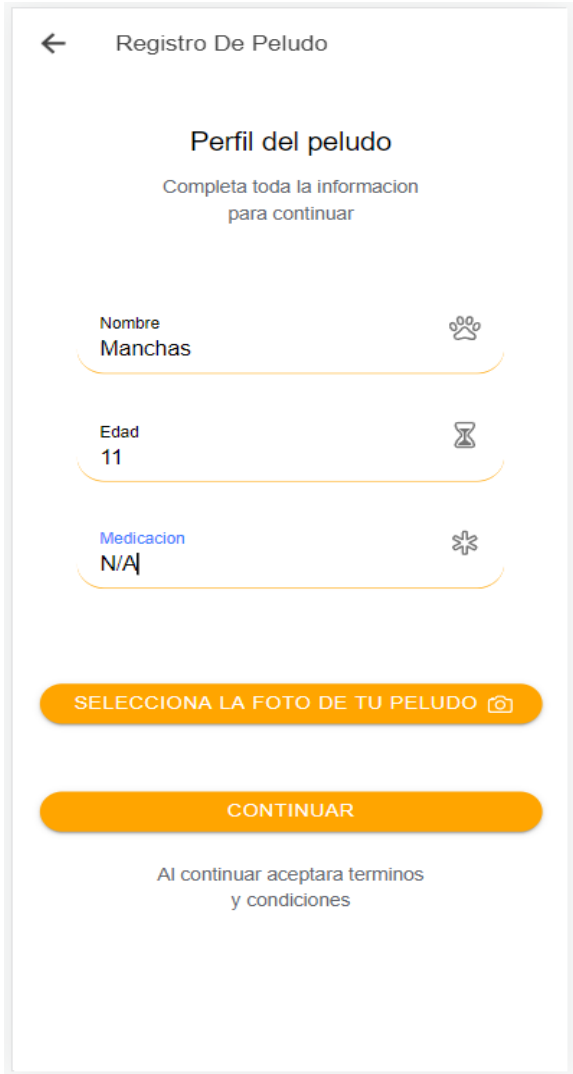
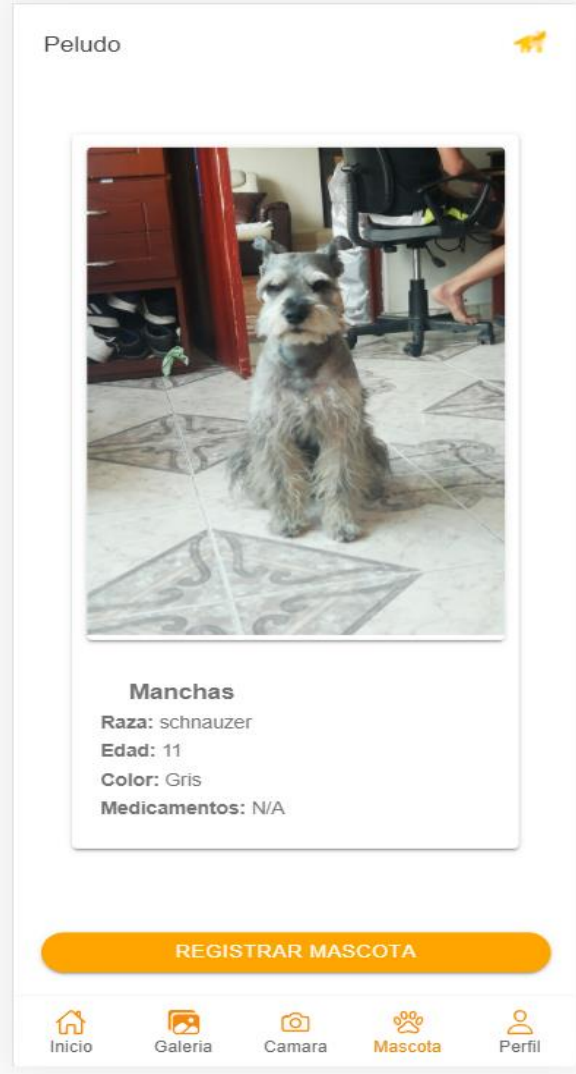
Pantalla registro perro	Pantalla mascota
	

Tabla 27

Registro perro 1, usuario 3

Pantalla registro perro	Pantalla mascota
	

### 10.1.4 Punto de verificación 4 – reconocimiento huella digital perro

En este punto se verificó que el usuario capturara una foto y al momento de dar clic en el botón analizar el algoritmo, la analizará e indicara si encontraba alguna coincidencia y de ser así, muestre la información del dueño del perro. Si no se encuentra ninguna coincidencia la aplicación generara un mensaje indicando que no se encontró un perro con esas características.


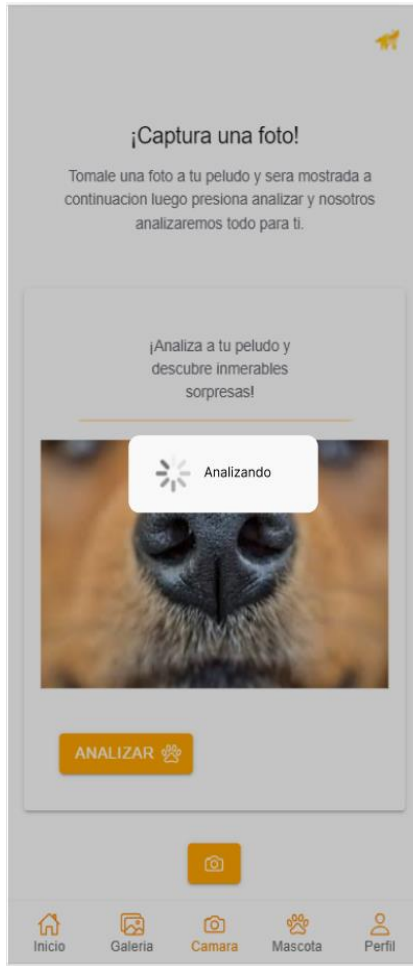
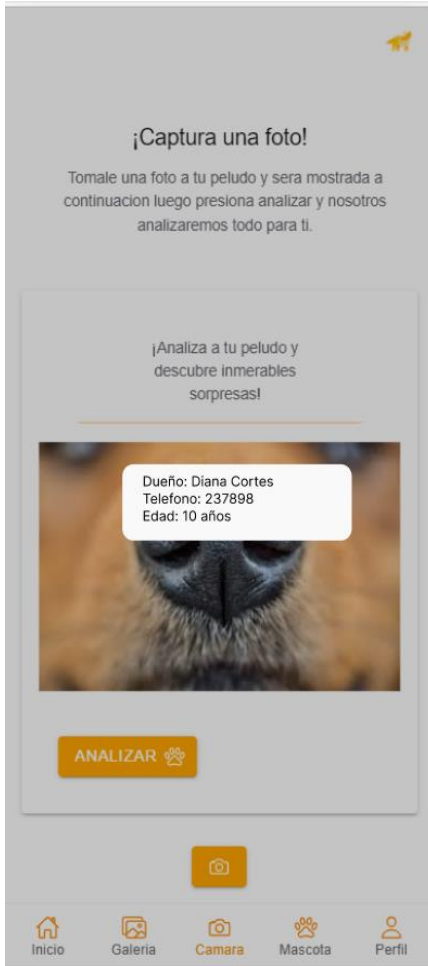
**Tabla 28**

*Análisis huella digital perro 1*

Pantalla cámara	Pantalla cámara, botón analizar	Pantalla cámara, resultado análisis

Tabla 29

## Análisis huella digital perro 2

Pantalla cámara	Pantalla cámara, botón analizar	Pantalla cámara, resultado análisis
		

**Tabla 30**

*Análisis huella digital perro 3*

Pantalla cámara	Pantalla cámara, botón analizar	Pantalla cámara, resultado análisis

### 10.1.5 Punto de verificación 5 – reconocimiento silueta perro

En este punto se verificó que el usuario capturara una foto y al dar clic en el botón de analizar el algoritmo identifique la raza del perro.

**Tabla 31**

*Análisis silueta perro 1*


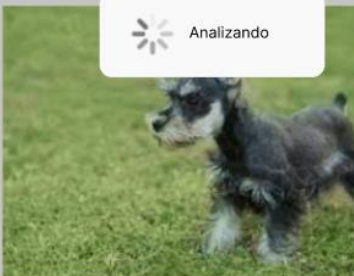

Pantalla cámara	Pantalla cámara, botón analizar	Pantalla cámara, resultado análisis
<p>¡Captura una foto!</p> <p>Tomale una foto a tu peludo y sera mostrada en la galeria. Luego presiona analizar y nosotros analizaremos todo para ti.</p> <p>¡Analiza a tu peludo y descubre innumerables sorpresas!</p>  <p>ANALIZAR 🐾</p> <p>Inicio   Galería   Camara   Mascota</p>	<p>¡Captura una foto!</p> <p>Tomale una foto a tu peludo y sera mostrada en la galeria. Luego presiona analizar y nosotros analizaremos todo para ti.</p> <p>¡Analiza a tu peludo y descubre innumerables sorpresas!</p> <p>Analizando</p>  <p>ANALIZAR 🐾</p> <p>Inicio   Galería   Camara   Mascota</p>	<p>¡Captura una foto!</p> <p>Tomale una foto a tu peludo y sera mostrada en la galeria. Luego presiona analizar y nosotros analizaremos todo para ti.</p> <p>¡Analiza a tu peludo y descubre innumerables sorpresas!</p> <p>Raza: Schanauzer</p>  <p>ANALIZAR 🐾</p> <p>Inicio   Galería   Camara   Mascota</p>

Tabla 32

## Análisis silueta perro 2

Pantalla cámara, resultado análisis

Pantalla cámara

Pantalla cámara, botón analizar

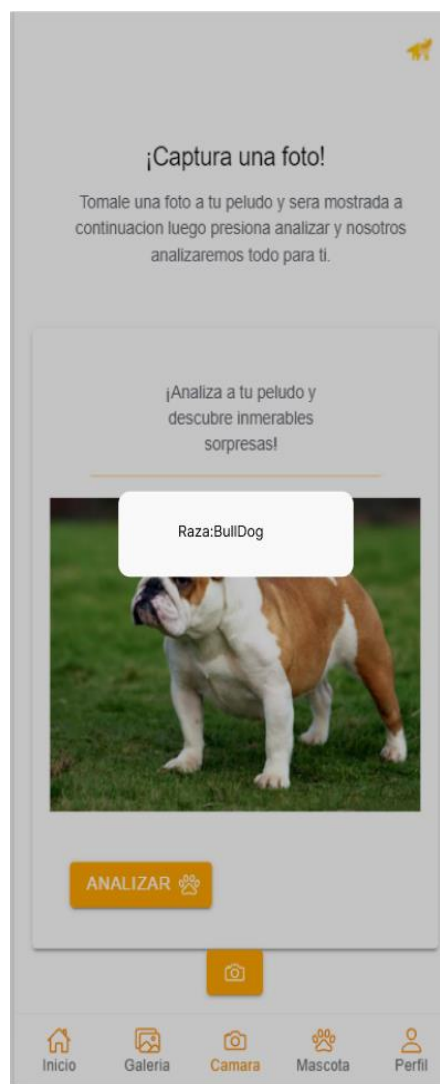
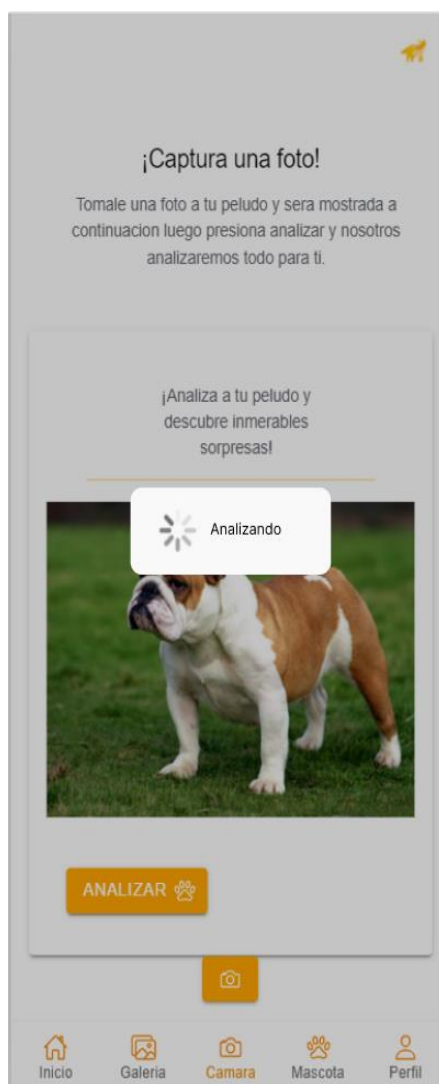


Tabla 33

## Análisis silueta perro 3



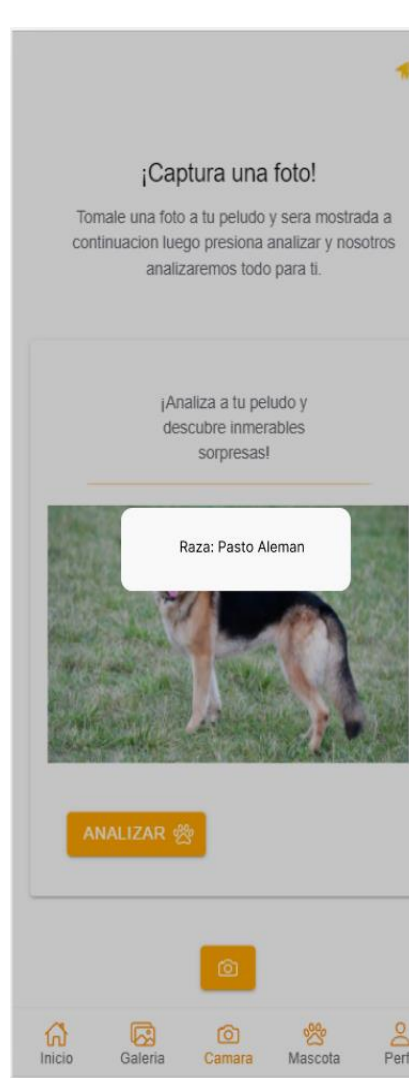
Pantalla cámara	Pantalla cámara, botón analizar	Pantalla cámara, resultado análisis
		



Tabla 34

## Análisis silueta perro 4

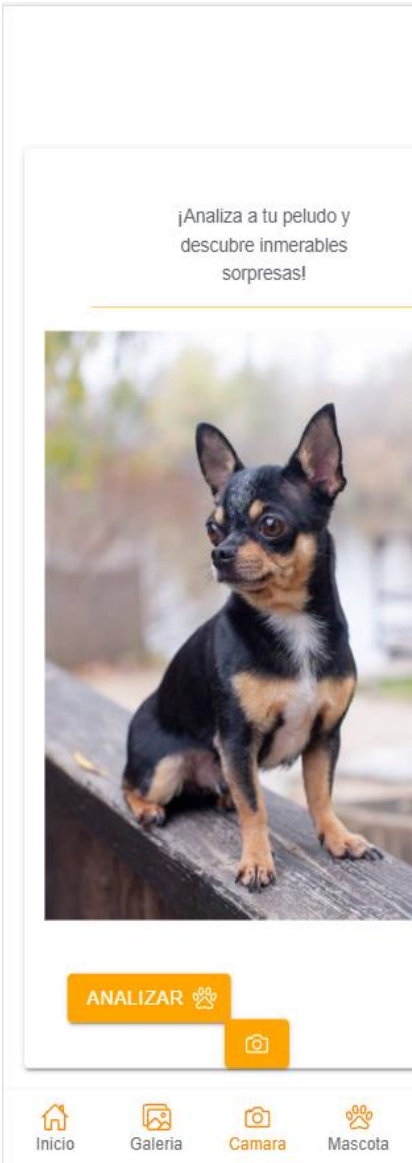
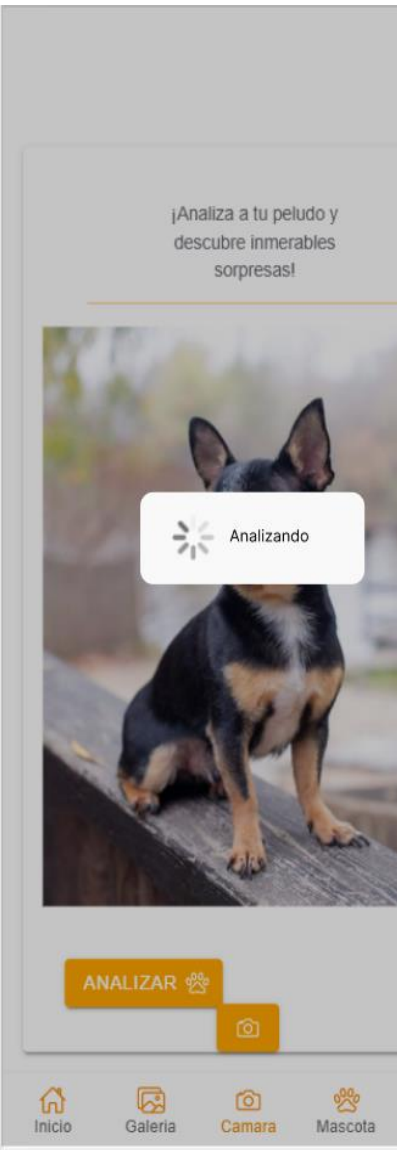
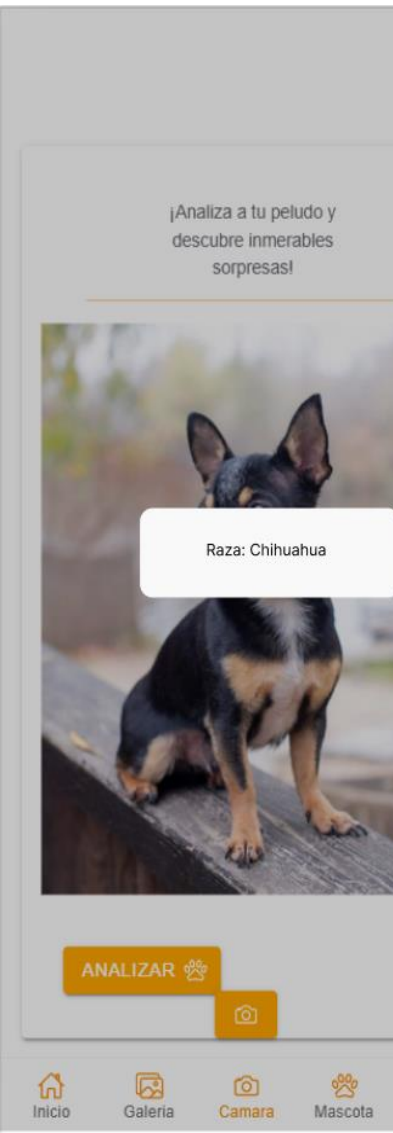
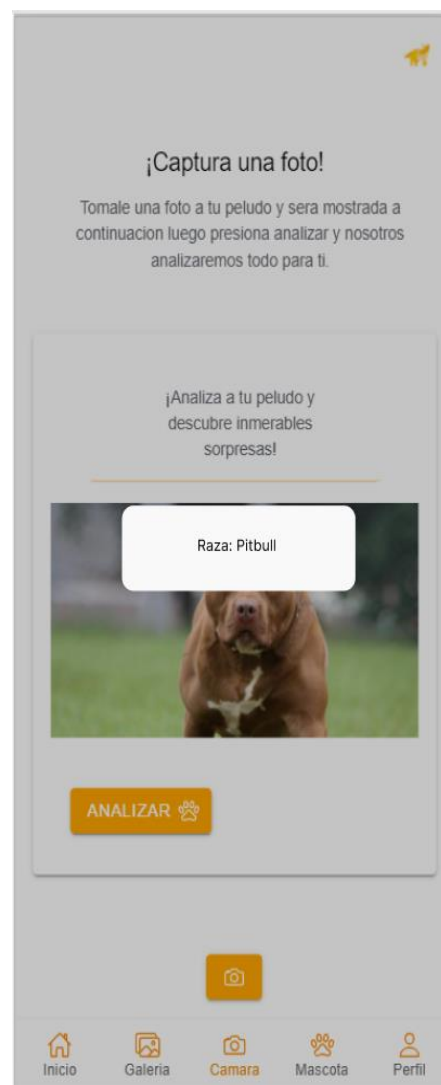
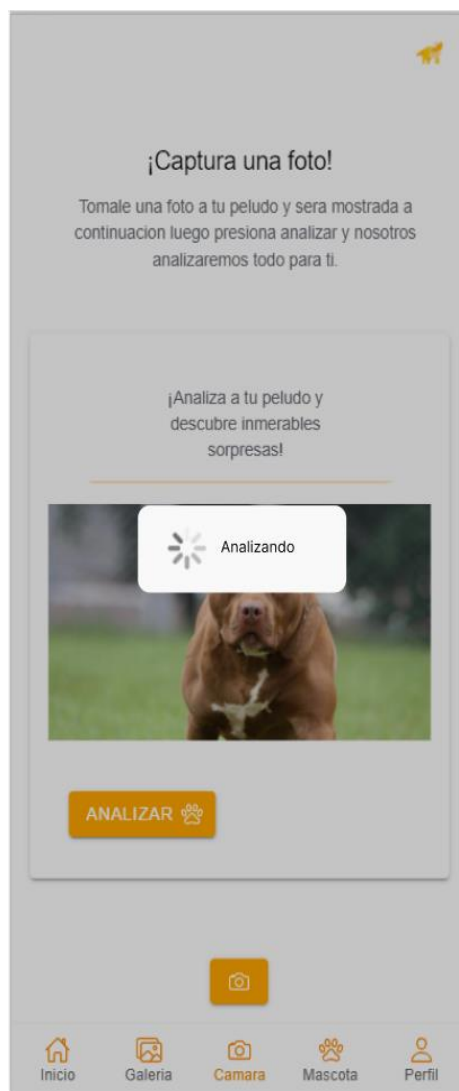
Pantalla cámara	Pantalla cámara, botón analizar	Pantalla cámara, resultado análisis
		

Tabla 35

## Análisis silueta perro 5

Pantalla cámara

Pantalla cámara, botón analizar

Pantalla cámara, resultado  
análisis

### 10.1.6 Punto de verificación 6 – actualización datos usuario

En este punto se verificó que el usuario lograra actualizar sus datos personales. En este formulario se validan la estructura de los campos, por ejemplo, cédula, edad y teléfono sean numéricos, si el usuario digital incorrectamente uno de estos campos la aplicación generará un mensaje de error indicando que el campo no es válido.

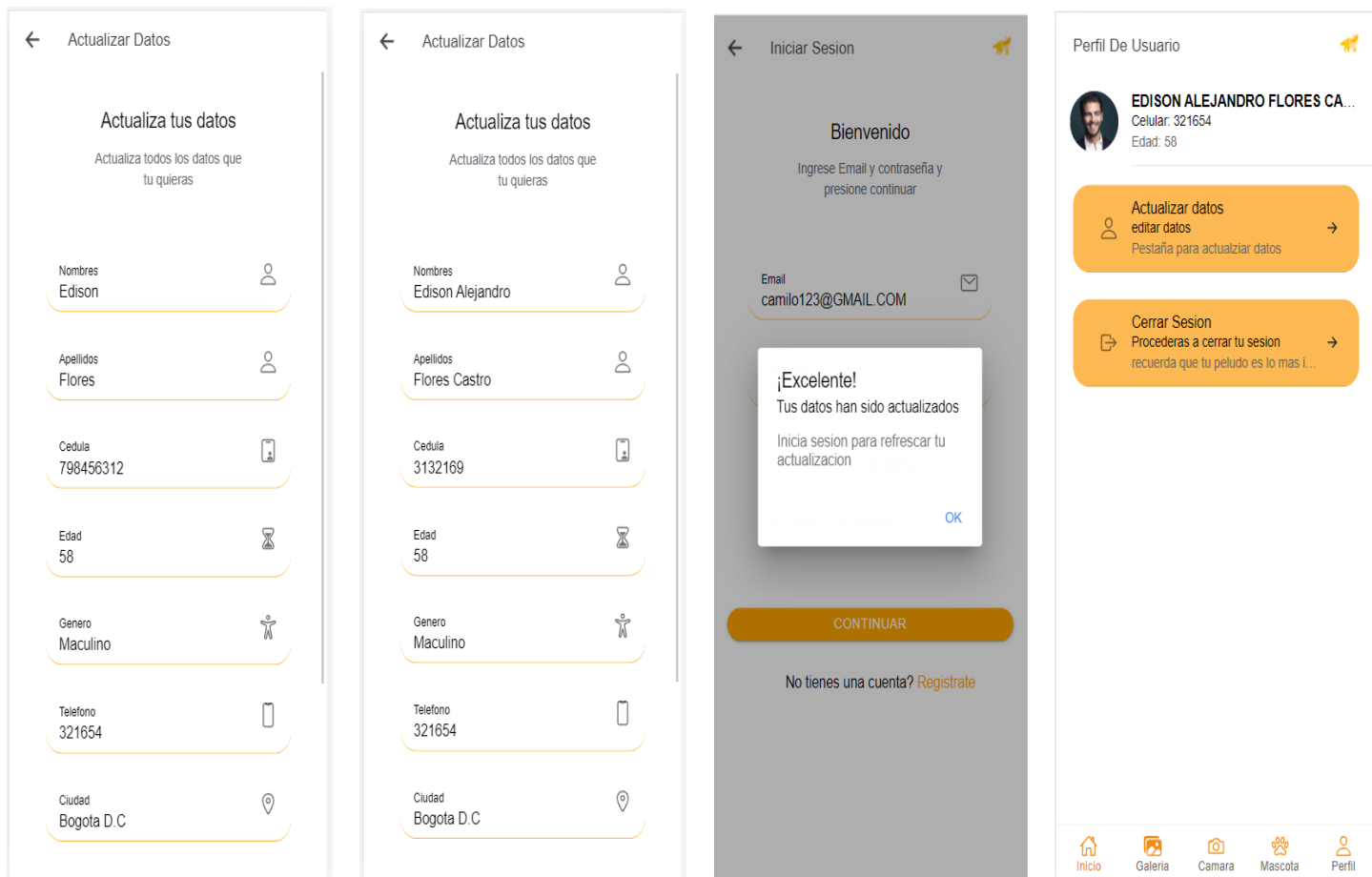
#### **Tabla 36**

##### *Actualización datos usuario 1*

---

Pantalla actualización datos	Pantalla actualización datos	Pantalla actualización de datos exitosa	Pantalla perfil
---------------------------------	---------------------------------	--	-----------------

---



### 10.1.7 Punto de verificación 7 – actualización datos perro

En este punto se verificó que el usuario logre actualizar los datos de su perro.

**Tabla 37**

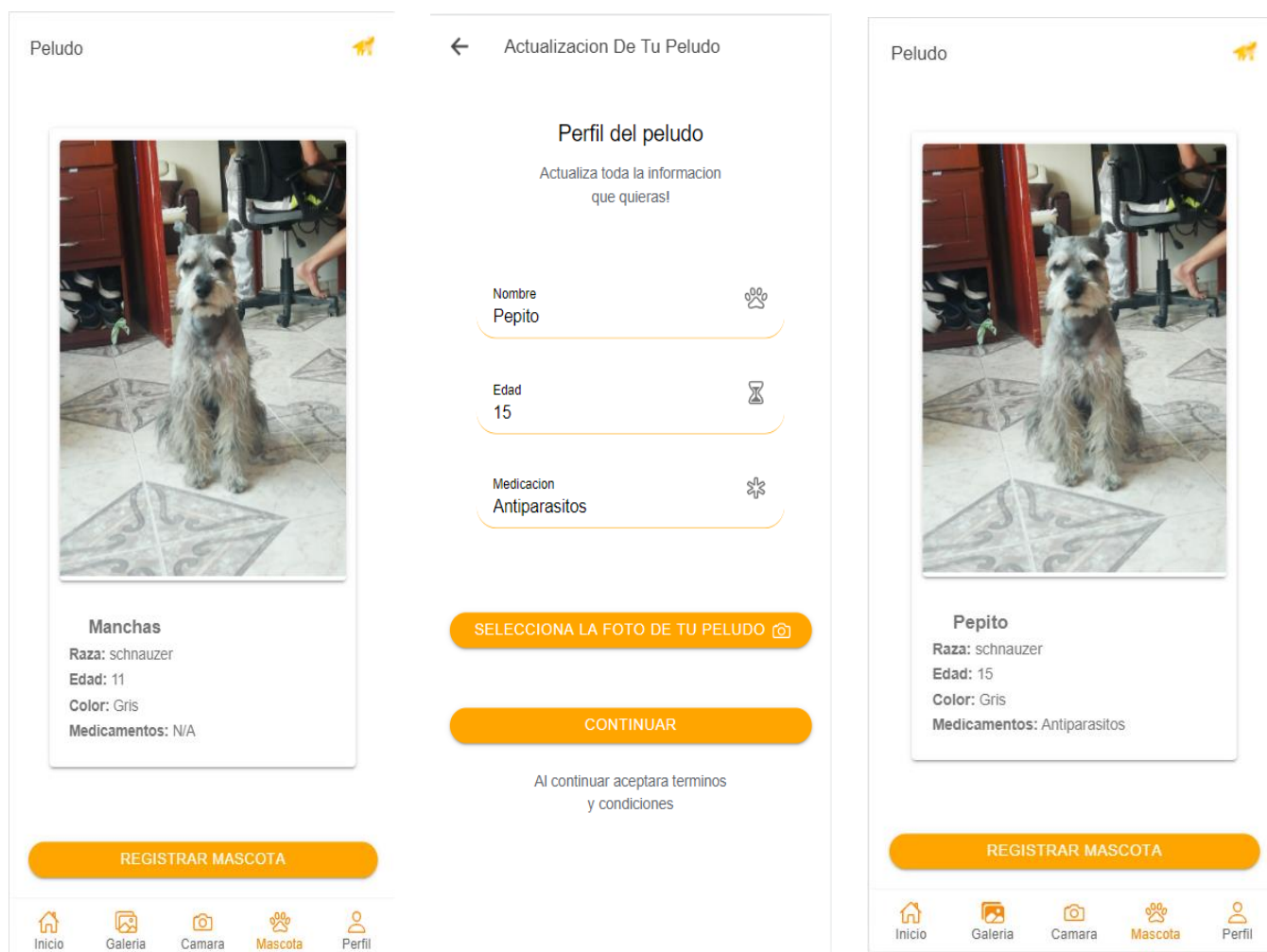
*Actualización datos perro, usuario 1*

Pantalla mascota

Pantalla actualización datos

Pantalla mascota

mascota



## 11 Análisis y Resultados

De acuerdo con el objetivo general del Proyecto, en el presente capítulo se expone el proceso de construcción y entrenamiento de los algoritmos de IA integrados en la aplicación y se muestran los resultados obtenidos de la implementación del Sistema inteligente para el reconocimiento e identificación de la huella digital del perro y su silueta.

Para el entrenamiento, verificación, análisis y pruebas de los algoritmos de IA se utilizó la herramienta Colaboratory (Colab), producto de Google Research que permite escribir y ejecutar código Python en el navegador, es especialmente adecuado para tareas de aprendizaje automático, análisis de datos y educación. Técnicamente, Colab es un servicio de cuaderno alojado de Jupyter que no requiere configuración y que ofrece acceso sin costo a recursos informáticos (Google, 2022).

### **Reconocimiento de la silueta**

El tipo de aprendizaje (aprendizaje supervisado) y algoritmo de clasificación con una red neuronal convolucional que se utilizó en el proyecto, requiere que se proporcionen como entrada la siguiente información:

- **Set de entrenamiento:** imágenes clasificadas con 120 razas de perros diferentes.
- **Set de pruebas:** imágenes sin clasificar para realizar predicciones cuando se entrene el algoritmo.
- **Archivo CSV etiquetas:** en este archivo se encuentran etiquetadas todas las imágenes del set de entrenamiento, esto para que el algoritmo infiera qué imagen corresponde a cada raza de perro.

Cabe aclarar que se utilizó un modelo preentrenado de la biblioteca TensorFlow.

#### **11.1.1 Preparación de los datos**

En este punto del proyecto, se inicia con la preparación de los datos y se valida que las etiquetas del archivo se encuentren correctas y se carga al editor.

**Imagen 49**

*Preparación de los datos, etiquetas archivo*

```
count          id          breed
10222          10222      10222
unique         10222      120
top  000bec180eb18c7604dcecc8fe0dba07  scottish_deerhound
freq          1          126
```

*Nota.* La imagen ilustra la salida por consola que genero la preparación de los datos con las etiquetas definidas en el archivo. Creación propia, 2022

Como se mencionó anteriormente, el set de entrenamiento contó con 120 tipos de razas de perros diferentes y en su totalidad se tenían 10222 imágenes, entre más imágenes etiquetadas se le proporcionen al algoritmo más eficiente es y su intervalo de confianza se acerca más al punto óptimo.

Posteriormente, se realiza un conteo de la cantidad de imágenes etiquetadas según raza.

**Imagen 50***Preparación de los datos, conteo por raza*

```

scottish_deerhound      126
maltese_dog            117
afghan_hound           116
entlebucher           115
bernese_mountain_dog   114
...
golden_retriever        67
brabancon_griffon      67
komondor                67
eskimo_dog              66
briard                  66
Name: breed, Length: 120, dtype: int64

```

*Nota.* La imagen ilustra la salida por consola que genero la preparación de los datos realizando el conteo de las razas. Creación propia, 2022

Se realizó un conteo de imágenes por raza con el objeto de determinar la media en cuanto a número de imágenes por raza, y se obtuvieron 82. Se realiza una creación de los nombres de las rutas donde se encuentra almacenada cada imagen a partir de su ID.

**Imagen 51***Construcción de rutas, imagenes de entrenamiento*

```

['drive/My Drive/Proyecto_Grado/Clasificador_Raza/train/000bec180eb18c7604dcecc8fe0dba07.jpg',
'drive/My Drive/Proyecto_Grado/Clasificador_Raza/train/001513dfcb2ffafc82cccf4d8bbaba97.jpg',
'drive/My Drive/Proyecto_Grado/Clasificador_Raza/train/001cdf01b096e06d78e9e5112d419397.jpg',
'drive/My Drive/Proyecto_Grado/Clasificador_Raza/train/00214f311d5d2247d5dfe4fe24b2303d.jpg',
'drive/My Drive/Proyecto_Grado/Clasificador_Raza/train/0021f9ceb3235effd7fcde7f7538ed62.jpg',
'drive/My Drive/Proyecto_Grado/Clasificador_Raza/train/002211c81b498ef88e1b40b9abf84e1d.jpg',
'drive/My Drive/Proyecto_Grado/Clasificador_Raza/train/00290d3e1fdd27226ba27a8ce248ce85.jpg',
'drive/My Drive/Proyecto_Grado/Clasificador_Raza/train/002a283a315af96eaea0e28e7163b21b.jpg',
'drive/My Drive/Proyecto_Grado/Clasificador_Raza/train/003df8b8a8b05244b1d920bb6cf451f9.jpg',
'drive/My Drive/Proyecto_Grado/Clasificador_Raza/train/0042188c895a2f14ef64a918ed9c7b64.jpg']

```

*Nota.* La imagen ilustra la salida por consola que genero la construcción de las rutas para las imágenes de entrenamiento. Creación propia, 2022



A continuación, se preparan todas las etiquetas y se realizan una serie de validaciones para comprobar que no existan archivos perdidos y/o corruptos.

### *Imagen 52*

#### *Preparación de etiquetas*

```
0          boston_bull
1          dingo
2          pekinese
3          bluetick
4          golden_retriever
...
10217         borzoi
10218         dandie_dinmont
10219         airedale
10220         miniature_pinscher
10221         chesapeake_bay_retriever
Name: breed, Length: 10222, dtype: object
```

*Nota.* La imagen ilustra la salida por consola que se generó al realizar la preparación de las etiquetas. Creación propia, 2022

Se convierte cada etiqueta en una matriz booleana, como se nota en la siguiente imagen.

### Imagen 53

#### Conversión etiquetas en matrices booleanas

```
[array([False, False, False, False, False, False, False, False, False,
        False, False, False, False, False, False, False, False, False,
        False, True, False, False, False, False, False, False, False,
        False, False, False, False, False, False, False, False, False,
        False, False, False, False, False, False, False, False, False,
        False, False, False, False, False, False, False, False, False,
        False, False, False, False, False, False, False, False, False,
        False, False, False, False, False, False, False, False, False,
        False, False, False, False, False, False, False, False, False,
        False, False, False, False, False, False, False, False, False,
        False, False, False, False, False, False, False, False, False,
        False, False, False]),
 array([False, False, False, False, False, False, False, False, False,
        False, False, False, False, False, False, False, False, False,
        False, False, False, False, False, False, False, False, False,
        False, True, False, False, False, False, False, False, False,
        False, False, False, False, False, False, False, False, False,
        False, False, False, False, False, False, False, False, False,
        False, False, False, False, False, False, False, False, False,
        False, False, False, False, False, False, False, False, False,
        False, False, False, False, False, False, False, False, False,
        False, False, False, False, False, False, False, False, False,
        False, False, False, False, False, False, False, False, False,
        False, False, False, False, False, False, False, False, False,
        False, False, False])]
```

*Nota.* La imagen ilustra la salida por consola al realizar la conversión de las etiquetas en matrices booleanas. Creación propia, 2022

#### 11.1.2 Creación del set de validación

En este apartado, se configuran variables y se define un número inicial de imágenes para entrenar, luego, se dividen los datos para su procesamiento. Haciendo uso de la librería de TensorFlow, se convierten las imágenes JPG a tensores, se normalizan las imágenes convirtiéndolas del canal de color de 0 – 255 a 0 – 1 y por último, se les cambia el tamaño a 224 x 224 píxeles.

Los datos se procesan por lotes, por ende, se realiza un proceso para la estructuración de los datos en lotes.

**Imagen 54***Conversión de datos en lotes*

```

(<tf.Tensor: shape=(224, 224, 3), dtype=float32, numpy=
array([[0.00784314, 0.00392157, 0.          ],
       [0.00392157, 0.00784314, 0.          ],
       [0.00392157, 0.00784314, 0.          ],
       ...,
       [0.39637563, 0.34860766, 0.5459306 ],
       [0.38927397, 0.39288273, 0.43594864],
       [0.5954294 , 0.62155867, 0.598073  ]]),

[[0.00784314, 0.00392157, 0.          ],
 [0.00784314, 0.00392157, 0.          ],
 [0.00392157, 0.00784314, 0.          ],
 ...,
 [0.6178841 , 0.5807307 , 0.71904504],
 [0.5324265 , 0.5426441 , 0.58257955],
 [0.5685837 , 0.59211314, 0.58913547]],

[[0.01321779, 0.00123424, 0.          ],
 [0.00784314, 0.00392157, 0.          ],
 [0.00392157, 0.00784314, 0.          ],
 ...,
 [0.9097923 , 0.89815587, 0.95931107],
 [0.5557031 , 0.56761146, 0.59791577],
 [0.5263359 , 0.5498653 , 0.54780173]],

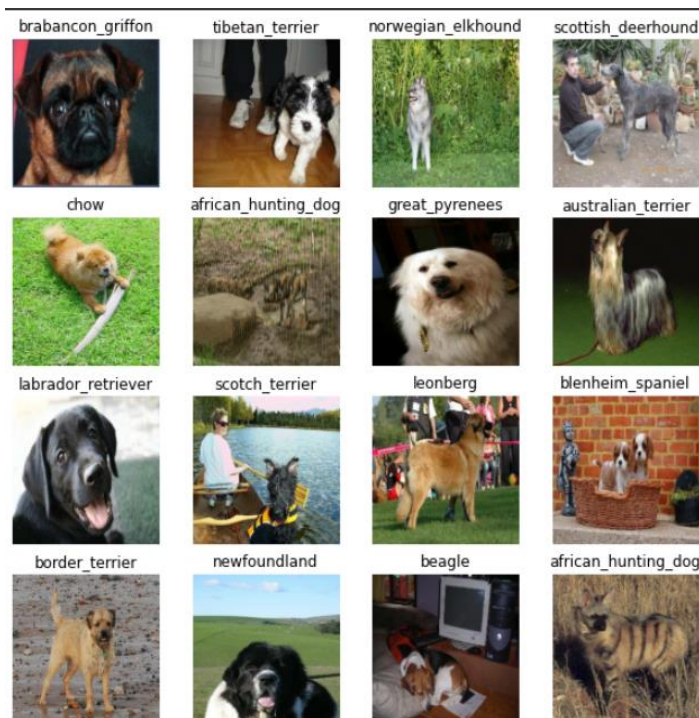
```

*Nota.* La imagen ilustra la salida por consola al realizar la conversión de los datos en lotes.

Creación propia, 2022

## Imagen 55

### Visualización del set de validación



*Nota.* La imagen ilustra una parte de las imágenes tomadas para el set de validación. Creación propia, 2022

### 11.1.3 Construcción del modelo

Para este entrenamiento se utilizó el modelo *Imagenet (ILSVRC-2012-CLS) classification with MobileNet V2 (depth multiplier 1.30)*. Posteriormente, se importa el modelo, pero antes de procesar y entrenar el mismo se debe definir la forma de la imagen en forma de tensores y la forma de la salida en forma de tensores.

## Imagen 56

### Función para crear el modelo

```
def create_model(input_shape=INPUT_SHAPE, output_shape=OUTPUT_SHAPE, model_url=MODEL_URL):
    """
    Creates of specified input shape, output shape, with specified
    URL from TensorFlow Hub
    """
    print("Building model with:", model_url)

    # Configurar capas de modelo
    model = tf.keras.Sequential([
        hub.KerasLayer(model_url), # Capa 1 (capa de entrada)
        tf.keras.layers.Dense(units=output_shape,
                               activation="softmax") # Capa 2 (capa de salida)
    ])

    # Compilar modelo
    model.compile(
        loss=tf.keras.losses.CategoricalCrossentropy(),
        optimizer=tf.keras.optimizers.Adam(),
        metrics=["accuracy"]
    )

    # Construir el modelo
    model.build(INPUT_SHAPE)

    return model
```

*Nota.* La imagen ilustra un fragmento de código de la función que crea el modelo. Creación propia, 2022

Adicionalmente, se crearon los *callbacks*. Estos, son funciones de ayuda que un modelo puede usar durante el entrenamiento para hacer cosas como guardar su progreso, verificar su progreso o detener el entrenamiento temprano si un modelo deja de mejorar.

Los siguientes, son los callbacks creados:

- Uno para TensorBoard que ayuda a rastrear el progreso del modelo.
- Otro para dejar de entrenar antes de tiempo para evitar el sobreajuste

### 11.1.4 Entrenamiento

Teniendo todos los antecedentes mencionados anteriormente, se procede a entrenar el modelo, como se muestra en la siguiente ilustración.

### *Imagen 57*

#### *Función para entrenar el modelo*

```
def train_model():
    """
    Trains a given model and returns the trained version.
    """
    # Crear el modelo
    model = create_model()

    # Crear un nuevo TensorBoard sesión cada vez que entrenamos a un modelo
    tensorboard = create_tensorboard_callback()

    # Ajustar el modelo a los datos
    model.fit(x=train_data,
              epochs=NUM_EPOCHS,
              validation_data=val_data,
              validation_freq=1,
              callbacks=[tensorboard, early_stopping])

    # Retornar el modelo ajustado
    return model
```

*Nota.* La imagen ilustra un fragmento de código de la función que entrena el modelo. Creación propia, 2022

Al ejecutar la función y realizar el entrenamiento del mismo, se observa el resultado.

## Imagen 58

### Entrenamiento del modelo

```
Building model with: https://tfhub.dev/google/imagenet/mobilenet\_v2\_130\_224/classification/5
Epoch 1/100
25/25 [=====] - 103s 4s/step - loss: 4.7277 - accuracy: 0.0900 - val_loss: 3.3941 - val_accuracy: 0.2650
Epoch 2/100
25/25 [=====] - 39s 2s/step - loss: 1.7104 - accuracy: 0.6800 - val_loss: 2.0588 - val_accuracy: 0.5150
Epoch 3/100
25/25 [=====] - 38s 2s/step - loss: 0.5720 - accuracy: 0.9450 - val_loss: 1.5741 - val_accuracy: 0.5800
Epoch 4/100
25/25 [=====] - 40s 2s/step - loss: 0.2533 - accuracy: 0.9862 - val_loss: 1.4316 - val_accuracy: 0.6150
Epoch 5/100
25/25 [=====] - 40s 2s/step - loss: 0.1468 - accuracy: 0.9987 - val_loss: 1.3626 - val_accuracy: 0.6250
Epoch 6/100
25/25 [=====] - 41s 2s/step - loss: 0.1014 - accuracy: 1.0000 - val_loss: 1.3063 - val_accuracy: 0.6200
Epoch 7/100
25/25 [=====] - 38s 2s/step - loss: 0.0759 - accuracy: 1.0000 - val_loss: 1.2716 - val_accuracy: 0.6350
```

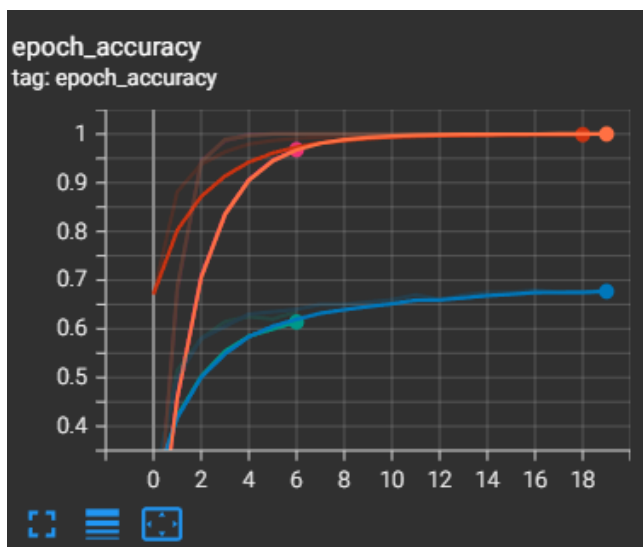
*Nota.* La imagen ilustra la salida por consola al ejecutar la función de entrenamiento del modelo.

Creación propia, 2022

El entrenamiento del modelo en este caso, tomó una duración de **2 h 26m 18s**, la cual arrojó como resultado un intervalo de confianza del **99%**. Con la ayuda de TensorBoard se graficaron los resultados de la etapa de entrenamiento. (Ver Ilustraciones 33 y 34)

**Imagen 59**

Gráfico de análisis de precisión según etapa



*Nota.* En la imagen de ilustra el comportamiento de cada una de las etapas involucradas en el proceso (entrenamiento y validación). Creación propia, 2022

**Imagen 60**

Etapas de entrenamiento

Name	Smoothed	Value	Step	Time	Relative
20210815-151424/train	1	1	19	Sun Aug 15, 10:34:43	18m 3s
20210815-151424/validation	0.6769	0.68	19	Sun Aug 15, 10:34:43	18m 3s
20210815-154252/train	0.9987	0.9987	18	Sun Aug 15, 13:29:01	2h 26m 18s
20210815-205137/train	0.9679	1	6	Sun Aug 15, 15:58:01	4m 40s
20210815-205137/validation	0.6142	0.635	6	Sun Aug 15, 15:58:01	4m 40s

*Nota.* En la imagen se ilustran los resultados obtenidos de cada una de las etapas (entrenamiento y validación). Creación propia, 2022



### 11.1.5 Validación

Con el set de validación definido se realiza una verificación y pruebas del modelo para determinar que es óptimo su funcionamiento. A continuación, se ilustrarán algunos ejemplos de las pruebas que se realizaron, en el título de cada imagen hay dos etiquetas, etiqueta predicción con su % de precisión y en seguida se encuentra la etiqueta original que tenía la imagen.

**Tabla 38**

*Validación set de entrenamiento*



---

miniature\_pinscher 77% miniature\_pinscher



kerry\_blue\_terrier 81% kerry\_blue\_terrier



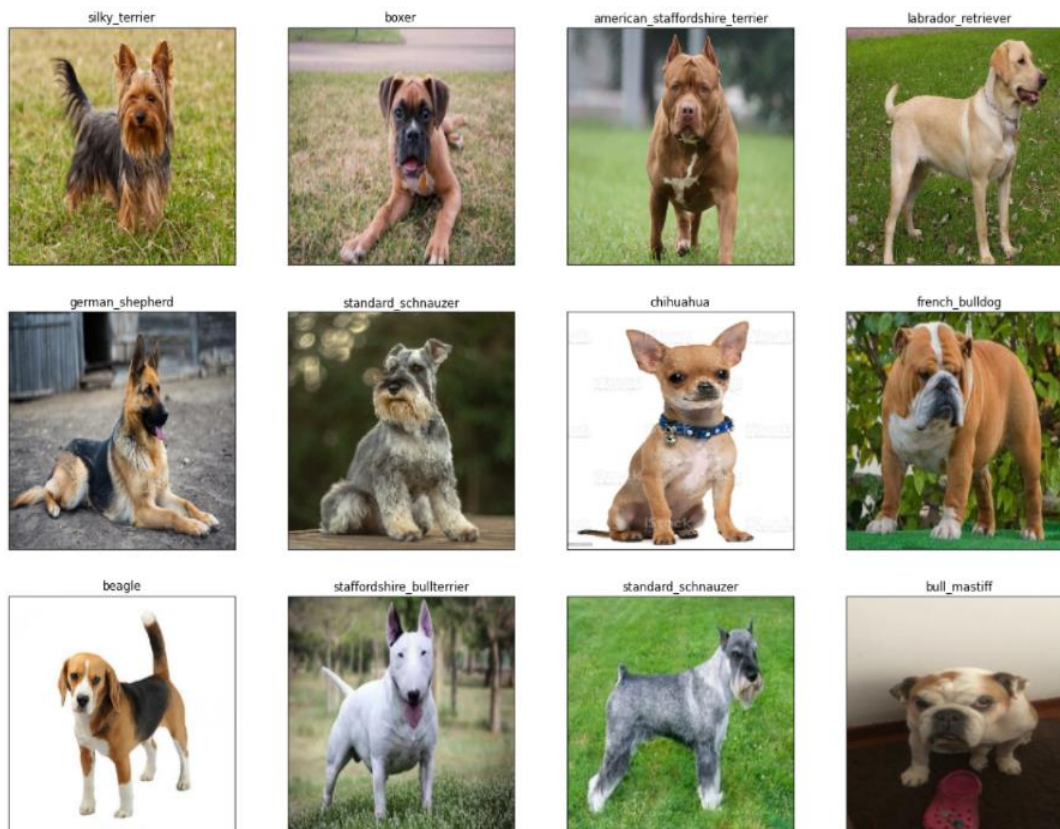
---

*Nota.* En la parte superior de cada imagen se encuentran dos etiquetas. Etiqueta con el porcentaje de precisión, etiqueta real.

Pruebas con las razas puntuales definidas en el alcance del proyecto.

## Imagen 61

### Pruebas con algunas razas



*Nota.* Creación propia, 2022

Inicialmente, para estas pruebas del modelo se habían definido 10 razas, se superaron las expectativas que se definieron inicialmente, en el alcance del proyecto se definió que el modelo se especializaría en reconocer 10 razas, luego, al realizar todo el proceso anteriormente descrito y tener una precisión del 99% se logró determinar gracias a las predicciones que se realizaron sobre el set de validación, que el modelo es capaz de reconocer e identificar la raza de 120 razas distintas, lo cual es un resultado excepcional para el planteamiento que se tenía inicialmente.

## Reconocimiento huella digital

El tipo de aprendizaje (aprendizaje supervisado) y algoritmo de clasificación con una red neuronal convolucional utilizado en el proyecto, requiere que se proporcionen como entrada la siguiente información:

- **Set de entrenamiento:** 1393 imágenes clasificadas de distintos perros.
- **Set de pruebas:** imágenes sin clasificar para realizar predicciones cuando se entrene el algoritmo.

### 11.1.6 Preparación de los datos

Inicialmente se parametrizan las rutas en donde se encuentran las imágenes, donde se va a guardar el log de entrenamiento y el modelo entrenado a utilizar dentro del desarrollo.

#### *Imagen 62*

*Definición rutas previas al entrenamiento*

```

PATH      = '/content/drive/My Drive/Proyecto_Grado/Identificacion_Huella/Imagenes'      # Path to the directory of the saved dataset
PATH_SAVE = '/content/drive/My Drive/Proyecto_Grado/Identificacion_Huella/Historial/'    # Path to the directory where the history will be stored
PATH_MODEL = '/content/drive/My Drive/Proyecto_Grado/Identificacion_Huella/Modelo/'    # Path to the directory where the model will be stored
SIZE      = (224,224,3)                      # Size of the input images
TEST_SPLIT = 0.1                             # Train/test ratio

LOAD_NET  = False                             # Load a network from a saved model? If True NET_NAME and START_EPOCH have to be precised
NET_NAME  = '2019.05.12.dogfacenet'           # Network saved name
START_EPOCH = 0                               # Start the training at a specified epoch
NBOF_EPOCHS = 250                             # Number of epoch to train the network
HIGH_LEVEL = True                             # Use high level training ('fit' keras method)
STEPS_PER_EPOCH = 300                         # Number of steps per epoch
VALIDATION_STEPS = 30                         # Number of steps per validation

```

*Nota.* La imagen ilustra un fragmento de código donde se configuran las rutas iniciales y parametrización general del algoritmo. Creación propia, 2022

### 11.1.7 Construcción del modelo

Para este entrenamiento no se utilizó ningún modelo pre entrenado, todo el algoritmo se desarrolló desde su inicio. A continuación, se ilustra una parte de la creación del modelo.

#### *Imagen 63*

#### *Definición del modelo, reconocimiento huella*

```

if LOAD_NET:
    print('Loading model from {:s}{:s}{:d}.h5 ...'.format(PATH_MODEL,NET_NAME,START_EPOCH))

    model = tf.keras.models.load_model(
        '{:s}{:s}{:d}.h5'.format(PATH_MODEL,NET_NAME,START_EPOCH),
        custom_objects={'triplet':triplet,'triplet_acc':triplet_acc})

    print('Done.')
else:
    from tensorflow.keras import Model
    from tensorflow.keras.layers import Input, Conv2D, MaxPooling2D, Add, GlobalAveragePooling2D, DepthwiseConv2D
    from tensorflow.keras.layers import Activation, Dropout, Flatten, Dense, Lambda, BatchNormalization

    """
    Model number 12: Paper version: a modified ResNet with Dropout layers and without bottleneck layers
    """

    print('Defining model {:s} ...'.format(NET_NAME))

    emb_size = 32

    inputs = Input(shape=SIZE)

    x = Conv2D(16, (7, 7), (2, 2), use_bias=False, activation='relu', padding='same')(inputs)
    x = BatchNormalization()(x)
    x = MaxPooling2D((3,3))(x)

    for layer in [16,32,64,128,512]:

        x = Conv2D(layer, (3, 3), strides=(2,2), use_bias=False, activation='relu', padding='same')(x)
        r = BatchNormalization()(x)

        x = Conv2D(layer, (3, 3), use_bias=False, activation='relu', padding='same')(r)
        x = BatchNormalization()(x)
        r = Add()([r,x])

        x = Conv2D(layer, (3, 3), use_bias=False, activation='relu', padding='same')(r)
        x = BatchNormalization()(x)
        x = Add()([r,x])

    x = GlobalAveragePooling2D()(x)
    x = Flatten()(x)
    x = Dropout(0.5)(x)
    x = Dense(emb_size, use_bias=False)(x)
    outputs = Lambda(lambda x: tf.nn.l2_normalize(x,axis=-1))(x)

    model = tf.keras.Model(inputs,outputs)

    model.compile(loss=triplet,
                  optimizer='adam',
                  metrics=[triplet_acc])

    print('Done.')

```

*Nota.* La imagen ilustra un fragmento de código de la función que define el modelo de reconocimiento de la huella, Creación propia, 2022

### 11.1.8 Entrenamiento

Teniendo en cuenta los elementos anteriores, se procede a entrenar el modelo.

## Imagen 64

*Función que entrena el modelo, reconocimiento huella*

```
# Create saving folders
if not os.path.isdir(PATH_MODEL):
    os.makedirs(PATH_MODEL)
if not os.path.isdir(PATH_SAVE):
    os.makedirs(PATH_SAVE)

# Bug fixed: keras models are to be initialized by a training on a single batch
for images_batch, labels_batch in online_adaptive_hard_image_generator(
    filenames_train,
    labels_train,
    model,
    crt_acc,
    batch_size,
    nbof_subclasses=nbof_subclasses):
    h = model.train_on_batch(images_batch, labels_batch)
    break

for i in range(START_EPOCH, START_EPOCH + NBOF_EPOCHS):
    print("Beginning epoch number: " + str(i))

    hard_triplet_ratio = np.exp(-crt_loss * 10 / batch_size)
    nbof_hard_triplets = int(batch_size // 3 * hard_triplet_ratio)

    print("Current hard triplet ratio: " + str(hard_triplet_ratio))

    histories += [model.fit_generator(
        online_adaptive_hard_image_generator(filenames_train, labels_train, model, crt_loss, batch_size, nbof_subclasses=nbof_subclasses),
        steps_per_epoch=STEPS_PER_EPOCH,
        epochs=1,
        validation_data=image_generator(filenames_test, labels_test, batch_size, use_aug=False),
        validation_steps=VALIDATION_STEPS)]

    crt_loss = histories[-1].history['loss'][0]
    crt_acc = histories[-1].history['triplet_acc'][0]

    # Save model
    model.save('{:s}{:s}.{:d}.h5'.format(PATH_MODEL, NET_NAME, i))

    # Save history
    loss = np.empty(0)
    val_loss = np.empty(0)
    acc = np.empty(0)
    val_acc = np.empty(0)

    for history in histories:
        loss = np.append(loss, history.history['loss'])
        val_loss = np.append(val_loss, history.history['val_loss'])
        acc = np.append(acc, history.history['triplet_acc'])
        val_acc = np.append(val_acc, history.history['val_triplet_acc'])

    history_ = np.array([loss, val_loss, acc, val_acc])
    np.save('{:s}{:s}.{:d}.npy'.format(PATH_SAVE, NET_NAME, i), history_)
```

*Nota.* La imagen ilustra un fragmento de código de la función que entrena el modelo de reconocimiento de la huella. Creación propia, 2022

Se ejecuta la función y se realiza el entrenamiento del mismo, como se nota en la siguiente Imagen.

### *Imagen 65*

#### *Entrenamiento del modelo*

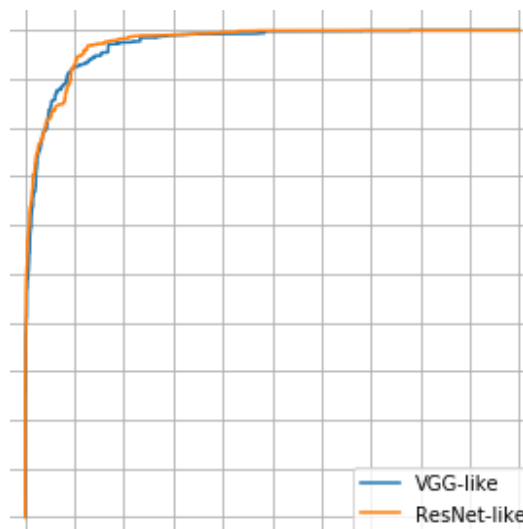
```

Beginning epoch number: 1
Current hard triplet ratio: 0.1439600357740038
300/300 [=====] - 481s 2s/step - loss: 3.2250 - triplet_acc: 0.5817 - val_loss: 2.2792 - val_triplet_acc: 0.6900
Beginning epoch number: 2
Current hard triplet ratio: 0.3412945384205913
300/300 [=====] - 532s 2s/step - loss: 3.8207 - triplet_acc: 0.5180 - val_loss: 2.0703 - val_triplet_acc: 0.6567
Beginning epoch number: 3
Current hard triplet ratio: 0.2798306676796238
300/300 [=====] - 488s 2s/step - loss: 3.0139 - triplet_acc: 0.5547 - val_loss: 1.4317 - val_triplet_acc: 0.6700
Beginning epoch number: 4
Current hard triplet ratio: 0.3661765717101257
300/300 [=====] - 532s 2s/step - loss: 3.2992 - triplet_acc: 0.4120 - val_loss: 1.3966 - val_triplet_acc: 0.6033
Beginning epoch number: 5
Current hard triplet ratio: 0.3329568910722319
300/300 [=====] - 524s 2s/step - loss: 2.4716 - triplet_acc: 0.3527 - val_loss: 1.2533 - val_triplet_acc: 0.5433
Beginning epoch number: 6
Current hard triplet ratio: 0.43872903822054743
300/300 [=====] - 587s 2s/step - loss: 2.6004 - triplet_acc: 0.2270 - val_loss: 1.4823 - val_triplet_acc: 0.2700
Beginning epoch number: 7
Current hard triplet ratio: 0.42029708456110815

```

*Nota.* La imagen ilustra la salida por consola generada en la etapa de entrenamiento del modelo de reconocimiento de la huella. Creación propia, 2022

El entrenamiento del modelo para este caso tomo una duración de **288 h 40m 54s**, la cual arrojó como resultado un intervalo de confianza del **70%**. Con la ayuda de TensorBoard se graficaron los resultados de la etapa de entrenamiento.

**Imagen 66***Gráfico de análisis de precisión según etapa*

*Nota.* La imagen ilustra el comportamiento de las dos etapas involucradas en la etapa de entrenamiento del modelo de reconocimiento de la huella. Creación propia, 2022

Tanto la etapa de entrenamiento (ResNet-like) como la etapa de validación (VGG-like) tuvieron un comportamiento similar, pero, como la etapa de validación tuvo una pérdida considerable en la parte final, esto se debe a que el intervalo de confianza obtenido no es el óptimo.

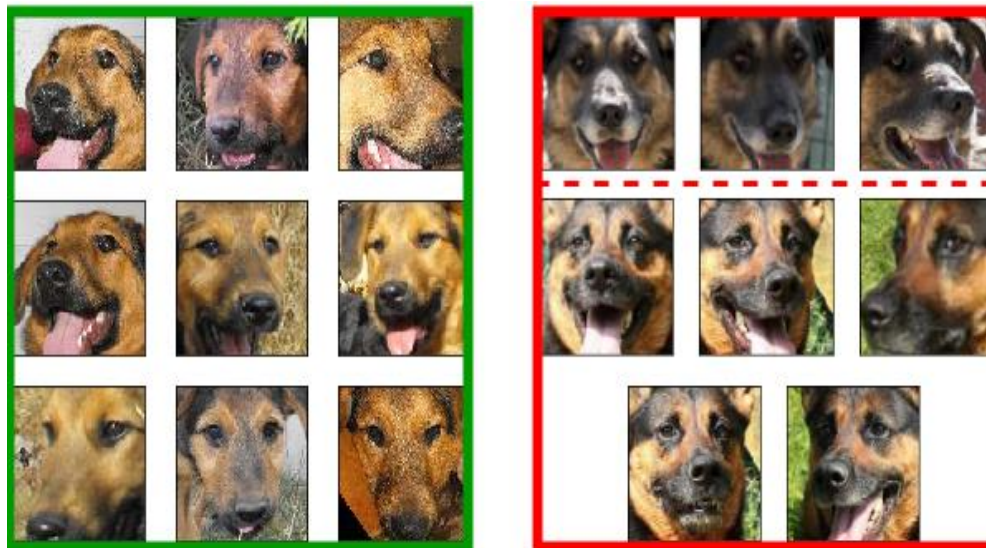
**11.1.9 Validación**

Con el set de validación definido se realiza una verificación y pruebas del modelo para determinar que es óptimo su funcionamiento. A continuación, se ilustrarán algunos ejemplos de las pruebas que se realizaron, el objetivo de la prueba consistió en conseguir que el modelo agrupara los perros según su rostro.



### *Imagen 67*

#### *Pruebas algoritmo, agrupamiento*



*Nota.* La imagen ilustra el agrupamiento realizado en las pruebas. Creación propia, 2022

Para este modelo no se obtuvieron tan buenos resultados, con una precisión o intervalo de confianza de 70% el modelo tiende a fallar en sus predicciones, lo que ocasiona imprecisión. En parte, por el reentrenamiento del algoritmo por más de 5 veces, y de estas, la más viable fue la que estaba en el intervalo de confianza mencionado anteriormente.

En la documentación referenciada no se encontró ningún modelo y/o guía para implementar este tipo de soluciones, este algoritmo también usa Deep Learning lo que aumenta la complejidad de este. Es por ello por lo que inicialmente se estableció esta versión del modelo y se espera tener una mejor noción del funcionamiento del modelo cuando se despliegue la aplicación en producción.

## 12 Conclusiones

- Se aplicó correctamente la metodología ágil, logrando una correcta organización, desarrollo del proyecto y despliegue de entregables.
- Se utilizaron herramientas para el desarrollo del proyecto como SDK, frameworks, motores de bases de datos, al igual que herramientas para el testing de servicios web como Postman y Thunder. Esto enriqueció el entorno de trabajo y optimizó de manera positiva el desarrollo del proyecto.
- Se desarrolló de manera eficiente el algoritmo de reconocimiento de la silueta con un intervalo de confianza del 99%, este es un resultado óptimo lo que garantiza el buen desempeño del mismo. modelo usado Imagenet (ILSVRC-2012-CLS) classification with MobileNet V2 (depth multiplier 1.30).
- El desarrollo del algoritmo de reconocimiento e identificación de la huella digital del perro con un intervalo de confianza del 70%, el cual es considerado que no fue óptimo, pues no garantiza el correcto funcionamiento cuando se hace uso de este, el 30% restante (errores) se presentaron por falta de iluminación, detalle, cámara inadecuada, definición de la imagen inadecuada, entre otros. El próximo objetivo con el proyecto será mejorar el desempeño de este para garantizar calidad y buen desempeño para no entorpecer la experiencia del usuario.
- Se logró integrar tecnologías a nivel de arquitectura de la solución de una manera eficiente, obteniendo un desempeño y experiencia de usuario óptimo
- Se planteó un avance inicial en Colombia en cuanto al desarrollo de aplicaciones con inteligencia artificial y se tiene un precedente para continuar el proceso de investigación.

### 13 Referencias

- Barnes, J. (2015). *Azure Machine Learning*. Microsoft Press.
- Booch , G., Rumbaugh , J., & Jacobson, I. (1998). *The Unified Modeling Language User Guide*. GiantDino.
- C.Pinilla, A. A. (1997). *Filtrado de imagenes en el dominio de la frecuencia*. Jaen: Universida de Jaen.
- Cabello Pardos, E. (2004). *Técnicas de reconocimiento facial mediante redes neuronales*. Madrid.
- Colombia, R. d. (2016). *Derechos de los animales*. Bogota: Constitucion politica de Colombia.
- Courville, A., Goodfellow, I., & Bengio, Y. (2015). *Deep Learning*. Londos, England: The MIT Press.
- Dominguez, A. T. (1996). *Procesamiento digital de imagenes*. Distrito Federal, Mexico: Perfiles educativos.
- educación, m. d. (2012). *Aplicación práctica de la visión artificial en el control de procesos industriales*. España.
- Española, R. A. (2021). *Diccionario de la lengua española*. Madrid: Real Academia Española.
- Fernandez, L. A. (1997). *Transformada de Fourier, Aplicacion al filtrado de imagenes*. Valencia españa: Universitat politecnica de valencia.
- Google. (2022, 04 09). *Google*. Retrieved from <https://research.google.com/colaboratory/intl/es/faq.html>
- Grevelink, E. (2017, 12 18). *IoT Developer Program*. Retrieved from <https://software.intel.com/content/www/us/en/develop/articles/a-closer-look-at-object-detection-recognition-and-tracking.html>
- Inmaculada Garcia, J. S. (1985). *Proceso digital de imagen en la investigacion arqueologica* . Murcia: Universidad de Murcia.
- Kaiming , H., Xiangyu, Z., Shaoqing, R., & Sun, J. (2015). *Deep Residual Learning for Image Recognition*. Microsoft Research.
- Muntané Relat, J. (2010). *INTRODUCCIÓN A LA INVESTIGACIÓN BÁSICA*. Hospital Universitario Reina Sofía.
- Republica, C. d. (2012). *Proteccion de datos*. Bogota: Constitucion politica de Colombia.
- Rouhiainen, L. (2018). 101 cosas que debes saber hoy sobre nuestro futuro. *alienta*, 22.
- Ruiz , C. A., & Basualdo, M. S. (2001). *Redes Neuronales: Conceptos Básicos y Aplicaciones*. Universidad del Rosario.
- SCRUMstudy. (2017). *CUERPO DE CONOCIMIENTO DE SCRUM ( GUIA SBOK )*. Estado Unidos: SCRUMstudy.
- servissoftcorp. (2020). *servissoftcorp*. Retrieved from <https://servissoftcorp.com/definicion-y-como-funcionan-las-aplicaciones-moviles/>
- Straube, S., & Fahle, M. (2011). *Visual detection and identification are not the same: Evidence from psychophysics and fMRI*. Brain and Cognition.
- Tabor, M., & Vrdoljak, M. (2016). *Don´t Panic GUIA A LA GALAXIA DE LAS APLICACIONES MOVILES*. Alemania: Enough Software gmbH + Co. KG.
- Team, O. (2021). *OpenCV Documentation*. Santa Clara: OpenCVTeams.
- Telefonica, a. u. (2014). *Bases de datos NoSQL. que son y tipos que nos podemos encontrar*. Desconocida: acenswhitepapers.

*Universidad de Oviedo*. (n.d.). Retrieved from <http://www6.uniovi.es/vision/intro/node2.html>  
University, S. (2020). *Stanford AI4ALL*. Retrieved from <https://ai4all.spcs.stanford.edu/>  
Vilet, J. R. (2004). *Procesamiento digital de imagenes* . San Luis Potosi: UASLP.