



**Instituto Politécnico Nacional  
Unidad Profesional Interdisciplinaria de  
Ingeniería campus Zacatecas**

**Área de ubicación para el desarrollo del  
trabajo**

Ingeniería en Sistemas Computacionales

**Línea de investigación**

Desarrollo de aplicaciones móviles

**Título del proyecto de Trabajo Terminal**

Sistema para muestreo de aves en la ciudad de  
Zacatecas

**Presenta(n):**

Axel Frederick Félix Jiménez  
Vania Stephany Sánchez Lee.

**Director:**

ISC. Efraín Arredondo Morales

**Asesores:**

M.I.S Isaul Ibarra Belmonte



Zacatecas, Zacatecas a 12 de 06 de 2024

# Índices

## Índice de contenido

Resumen.....	1
Definición del problema.....	1
Contexto y antecedentes generales del problema.....	2
Situación problemática o problema de investigación.....	3
Estado del arte.....	4
Descripción del proyecto.....	9
Objetivo general del proyecto.....	10
Objetivos particulares del proyecto.....	10
Justificación.....	10
Marco teórico.....	14
Marco Metodológico.....	26
Metodologías.....	26
Procedimientos y Actividades .....	28
Análisis y Discusión de los Resultados.....	31
Análisis de la selección de la metodología (TT-I): .....	31
Gestión del proyecto .....	34
1. Plan del proyecto.....	34
2. Manejo de desviaciones en la ejecución del plan.....	36
3. Plan de los riesgos del proyecto.....	38
Desarrollo del proyecto .....	38
1. Resumen del análisis del sistema. ....	38
2. Diseño del sistema.....	41
a. Arquitectura del sistema.....	41
b. Matriz de trazabilidad .....	44
c. Diseño de la base de datos. ....	45
d. Manejo de archivos. ....	47

e.	Diagramas de actividad.....	49
f.	Diagramas de secuencia.....	50
g.	Mockups.....	52
	Análisis de resultados.....	53
	Conclusiones y Recomendaciones .....	57
	Fuentes de consulta .....	59
	Firmas.....	61
	Autorización.....	61
	Apéndices.....	62

## Índice de tablas

Tabla I	Ventajas y desventajas del sistema de muestreos manual.....	3
Tabla II	Comparación de software para la identificación de aves.....	4
Tabla III	Características de Merlin Bird ID .....	5
Tabla IV	Características de eBird .....	6
Tabla V	Características de Birdsnap .....	8
Tabla VI	Actividades del manual de métodos de observación de aves.....	11
Tabla VII	Tabla comparativa del método manual y IdBird .....	12
Tabla VIII	Tabla comparativa del sistema propuesto con los existentes .....	13
Tabla IX	Características mínimas de teléfonos de gama media-alta.....	18
Tabla XI	Matriz de trazabilidad .....	45

## Índice de figuras

Figura 1	Muestra de Merlin Bird ID .....	6
Figura 2	Muestra de eBird.....	7
Figura 3	Muestra de Birdsnap .....	8
Figura 4	Carpintero de Arizona.....	15
Figura 5	Papamoscas Pinero .....	15
Figura 6	Carbonero Mexicano.....	16
Figura 7	Zacatonero Serrano .....	16
Figura 8	Papamoscas Cardenalito .....	17
Figura 9	Logo de TensorFlow Lite .....	19
Figura 10	Logo de Kotlin .....	20
Figura 11	Logo de SQL.....	21
Figura 12	Logo de python .....	21
Figura 13	Logo de SQLite.....	22

Figura 14 Logo de MongoDB .....	22
Figura 15 Logo de PostageApp .....	23
Figura 16 Representación gráfica del modelo V .....	24
Figura 17 Evidencia de las horas estimadas y registradas en el proyecto .....	35
Figura 18 Evidencia de las horas estimadas y registradas por etapa del proyecto .....	35
Figura 19 Plan de proyecto de TT1 completo .....	36
Figura 20 Evidencia de las desviaciones del proyecto .....	36
Figura 21 Evidencia de las desviaciones del reporte de proyecto .....	37
Figura 22 Diagrama de casos de uso .....	39
Figura 23 Diagrama de contexto .....	41
Figura 24 Diagrama de contenedor .....	42
Figura 25 Diagrama de componentes .....	43
Figura 26 Diagrama de clases .....	44
Figura 27 Diagrama de base de datos.....	46
Figura 28 Diagrama de entidad – relación .....	47
Figura 29 Diagrama de manejo de archivos .....	48
Figura 30 Diagrama de actividad de exportación de bitácoras de campo .....	50
Figura 31 Diagrama de secuencia de inicio de sesión.....	51
Figura 32 Diseño de prototipos .....	53
Figura 33 Cronograma propuesto para TT2 .....	58

## **Resumen.**

Este proyecto tiene como objetivo desarrollar una herramienta para muestrear aves en la ciudad de Zacatecas, utilizando una cámara de celular para capturar imágenes de aves y mostrar información sobre sus características al usuario. El proyecto busca mejorar el sistema de inventario de aves utilizado por los biólogos al facilitar la toma de datos y reducir el tiempo requerido para la recolección de información. Se utilizarán técnicas de procesamiento de imágenes y reconocimiento de patrones para identificar y clasificar las aves de manera rápida y efectiva, lo que permitirá un seguimiento más preciso de las aves en la región.  
La implementación del sistema se realizará a través de una aplicación móvil para dispositivos Android. Por lo que, este proyecto tiene el potencial de mejorar significativamente el sistema de inventario de aves utilizado por los biólogos en la ciudad de Zacatecas empleando diversas tecnológicas e inteligencia artificial.

**Palabras clave:** Sistema, Aves, Ciudad de Zacatecas, Cámara de celular, Procesamiento de imágenes, Reconocimiento de patrones, Mejoramiento del sistema de inventario de Aves, Implementación del Sistema, Aplicación móvil.

## **Definición del problema.**

La falta de información que permite conocer los procesos evolutivos, biogeográficos y ecológicos generados por la limitada investigación y monitoreo de las aves existentes en Zacatecas, provoca falta de conocimiento del sistema biológico enfocado a las aves de Zacatecas, vital para su estudio y preservación, así como dar seguimiento a las especies en peligro de extinción y especies endémicas [1].

Según un informe de la Comisión Nacional para el Conocimiento y Uso de la Biodiversidad (CONABIO), México enfrenta una escasez significativa de datos sobre muchas especies de aves, especialmente en regiones menos estudiadas como Zacatecas [29][30]. Esta falta de información ha sido señalada como un obstáculo importante para la conservación de la biodiversidad [31]. Además, estudios recientes indican que el monitoreo continuo y

sistemático es crucial para detectar cambios en las poblaciones de aves y responder adecuadamente a las amenazas.

La falta de un sistema de monitoreo robusto restringe nuestra capacidad para comprender los cambios en la distribución y abundancia de las aves, lo que a su vez dificulta la toma de decisiones informadas **realizadas específicamente por los biólogos encargados de la Universidad Autónoma de Zacatecas**. Esta deficiencia de datos es particularmente problemática en regiones como Zacatecas, donde muchas especies pueden estar en declive sin que haya información precisa que lo confirme. Sin un monitoreo adecuado, es difícil evaluar el estado real de las poblaciones de aves y las amenazas que enfrentan, lo que impide la implementación de estrategias de conservación efectivas.

## **Contexto y antecedentes generales del problema.**

**La problemática de la falta de un sistema eficiente para el muestreo de aves en la ciudad de Zacatecas es la más importante a considerar.** Actualmente, los biólogos y científicos que se dedican a la investigación de la fauna silvestre en la zona tienen que realizar conteos de manera manual y visual, lo que puede ser muy limitado y poco efectivo en términos de la calidad y cantidad de información recopilada. Además, la falta de un sistema automatizado de inventario de aves puede obstaculizar la toma de decisiones en relación con la conservación y protección de estas especies.

**Por otro lado, existen diversos proyectos y sistemas en todo el mundo que utilizan tecnologías avanzadas, como cámaras trampa, drones y sistemas de seguimiento GPS, para la recolección y análisis de datos de fauna silvestre.** Sin embargo, en la región de Zacatecas, la mayoría de los estudios de fauna silvestre se han realizado de manera manual y visual, lo que ha limitado significativamente la calidad y cantidad de información recopilada.

A continuación, en la Tabla I, se comparan las ventajas y desventajas del sistema de muestreos actual hecho de manera manual.

*Tabla I Ventajas y desventajas del sistema de muestreos manual*

Ventajas	Desventajas
<ul style="list-style-type: none"><li>• Permite un control detallado y específico del monitoreo y manejo directo de las aves.</li><li>• Adaptable a diversos tipos de estudios y necesidades específicas.</li><li>• Proporciona una interacción directa con las aves y el entorno, lo que puede ser importante para ciertos estudios.</li><li>• El registro manual permite anotar observaciones detalladas y contextualizadas.</li></ul>	<ul style="list-style-type: none"><li>• Mayor tiempo y probabilidad de errores humanos durante la transcripción de datos.</li><li>• Depende en gran medida de la experiencia y habilidades individuales del observador.</li><li>• Puede ser invasivo para las aves y perturbador para su entorno natural.</li><li>• La variabilidad en la ejecución puede llevar a inconsistencias en los datos.</li><li>• Menos capacidad para manejar grandes volúmenes de datos o realizar análisis complejos.</li></ul>

Sin embargo, en la región de Zacatecas, la mayoría de los estudios de fauna silvestre se han realizado de manera manual y visual, lo que ha limitado significativamente la calidad y cantidad de información recopilada. Por lo tanto, la creación de un sistema automatizado para el muestreo de aves podría llenar este vacío en la investigación de fauna silvestre en la región de Zacatecas.

## **Situación problemática o problema de investigación.**

La situación problemática que se aborda en este proyecto es la necesidad de mejorar el sistema de inventario de aves en la ciudad de Zacatecas, México. Actualmente, los biólogos y expertos en aves utilizan métodos manuales para identificar y contabilizar las especies de aves presentes en la zona, lo que puede ser muy limitado y propenso a errores. Además, esta tarea puede ser muy laboriosa y costosa en términos de tiempo y recursos [2] Por lo tanto, se requiere una solución más eficiente y precisa para la identificación y el registro de las aves en la ciudad.

Para respaldar esta problemática, se han realizado varios estudios y publicaciones sobre la importancia de mejorar los sistemas de inventario y monitoreo de aves [4] [5]. Un estudio reciente realizado en Poza Rica, Veracruz, México, resalta la necesidad de realizar inventarios de aves, especialmente en áreas donde el conocimiento de la avifauna es aún deficiente1. Este estudio también destaca cómo la urbanización rápida ha modificado los entornos donde viven las aves, lo que resulta en cambios en la composición de las aves y lleva a la necesidad de realizar inventarios [6].

Además, con el avance de la tecnología, se han desarrollado herramientas de aprendizaje automático para la identificación de aves, ayudando a los ecológos a monitorear especies raras en la naturaleza [7]. También se han desarrollado métodos automáticos de reconocimiento de especies de aves a partir de imágenes que aprovechan el aprendizaje contrastante y la mejora de características para abordar estos desafíos [8].

Estos avances tecnológicos pueden ser de gran ayuda para mejorar los sistemas de inventario y monitoreo de aves en la ciudad de Zacatecas, y en otras partes de México.

## Estado del arte.

El muestreo de aves es una actividad que se realiza para el estudio y conservación de la fauna. Sin embargo, la identificación de aves puede resultar una tarea difícil para personas que no son expertas en el tema. Con el fin de facilitar la tarea de identificación de aves, surge la idea de desarrollar un sistema para muestreo de aves utilizando el reconocimiento de imágenes mediante una cámara de celular.

En la actualidad, existen diversas aplicaciones móviles y software especializado en la identificación de aves mediante el reconocimiento de imágenes. A continuación, se presenta la Tabla II que compara las características de algunos de este software.

*Tabla II Comparación de software para la identificación de aves*

No.	Características	Merlin Bird ID	eBird	Birdsapp
1	Identificación de aves mediante imágenes	X	-	X
2	Identificación de aves mediante sonidos	X	-	-
3	Gratis	X	X	-
4	Disponible para iOS	X	-	-
5	Disponible para Android	X	-	-
6	Registro de observaciones	-	X	-
7	Generación de mapas de distribución	-	X	-
8	Información detallada sobre las aves	X	X	-
9	Uso de inteligencia artificial	X	-	X
10	Enfoque en aficionados	X	-	-
11	Enfoque en investigación científica	-	X	X
12	Plataforma en línea	-	X	-

## Desglose de Software y Proyectos:

### 1. Merlin Bird ID:

- **Descripción:** Aplicación móvil desarrollada por el Cornell Lab of Ornithology que utiliza inteligencia artificial para identificar aves a partir de fotos o grabaciones de su canto.
- **Características:** Gratuita, disponible para dispositivos iOS y Android, ofrece información detallada sobre las características de cada ave, incluyendo su distribución geográfica y los sonidos que emite [9]. Mostradas en la Tabla III.
- **Enfoque:** Identificación de aves para aficionados y observadores de aves.

Tabla III Características de Merlin Bird ID

Característica	Detalles de Merlin Bird ID
Identificación de aves mediante imágenes	Utiliza inteligencia artificial para identificar aves a partir de fotos.
Identificación de aves mediante sonidos	Reconoce aves mediante grabaciones de su canto.
Gratis	Sí
Disponible para iOS	Sí, disponible en la App Store.
Disponible para Android	Sí, disponible en Google Play.
Registro de observaciones	No
Generación de mapas de distribución	No
Información detallada sobre las aves	Proporciona características, distribución geográfica y sonidos de cada ave.
Uso de inteligencia artificial	Sí, utiliza algoritmos avanzados para el reconocimiento de imágenes y sonidos.
Enfoque en aficionados	Sí, diseñado para observadores de aves de todos los niveles.
Enfoque en investigación científica	No, no está específicamente diseñado para investigación científica.
Plataforma en línea	No, es una aplicación móvil.

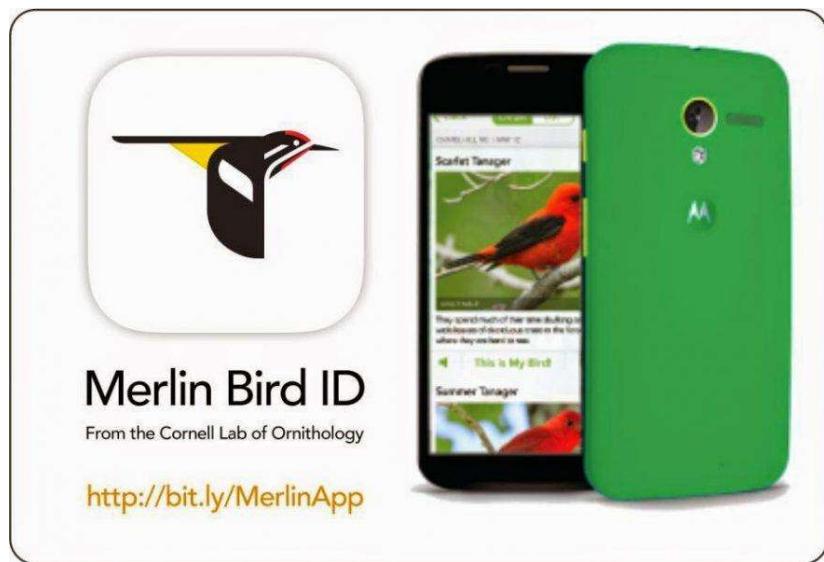


Figura 1 Muestra de Merlin Bird ID

## 2. eBird:

- **Descripción:** Plataforma en línea desarrollada por el Cornell Lab of Ornithology para la observación y registro de aves en todo el mundo.
- **Características:** Permite cargar registros de observación con ubicación, fecha, hora, especies observadas y número de aves. Genera mapas de distribución de aves en todo el mundo [10]. Mostradas en la tabla IV.
- **Enfoque:** Herramienta valiosa para la investigación científica y la conservación de aves, proporcionando datos sobre la presencia y abundancia de especies en diferentes áreas geográficas [12].

Tabla IV Características de eBird

Característica	Detalles de Merlin Bird ID
Identificación de aves mediante imágenes	No
Identificación de aves mediante sonidos	No
Gratis	Sí, es una plataforma gratuita.
Disponible para iOS	No
Disponible para Android	Sí, disponible en Google Play.

Registro de observaciones	Sí, permite a los usuarios registrar observaciones con ubicación, fecha, hora, especies y número de aves.
Generación de mapas de distribución	Sí, genera mapas de distribución de aves en todo el mundo basados en las observaciones de los usuarios.
Información detallada sobre las aves	Proporciona datos sobre la presencia y abundancia de especies en diferentes áreas geográficas.
Uso de inteligencia artificial	No especificado.
Enfoque en aficionados	No, está enfocado principalmente en la investigación y conservación.
Enfoque en investigación científica	Sí, proporciona datos valiosos para la investigación científica y la conservación de aves.
Plataforma en línea	Sí, es una plataforma en línea accesible desde cualquier navegador web.

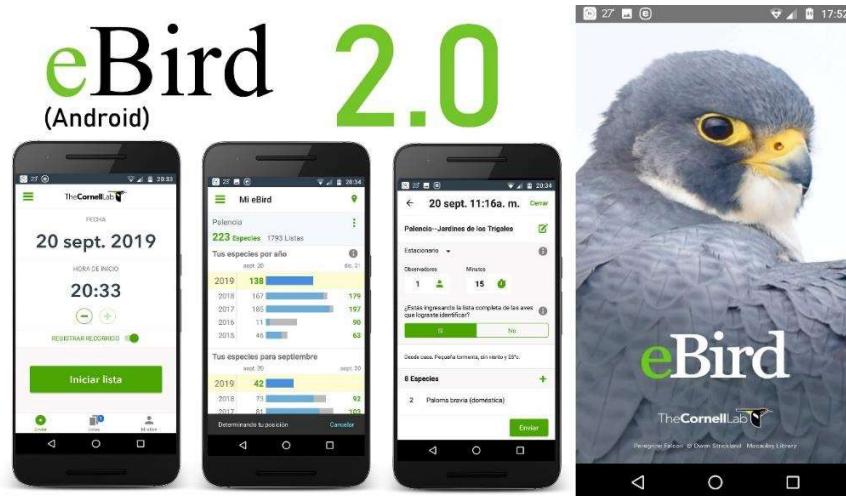


Figura 2 Muestra de eBird

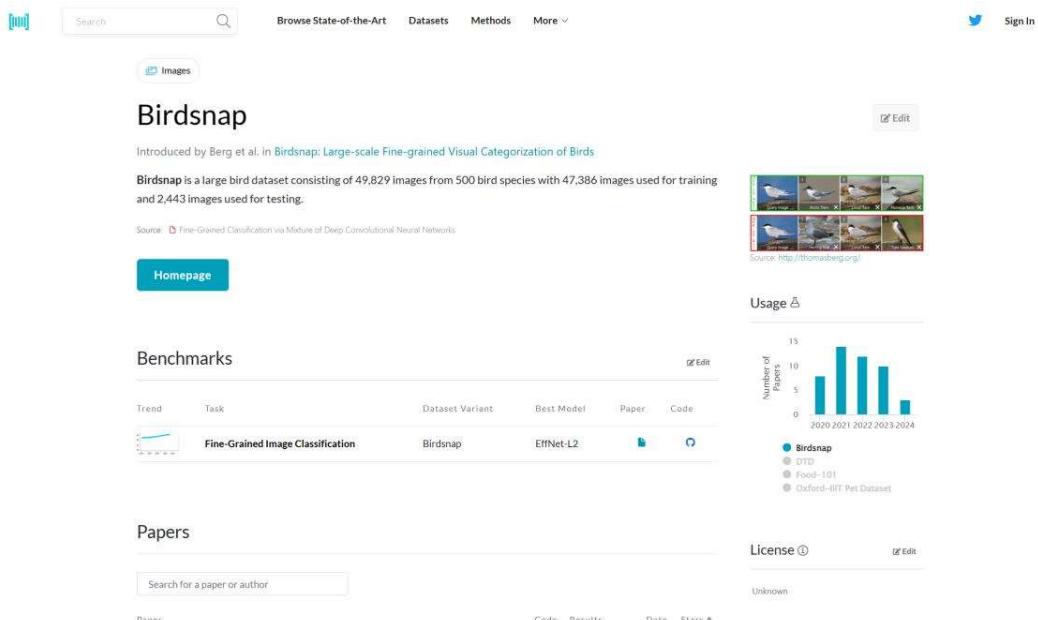
### 3. Birdsnap:

- **Descripción:** Proyecto de investigación desarrollado por la Universidad de Columbia y la Universidad de Maryland.
- **Características:** Utiliza algoritmos de aprendizaje automático para identificar aves a partir de imágenes [13]. Mostradas en la tabla V.

- **Enfoque:** Investigación en técnicas de inteligencia artificial para la identificación de aves.

*Tabla V Características de Birdsnap*

Característica	Detalles de Merlin Bird ID
Identificación de aves mediante imágenes	Utiliza algoritmos de aprendizaje automático para identificar aves a partir de imágenes.
Identificación de aves mediante sonidos	No
Gratis	No especificado.
Disponible para iOS	No especificado.
Disponible para Android	No especificado.
Registro de observaciones	No permite el registro de observaciones.
Generación de mapas de distribución	No genera mapas de distribución.
Información detallada sobre las aves	No especificado.
Uso de inteligencia artificial	Sí, utiliza algoritmos avanzados de aprendizaje automático para el reconocimiento de imágenes.
Enfoque en aficionados	No está específicamente diseñado para aficionados.
Enfoque en investigación científica	Sí, está diseñado para la investigación y el desarrollo de técnicas de inteligencia artificial en la identificación de aves.
Plataforma en línea	No especificado.



*Figura 3 Muestra de Birdsnap*

En el contexto mexicano, se han desarrollado proyectos para el estudio de la fauna, como el proyecto Biósfera Urbana, que busca documentar la biodiversidad de la ciudad de México utilizando tecnología y técnicas de ciencia ciudadana. Además, la Comisión Nacional para el Conocimiento y Uso de la Biodiversidad (CONABIO) cuenta con un catálogo de aves mexicanas, que incluye información acerca de las características de las especies y su distribución geográfica [14].

## **Descripción del proyecto.**

El proyecto que se propone es el desarrollo de un sistema para muestreo de aves en la ciudad de Zacatecas, el cual utilizará una cámara de celular para identificar las características de las aves captadas por el usuario. El objetivo principal de este proyecto es mejorar el sistema de inventario de aves utilizado por los biólogos, proporcionando una herramienta fácil y accesible para la identificación y recopilación de datos de las aves presentes en la ciudad.

El sistema propuesto funcionará mediante la utilización de técnicas de procesamiento de imágenes y aprendizaje automático. Cuando el usuario capture por medio de la cámara del celular o seleccione una imagen de un ave en particular, la aplicación utilizará algoritmos de reconocimiento de patrones para identificar la especie de ave y recopilar datos relevantes sobre su tamaño, forma, coloración, entre otros aspectos. Estos datos serán almacenados en una base de datos y estarán disponibles para su posterior análisis por parte de los biólogos.

La aplicación también contará con una interfaz de usuario donde se podrán ingresar datos del muestreo, registrar la ubicación y hora de la observación, así como tomar notas adicionales sobre el comportamiento y hábitat de las aves observadas. Además, se considera la posibilidad de integrar una funcionalidad de georreferenciación para obtener datos precisos sobre la ubicación de las aves y su relación con el entorno.

Este sistema proporcionará una herramienta valiosa para la identificación y seguimiento de las poblaciones de aves en la ciudad de Zacatecas, al permitir a los usuarios identificar y registrar las características de las aves utilizando una cámara de celular y una base de datos de referencia. Al utilizar tecnologías de reconocimiento de imágenes y aprendizaje automático, la aplicación podrá identificar la especie de ave y proporcionar información útil sobre sus características físicas y hábitat [3]. Esto permitirá una identificación más precisa y detallada de las especies de aves presentes en la ciudad, lo que puede ser muy valioso para los biólogos y otros expertos en aves que trabajan en la conservación y el estudio de la fauna local.

## **Objetivo general del proyecto.**

Llevar a cabo un muestreo no invasivo de las aves en la zona conurbada para generar un registro de las características de las aves analizadas y llevar dichos a la visualización del usuario.

## **Objetivos particulares del proyecto.**

- Identificar, registrar, analizar, clasificar a las aves muestreadas.
- Identificar conforme a la normativa de PROFEPA (Procuraduría Federal de Protección al Ambiente) si las aves muestreadas se encuentran en peligro de extinción.
- Generar una bitácora con las aves registradas.
- Localizar la posición del registro en tiempo real a través de coordenadas UTP.

## **Justificación.**

El proyecto de Sistema para muestreo de aves en la ciudad de Zacatecas busca crear un sistema de inventario de aves más eficiente y efectivo para los biólogos que se encargan de la supervisión de las poblaciones de aves en la región. Actualmente, los métodos de

recolección de datos son en su mayoría manuales, tal como se describe en el manual de métodos[55], y requieren de mucho tiempo por parte de los biólogos en cada actividad que solicita el manual según la Tabla VI, lo que limita su capacidad para monitorear de manera constante las poblaciones de aves y su comportamiento.

*Tabla VI Actividades del manual de métodos de observación de aves*

Actividad	Manual de Métodos
Preparación y Desplazamiento	1-2 horas
Instalación de Equipos (como redes de niebla)	2-3 horas
Captura y Anillamiento de Aves	4-6 horas (por día)
Recolección de Datos y Desmontaje de Equipos	1-2 horas
Procesamiento y Análisis de Datos	2-3 horas por día de muestreo
Total Aproximado por Día de Muestreo	10-16 horas
Duración del Muestreo Completo	2-4 semanas (dependiendo de la frecuencia y el área de muestreo)

La implementación de una aplicación móvil con la capacidad de identificar y clasificar las características de las aves a través del uso de cámaras de celular permitiría a los biólogos recolectar información de manera más rápida y efectiva. Además, el uso de técnicas de procesamiento de imágenes y reconocimiento de patrones ayudará a mejorar la precisión en la identificación y seguimiento de las poblaciones de aves, lo que permitirá una mejor comprensión de su comportamiento y hábitat. Se puede apreciar estas ventajas en la Tabla VII.

Este proyecto no solo tendría un impacto significativo en la investigación y conservación de las poblaciones de aves en la región de Zacatecas, sino que también podría tener aplicaciones en otras áreas de la biología y la conservación. Además, el uso de tecnología accesible como las cámaras de celular, hace que la implementación de este sistema sea accesible y de bajo costo.

*Tabla VII Tabla comparativa del método manual y IdBird*

Actividad	Manual de Métodos	IdBird
Preparación y Desplazamiento	Requiere tiempo considerable para preparación y traslado al sitio de muestreo.	Requiere tiempo mínimo para preparación y traslado al sitio de muestreo.
Instalación de Equipos	Necesita instalación de equipos como redes de niebla.	No requiere instalación de equipos adicionales.
Captura de Datos	Captura y anillamiento de aves, métodos laboriosos y manuales.	Captura de imágenes de aves mediante dispositivo móvil.
Recolección de Datos	Recolección manual de datos y desmontaje de equipos.	Recolección de datos automática al tomar las fotos.
Procesamiento de Datos	Procesamiento y análisis de datos requiere tiempo y es manual.	Procesamiento automático y en tiempo real de las imágenes.
Análisis de Datos	Ánalisis detallado y manual de datos recolectados.	Ánalisis automático y rápido mediante algoritmos de reconocimiento de patrones.
Manejo de Datos	Datos almacenados manualmente en registros físicos o bases de datos tradicionales.	Datos almacenados automáticamente en bases de datos electrónicas.
Frecuencia de Muestreo	Frecuencia de muestreo limitada por el tiempo y recursos disponibles.	Mayor frecuencia de muestreo posible debido a la eficiencia del proceso.
Personal Necesario	Requiere un equipo de personal capacitado para la instalación de equipos y recolección de datos.	Puede ser realizado por un número reducido de personas con dispositivos móviles.
Capacitación del Personal	Requiere capacitación extensa para el uso de métodos tradicionales y equipos.	Requiere capacitación mínima para el uso de la aplicación móvil.
Condiciones Meteorológicas	Las condiciones climáticas adversas pueden afectar significativamente el proceso de muestreo.	Menor impacto de las condiciones climáticas debido a la rapidez del proceso.

La implementación de una aplicación móvil con la capacidad de identificar y clasificar las características de las aves a través del uso de cámaras de celular permitiría a los biólogos recolectar información de manera más rápida. Además, el uso de técnicas de procesamiento de imágenes y reconocimiento de patrones ayudará a mejorar la precisión en la identificación y seguimiento de las poblaciones de aves, lo que permitirá una mejor comprensión de su comportamiento y hábitat. Se pueden apreciar estas ventajas en la Tabla VII.

*Tabla VIII Tabla comparativa del sistema propuesto con los existentes*

No.	Característica	IdBird	Merlin Bird ID	eBird	Birdsnap	Monitoreo Manual
1	Identificación de aves mediante imágenes	X	X	-	X	-
2	Identificación de aves mediante sonidos	-	X	-	-	-
3	Gratis	X	X	X	-	X
4	Disponible para iOS	-	X	-	-	-
5	Disponible para Android	X	X	-	-	-
6	Registro de observaciones	X	-	X	-	X
7	Generación de mapas de distribución		-	X	-	-
8	Información detallada sobre las aves	X	X	X	-	X
9	Uso de inteligencia artificial	X	X	-	X	-
10	Enfoque en aficionados	X	X	-	-	-
11	Enfoque en investigación científica	X	-	X	X	X
12	Plataforma en línea	-	-	X	-	-

Al analizar la Tabla VIII comparativa de los diferentes sistemas de identificación y monitoreo de aves, se observa que IdBird se destaca significativamente por su integración de múltiples funcionalidades clave. IdBird utiliza inteligencia artificial para identificar aves mediante imágenes, lo que facilita el proceso de identificación para los usuarios, similar a Merlin Bird ID y Birdsnap.

Una de las mayores fortalezas de IdBird es su capacidad para permitir el registro de observaciones, características que no están presentes en Merlin Bird ID y Birdsnap, pero que comparte con eBird. Estas funciones son cruciales para la investigación y conservación de aves, proporcionando datos valiosos y visualizaciones geográficas de las observaciones.

IdBird también proporciona información detallada sobre las aves y utiliza inteligencia artificial para la identificación de estas. Estas características la comparan favorablemente con otras herramientas en estas áreas. Además, IdBird es útil tanto para aficionados como para la investigación científica.

## **Marco teórico.**

### **1. Sistema:**

Un sistema es un conjunto de componentes y procesos que trabajan juntos para procesar y almacenar información. Estos componentes incluyen hardware, software y datos, cuya interacción permite que el sistema cumpla con su función específica [15].

En el presente proyecto, el sistema desarrollado es una aplicación móvil que utiliza sensores, como la cámara del celular, para registrar y procesar datos con el objetivo de llevar un conteo de las aves endémicas en la región de Zacatecas.

### **2. Aves:**

Son un grupo de vertebrados endotérmicos y ovíparos pertenecientes a la clase Aves. Se caracterizan por tener un cuerpo cubierto de plumas, un pico córneo sin dientes, un esqueleto ligero y huesos huecos, un sistema respiratorio altamente eficiente que incluye sacos aéreos y un corazón con cuatro cámaras. Además, la mayoría de las aves tienen la capacidad de volar [16]. En el contexto del proyecto "IdBird", características como color, dimensiones, y rasgos distintivos, servirán para la identificación y clasificación de las aves mediante técnicas de procesamiento de imágenes.

Las aves que se van a utilizar para su identificación en este proyecto incluyen:

- **Carpintero de Arizona (*Dryobates arizonae*)**: Es una especie de pájaro carpintero pequeño, de aproximadamente 20 cm de longitud. Tiene un plumaje principalmente negro con barras blancas en las alas y la espalda. Los machos presentan una mancha roja en la parte posterior de la cabeza [32].



Figura 4 Carpintero de Arizona

- **Papamoscas Pinero (*Empidonax affinis*)**: Este pequeño pájaro cantor mide alrededor de 13 cm de longitud. Tiene un plumaje verde oliva en el dorso y partes inferiores más claras, casi blancas. Sus alas presentan dos barras alares claras. Los ojos están rodeados por un anillo blanco tenue. Este pequeño pájaro cantor mide alrededor de 13 cm de longitud. Tiene un plumaje verde oliva en el dorso y partes inferiores más claras, casi blancas. Sus alas presentan dos barras alares claras. Los ojos están rodeados por un anillo blanco tenue [33].



Figura 5 Papamoscas Pinero

- **Carbonero Mexicano (*Poecile sclateri*):** Mide alrededor de 12 cm de longitud. Tiene un plumaje gris en el dorso y blanco en las partes inferiores. La característica distintiva de esta especie es su gorra negra que cubre la cabeza hasta el cuello, y sus mejillas blancas. Los ojos son oscuros y el pico corto y cónico [34].



Figura 6 Carbonero Mexicano

- **Zacatonero Serrano (*Oreothlypis superciliosus*):** Con una longitud de aproximadamente 12 cm, esta ave presenta un plumaje verde oliva en el dorso y amarillo en las partes inferiores. Una característica distintiva es la franja blanca sobre el ojo, que contrasta con su cara oscura [35].



Figura 7 Zacatonero Serrano

- **Papamoscas Cardenalito (*Pyrocephalus rubinus*):** Los machos de esta especie son llamativos, con un plumaje rojo brillante en la cabeza, el pecho y el vientre, y un dorso marrón oscuro. Las hembras tienen un plumaje más discreto, de color marrón grisáceo con tonos rojizos en el pecho. Mide aproximadamente 14 cm de longitud [36].



Figura 8 Papamoscas Cardenalito

### 3. Niveles de Endemismo:

El endemismo se refiere a la condición de una especie que se encuentra restringida a una ubicación geográfica específica y no se encuentra naturalmente en ningún otro lugar. Los niveles de endemismo pueden variar desde endémico (exclusivo de una región particular) hasta casi endémico (presente en una región limitada, pero también en áreas adyacentes) [37].

- **Endémico:** Una especie que es exclusiva de una región geográfica específica. Por ejemplo, el Zorzalero Serrano (*Oreothlypis superciliosus*) es una especie endémica de México, lo que significa que no se encuentra naturalmente en ningún otro lugar del mundo [38].
- **Cuasiendémico:** Una especie que se encuentra principalmente en una región, pero también puede estar presente en áreas adyacentes. En este proyecto, las siguientes aves son consideradas casi endémicas [39]:
  - Carpintero de Arizona (*Dryobates arizonae*)[40].
  - Papamoscas Pinero (*Empidonax affinis*)[41].
  - Carbonero Mexicano (*Poecile sclateri*)[42]
  - Papamoscas Cardenalito (*Pyrocephalus rubinus*)[43].

#### **4. Cámara de celular:**

Es un dispositivo que se encuentra en la mayoría de los teléfonos móviles modernos y que permite capturar imágenes y videos de alta calidad. Esta cámara utiliza una lente para enfocar la luz en un sensor de imagen, que convierte la luz en señales eléctricas que se procesan para formar una imagen digital. La calidad de la cámara del celular puede variar según el modelo, y generalmente se mide en megapíxeles. Otros factores que pueden influir en la calidad de la cámara son el tamaño de la lente, la apertura y el procesamiento de la imagen por software [17].

Para garantizar un desempeño óptimo de la aplicación "IdBird", se requiere un teléfono móvil de gama media-alta con las siguientes características mínimas:

*Tabla IX Características mínimas de teléfonos de gama media-alta*

Aspectos Diferenciadores de un Teléfono Móvil de Gama Media-Alta	
Procesador	Snapdragon serie 700, Mediatek Dimensity serie 800/900, o equivalentes. Octa-core, hasta 2.4 GHz, tecnología de 7nm o 6 nm para eficiencia y potencia.
Memoria y Almacenamiento:	4-8 GB Ram
Almacenamiento Interno:	128-256 Gb
Cámaras:	Características: Gran angular de 13 MP (f/1.8) con magnificación máxima de 0.21x, Ultra gran angular de 2 MP (f/2.2) con magnificación máxima de 0.21x, Lente de profundidad de 2 MP (f/2.4), Funciones: Enfoque automático, flash LED, geoetiquetado, HDR.

Estas características aseguran que el dispositivo tenga la capacidad de capturar y procesar las imágenes, lo cual es esencial para el funcionamiento de la aplicación de una manera optima. La cámara del celular en particular se utiliza para capturar las fotos o imágenes de las aves que serán procesadas para su identificación y clasificación dentro de la aplicación.

#### **5. Procesamiento de imágenes:**

Es el conjunto de técnicas y algoritmos que se utilizan para analizar y manipular imágenes digitales con el fin de mejorar su calidad, extraer información útil, realizar reconocimiento de patrones, entre otros objetivos. El procesamiento de imágenes se utiliza en una amplia gama de campos, como la medicina, la industria, la seguridad, la vigilancia, la ciencia, entre

otros. El procesamiento de imágenes implica el uso de herramientas y técnicas como filtrado, segmentación, detección de bordes, transformaciones geométricas, extracción de características, entre otras [18].

En "IdBird", el procesamiento de imágenes permite analizar las fotografías tomadas por los usuarios para identificar y clasificar las aves observadas.

## 6. Reconocimiento de patrones:

Es una rama de la inteligencia artificial que se enfoca en el análisis de datos y el aprendizaje automático para identificar patrones en los datos y hacer predicciones sobre nuevas entradas. Se utiliza en una amplia gama de aplicaciones, como el reconocimiento de voz, el procesamiento de imágenes, el análisis de texto, la detección de fraudes, entre otros. El reconocimiento de patrones implica el uso de técnicas y algoritmos para extraer características de los datos, construir modelos estadísticos y de aprendizaje automático, y evaluar el rendimiento de estos modelos en conjuntos de datos de prueba [19]. En "IdBird", se emplearán técnicas de reconocimiento de patrones utilizando la librería TensorFlow Lite, para implementar el modelo MobileNet V3:

- **TensorFlow Lite:** Es una librería de TensorFlow diseñada para dispositivos móviles y embebidos. Ofrece un conjunto de herramientas y APIs que permiten la ejecución eficiente de modelos de aprendizaje automático en dispositivos con recursos limitados. TensorFlow Lite es conocido por su capacidad de optimizar modelos para reducir el tamaño y mejorar el rendimiento, lo cual es esencial para aplicaciones móviles que requieren procesar datos en tiempo real sin comprometer la precisión [44].



Figura 9 Logo de TensorFlow Lite

- **MobileNet V3:** Es un modelo de red neuronal convolucional optimizado para dispositivos móviles. MobileNet V3 combina técnicas avanzadas de arquitectura de redes neuronales y eficiencia computacional, permitiendo realizar tareas de

clasificación de imágenes con alta precisión y bajo consumo de recursos. Este modelo es ideal para ser implementado en aplicaciones móviles debido a su balance entre velocidad y precisión, lo cual es crucial para el reconocimiento de patrones en "IdBird" [45].

## 7. Sistema de inventario:

Es un conjunto de herramientas y técnicas que se utilizan para gestionar y controlar el inventario de una empresa. Estos sistemas ayudan a las empresas a optimizar sus procesos de inventario, reducir los costos de almacenamiento y mejorar la eficiencia operativa. Los sistemas de inventario pueden ser manuales o automatizados, y se utilizan en una amplia gama de sectores, como la manufactura, el comercio minorista, la logística, entre otros [20]. En el proyecto "IdBird", el sistema desarrollado es una herramienta automatizada que facilita el registro y monitoreo de aves, así como facilitación de la creación de bitácoras y muestreos, haciendo mas accesible este proceso para los biólogos.

## 8. Lenguajes de programación:

Un lenguaje de programación es un conjunto de reglas y sintaxis que permite a los desarrolladores escribir programas que pueden ser ejecutados por una computadora. Los lenguajes de programación son esenciales para el desarrollo de software y aplicaciones, ya que permiten la creación de algoritmos y la gestión de datos de manera eficiente [46].

En el proyecto "IdBird", se utilizarán los siguientes lenguajes de programación:

- **Kotlin:** Kotlin es un lenguaje de programación moderno y conciso que se utiliza principalmente para el desarrollo de aplicaciones Android. Puede trabajar junto con Java y ofrece características avanzadas que mejoran la productividad del desarrollador. En "IdBird", Kotlin se utilizará para desarrollar la interfaz de usuario y la lógica de la aplicación móvil [47].



Figura 10 Logo de Kotlin

- **SQL:** SQL (Structured Query Language) es un lenguaje estándar para gestionar bases de datos relacionales. Permite la creación, modificación y consulta de datos de manera eficiente. En "IdBird", SQL se utilizará para manejar la base de datos local, que almacenará las bitácoras y los datos de muestreo. La naturaleza relacional de SQL es ideal para la organización y el acceso rápido a grandes volúmenes de datos estructurados [48].



*Figura 11 Logo de SQL*

- **Python:** Python es un lenguaje de programación de alto nivel conocido por su simplicidad y versatilidad. Se utiliza ampliamente en ciencia de datos, inteligencia artificial y desarrollo web. En "IdBird", Python se utilizará para desarrollar el sistema de reconocimiento de patrones, aprovechando librerías como TensorFlow, NumPy.. Estas librerías permitirán implementar y optimizar los modelos de clasificación de aves [49].



*Figura 12 Logo de python*

Estos lenguajes de programación se seleccionaron por su capacidad para satisfacer los requisitos técnicos y de rendimiento del proyecto.

## **9. Base de Datos:**

Una base de datos es un conjunto organizado de datos que se almacena y gestiona electrónicamente. Las bases de datos permiten almacenar grandes cantidades de información de manera estructurada y eficiente, lo que facilita la recuperación, actualización y análisis de datos [50]. En el proyecto "IdBird", se utilizarán dos tipos de bases de datos:

- **SQLite:** SQLite es una biblioteca de software que proporciona una base de datos relacional ligera, ideal para aplicaciones móviles. Permite la creación, modificación y consulta de datos de manera eficiente [51]. En "IdBird", SQLite se utilizará para manejar la base de datos local, que almacenará las bitácoras y los datos de muestreo. La naturaleza relacional de SQLite,



Figura 13 Logo de SQLite

- **MongoDB:** MongoDB es una base de datos NoSQL orientada a documentos que permite el almacenamiento de datos en un formato flexible y escalable. MongoDB es especialmente adecuado para aplicaciones que requieren alta disponibilidad y escalabilidad horizontal [52]. En "IdBird", MongoDB se utilizará para almacenar la información de los usuarios que se registren en la aplicación. Esto incluye datos como nombres de usuario (nombre y apellido), contraseñas y correos electrónicos.



Figura 14 Logo de MongoDB

## 10. API:

Una API (Interfaz de Programación de Aplicaciones) es un conjunto de definiciones y protocolos que permite a diferentes aplicaciones de software comunicarse entre sí. Las APIs son esenciales para la integración de sistemas, permitiendo el intercambio de datos y la funcionalidad entre diferentes aplicaciones de manera segura y eficiente [53].

En el proyecto "IdBird", se utilizarán dos APIs principales:

- Postage App: Postage App es un servicio de envío de correos electrónicos que ofrece una API para validar y confirmar correos electrónicos. En "IdBird", Postage App se utilizará para enviar correos electrónicos de verificación a los usuarios que se registren en la aplicación. Esto asegura que las direcciones de correo electrónico proporcionadas sean válidas y que los usuarios puedan confirmar su registro de manera segura.



Figura 15 Logo de PostageApp

- MongoDB API: Se utilizará la API de MongoDB para enviar y gestionar los datos de los usuarios, incluyendo nombre de usuario, contraseña encriptada y correo electrónico. Esta API facilitará la comunicación entre la aplicación móvil y la base de datos, asegurando que la información de los usuarios se maneje de manera eficiente y segura.

## 11. Aplicación móvil:

Es un software diseñado para ser utilizado en dispositivos móviles como teléfonos inteligentes y tabletas. Estas aplicaciones se ejecutan en el sistema operativo del dispositivo móvil y pueden ser descargadas a través de una tienda de aplicaciones. Las aplicaciones móviles se utilizan para una amplia variedad de propósitos, como entretenimiento, comunicación, educación, negocios, finanzas, entre otros [21]. La aplicación "IdBird" se desarrollará para dispositivos Android, permitiendo a los usuarios capturar imágenes de aves, procesarlas y obtener información detallada sobre las especies. Los usuarios podrán tomar fotos de las aves con sus dispositivos, y la procesará las imágenes, identificará las especies y proporcionará información sobre cada una de ellas.

## 12. Modelo V:

El modelo V es un modelo de desarrollo de sistemas diseñado para simplificar la complejidad asociada con el desarrollo en ingeniería de sistemas. Se utiliza para definir un procedimiento uniforme para el desarrollo de productos o proyectos. El modelo V es un proceso de desarrollo de software que se puede suponer que es la extensión del modelo en casaca. El modelo V demuestra las relaciones entre cada fase del ciclo de vida del desarrollo y su fase asociada de pruebas [22].

En lugar de moverse hacia debajo de forma lineal, los pasos del proceso se doblan hacia arriba después de la fase de codificación, para formar la típica forma de V, como se ilustra en la Figura 1. El propósito del modelo V es mejorar la eficiencia y calidad del desarrollo de software y reflejar la relación entre las actividades de prueba y las actividades de desarrollo. Las pruebas de software son demasiado importantes como para dejarlas para el final del proyecto, y el modelo V de pruebas incorpora las pruebas en todo el ciclo de vida del desarrollo de software [23].

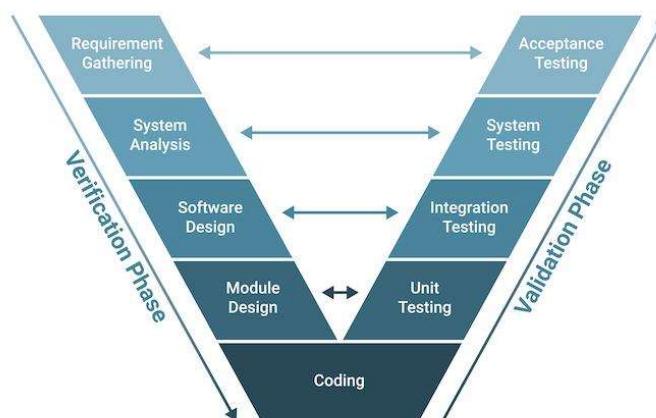


Figura 16 Representación gráfica del modelo V

## 13. Arquitectura C4

La arquitectura C4 significa contexto, contenedores, componentes y código: un conjunto de diagramas jerárquicos que puede usar para describir su arquitectura de software en diferentes niveles de zoom [24].

La arquitectura C4 se utiliza para describir la arquitectura del software en diferentes niveles de abstracción, proporcionando una visión clara y comprensible de cómo se estructura el sistema.

Este modelo evita que la documentación de la arquitectura sea compleja y confusa, así como evita que tenga poca información [27], se emplea para describir la arquitectura del software en diferentes niveles jerárquicos de abstracción:

- Nivel 1: Diagrama de contexto del sistema: Sirve para mostrar las interacciones de forma macro, sin mucho detalle, centrándose en las comunicaciones y dependencias entre los sistemas y los usuarios que componen e interactúan con el software.
- Nivel 2: Diagrama de contenedores: Sirve para describir sus contenedores y cómo se comunican/interactúan, se hace hincapié en la arquitectura y las tecnologías utilizadas.
- Nivel 3: Diagrama de componentes: Se detallan los componentes clave dentro de cada contenedor y se describen de las partes que lo componen.
- Nivel 4: Diagrama de clases o código: Explica cómo se implementa cada componente y para ello utiliza el diagrama de clases UML, se usa para una vista detallada del código fuente.

#### **14. Patrón de diseño MVC(Model-View-Controller):**

El patrón de diseño MVC (Model-View-Controller) es un patrón de diseño de software comúnmente utilizado para implementar interfaces de usuario, datos y lógica de control [25].

El patrón de diseño Modelo-Vista-Controlador (MVC) se emplea para organizar el código de la aplicación, separando la lógica de negocio, la interfaz de usuario y el control de entrada [26]. Se utiliza para organizar el código de la aplicación, usado en el diseño de la arquitectura, separando la lógica de negocio (Model), la interfaz de usuario (View) y el control de entrada (Controller). Esta separación facilita el mantenimiento del código y mejora la escalabilidad y reutilización de los componentes.

Las ventajas de utilizar MVC incluyen:

1. **Separación de Preocupaciones:** MVC separa la aplicación en tres componentes principales: el Modelo (gestión de datos y lógica de negocio), la Vista (interfaz de usuario) y el Controlador (manejo de la entrada del usuario y la lógica de flujo). Esta separación facilita el desarrollo y el mantenimiento del código, permitiendo que cada componente se desarrolle y se pruebe de manera independiente.
2. **Reutilización de Código:** La estructura MVC permite una mayor reutilización de código. Por ejemplo, las vistas pueden reutilizarse para diferentes controladores, y los modelos pueden ser utilizados por múltiples controladores. Esto reduce la duplicación de código y mejora la eficiencia del desarrollo.
3. **Facilidad de Mantenimiento y Escalabilidad:** Con MVC, es más sencillo localizar y corregir errores, ya que cada componente está claramente definido. Además, esta estructura modular facilita la adición de nuevas funcionalidades sin afectar negativamente al sistema existente.

## **Marco Metodológico.**

### **Metodologías**

La metodología seleccionada para el desarrollo de este proyecto es el modelo V. Este modelo se adapta adecuadamente a las necesidades del proyecto, ya que combina un enfoque estructurado y secuencial con la posibilidad de realizar pruebas de validación desde las primeras fases del desarrollo.

El modelo V garantiza que cada componente del software sea rigurosamente validado antes de proceder a la siguiente etapa. En cada fase del ciclo de vida del desarrollo del software, se realiza una fase correspondiente de pruebas y validación, lo que asegura que los errores se identifiquen y se corrijan de manera temprana, en lugar de acumularlos para el final del desarrollo.

Además, el modelo V facilita una mejor comprensión y seguimiento del progreso del proyecto, al proporcionar una estructura clara y bien definida. La implementación de esta modelo contribuye a asegurar que el sistema desarrollado cumple con los requisitos especificados y que cada componente funcione correctamente para integrarse en el sistema.

Adicionalmente se ha optado por una el uso de la arquitectura C4 y el patrón de diseño MVC, ya que esta combinación ofrece un enfoque integral que asegura que cada etapa del desarrollo esté validada y verificada, garantizando la calidad del producto final.

Los diagramas de la arquitectura C4 facilitan la comunicación entre los miembros del equipo y las partes interesadas, proporcionando una representación visual clara de la arquitectura del sistema y ayudando a todos a entender cómo se organiza y cómo interactúan sus componentes. Al descomponer la arquitectura del software en niveles manejables de detalle, facilita el enfoque en diferentes aspectos del sistema en cada fase del desarrollo y ayuda a gestionar la complejidad, haciendo más sencillo identificar y abordar cambios o mejoras necesarias sin afectar negativamente al sistema global.

El patrón de diseño MVC complementa esta estructura organizando el código de manera modular, mejorando la eficiencia en el desarrollo y mantenimiento del sistema, permitiendo que los cambios en una parte del sistema no afecten negativamente a otras partes.

La combinación del modelo V, la arquitectura C4 y el patrón MVC proporciona una metodología robusta que mejora la claridad, estructura, calidad y eficiencia del desarrollo del sistema, asegurando que se cumplan los requisitos del proyecto de manera efectiva y sostenible. La validación continua y la estructura clara y comprensible de la arquitectura del software garantizan un desarrollo de alta calidad y la implementación eficiente de funcionalidades como el reconocimiento de imágenes y la georreferenciación.

## **Herramientas y Técnicas:**

Para el desarrollo del proyecto, se planea utilizar Kotlin como lenguaje de programación para la creación de la aplicación móvil, y Python para desarrollar las APIs necesarias y conectar con el servicio de PostageApp y así como entrenar el modelo de reconocimiento de imágenes.

En cuanto a frameworks y librerías, se utilizará TensorFlow Lite en conjunto con el modelo de redes neuronales convolucionales (CNN) MobileNet v3 para el procesamiento de imágenes y reconocimiento de patrones.

Además, se gestionarán las tareas mediante un tablero Kanban para la organización y seguimiento durante la implementación, complementado con un cronograma para la distribución de tareas a lo largo del proyecto.

## **Procedimientos y Actividades**

### 1. Planificación del Proyecto

- Elección de Metodología:
  - Revisión de metodologías disponibles conforme al proyecto, entre ellas modelo cascada, modelo V y metodologías agiles como modelo SCRUM.
  - Lectura y selección de la metodología adecuada, tras la investigación de las diferentes metodologías se determinó que el modelo V basada en las necesidades del proyecto era el más apropiado.
- Planificación de Cronograma:
  - Creación y revisión del cronograma del proyecto conforme a la metodología elegida. Así como su validación del cronograma con el equipo y los asesores.

### 2. Análisis de Requerimientos

- Especificación de Requerimientos de Software (SRS):
  - Levantamiento de requisitos con el cliente para determinar las necesidades principales.

- Análisis y documentación de los requisitos obtenidos con el equipo de trabajo.
- Revisión y corrección del documento SRS y validación del documento final con el equipo y asesores.
- Diseño de Prototipos:
  - Creación y revisión de prototipos iniciales de la interfaz de usuario.
  - Corrección y verificación de los prototipos con el cliente para comprobar que su visión del proyecto sea plasmada en su realización.
- Matriz de Trazabilidad:
  - Creación de la matriz de trazabilidad para asegurar la relación entre requisitos, diseño y pruebas.
  - Revisión, actualización y validación de la matriz de manera continua conforme se completaban los documentos que la componen.
- Plan de Riesgos:
  - Identificación y evaluación de riesgos potenciales que se pueden presentar a lo largo del proyecto.
  - Documentación y validación de estrategias de prevención y de mitigación a aplicar en caso necesario.

### 3. Diseño del Sistema

- Diseño de la Arquitectura:
  - Creación de diagramas de contexto del sistema, contenedores, componentes y código siguiendo el modelo de modelo C4.
  - Revisión y validación de los diagramas.
- Diseño de Base de Datos y Manejo de Archivos:
  - Creación de diagramas de la base de datos y especificación del manejo de archivos.

- Revisión, corrección y verificación de los diseños.
- Diseño Detallado:
  - Creación de diagramas UML de comportamiento, para el proyecto se realizaron diagramas de casos de uso, diagramas de secuencia y diagramas de actividades.
  - Desarrollo del plan de pruebas de acuerdo a cada uno de los casos de uso.
  - Revisión y verificación de los diagramas y plan de pruebas.

Las siguientes secciones se realizarán en la segunda parte del trabajo terminal.

#### 4. Desarrollo del Sistema

- Selección de tareas:
  - Selección y distribución de tareas conforme a su dificultad y prioridad.
- Codificación:
  - Desarrollo de cada módulo de manera progresivo asegurando que cada parte funcione correctamente.
- Pruebas Unitarias e Integración:
  - Ejecución de pruebas unitarias para cada componente.
  - Realización de pruebas de integración para verificar la correcta interacción entre los componentes.
- Corrección de Errores:
  - Identificación y corrección de errores detectados durante las pruebas unitarias e integración.

#### 5. Pruebas de Sistema:

- Ejecución de pruebas completas del sistema para asegurar el cumplimiento de todos los requisitos.

## 6. Despliegue del Sistema

- Manual de Usuario, Operaciones y Mantenimiento:
  - Creación, revisión y validación de los manuales de usuario, operaciones y mantenimiento.
  - Preparación del sistema para su despliegue final.

## 7. Documentación y Presentación

- Reporte Final TT1:
  - Creación y revisión de múltiples versiones del reporte final.
  - Validación, ensayos y presentación del reporte.
- Implementación y Validación Final:
  - Despliegue final del sistema en el entorno de producción.
  - Validación final del sistema con todas las funcionalidades implementadas.

# Análisis y Discusión de los Resultados

## Análisis de la selección de la metodología (TT-I):

En esta sección se analizan las ventajas y desventajas resultantes de la implementación de la metodología elegida para la fase de documentación del "Sistema para muestreo de aves en la ciudad de Zacatecas". La metodología seleccionada combina el modelo V, el arquitectura c4 y el patrón de diseño MVC, como se puede ver en el documento de Marco Metodológico y Plan de Proyecto en el Anexo B. Aunque hasta ahora no se ha realizado ningún desarrollo de software, la documentación ha seguido estos modelos, y se evaluará cómo han influido en esta etapa del proyecto.

## **Ventajas de la Implementación de la Metodología:**

### **1. Estructura y Organización Clara:**

- Modelo V: La aplicación del modelo V en la fase de documentación ha permitido una estructuración clara y secuencial de los documentos. Cada fase de la documentación ha sido acompañada de su correspondiente fase de validación, asegurando que cada componente de la documentación cumpla con los requisitos establecidos.

### **2. Comunicación Eficaz:**

- Arquitectura c4: Los diagramas de la arquitectura c4 han sido esenciales para proporcionar una visión clara de la arquitectura propuesta del sistema. Estos diagramas han facilitado la comunicación entre los miembros del equipo y con las partes interesadas, asegurando una comprensión común y evitando malentendidos.

### **3. Claridad en la Definición de Roles y Responsabilidades:**

- Patrón de diseño MVC: Aunque el patrón MVC se aplicará principalmente en la fase de desarrollo, su consideración en la documentación ha permitido una clara definición de los componentes del sistema y sus interacciones. Esto ha facilitado la planificación y organización del trabajo futuro.

La combinación de la arquitectura y el modelo ha proporcionado una base sólida para la fase de desarrollo (TT-I). La documentación detallada y validada asegura que el equipo esté bien preparado para iniciar el desarrollo, con una comprensión clara de los requisitos y la arquitectura del sistema.

## **Desventajas de la Implementación de la Metodología:**

### **1. Complejidad en la Documentación:**

- La integración de múltiples metodologías ha añadido un nivel adicional de complejidad a la documentación. Asegurarse de que todos los miembros del equipo comprendan y sigan las metodologías elegidas ha requerido un esfuerzo considerable

en términos de planificación y coordinación, así como de investigación y capacitación respecto a cada metodología.

## **2. Requerimientos de Tiempo:**

- La necesidad de realizar una validación continua y documentar detalladamente cada fase ha incrementado el tiempo necesario para completar la documentación. Esto ha sido un desafío, especialmente en términos de cumplir con los plazos establecidos.

## **3. Curva de Aprendizaje:**

- La adopción de estos modelos ha implicado una curva de aprendizaje para los miembros del equipo. La familiarización con los conceptos y herramientas específicos de cada modelo ha requerido tiempo adicional para capacitación y adaptación.

A pesar de las desventajas mencionadas, la elección de estos modelos ha sido beneficiosa para la fase de documentación del proyecto. La combinación del modelo V, el arquitectura c4 y el patrón MVC ha proporcionado una estructura robusta y clara para la documentación, asegurando la calidad y facilitando la comunicación.

La documentación detallada y bien estructurada ha preparado al equipo para la fase de desarrollo, proporcionando una base sólida sobre la cual construir. Aunque la implementación de estas metodologías ha requerido un esfuerzo significativo, los beneficios obtenidos en términos de claridad, organización y preparación justifican plenamente la elección de estos modelos.

## **Gestión del proyecto**

### **1. Plan del proyecto.**

El cronograma presentado es el plan detallado realizado para la fase de Trabajo Terminal 1 del proyecto. Durante esta fase, se llevaron a cabo la planificación del proyecto, el análisis de requerimientos y el diseño del sistema, además de horas adicionales planificadas para la capacitación del equipo.

El plan inicial establecía un régimen de trabajo de 6 horas diarias, divididas en 3 horas por la mañana y 3 horas por la noche, durante 4 días a la semana, de lunes a jueves, además de 1 hora los viernes de cada dos semanas para reuniones. Este horario se mantendría a lo largo de 5 meses, desde febrero hasta junio, proporcionando un total máximo de 490 horas de trabajo por persona, debido a que el equipo está conformado por dos personas da un total de 980 de horas en total máximo.

Según el cronograma creado, se estimaba un total de 791 horas de trabajo, que debían ser divididas entre los dos integrantes del equipo, dependiendo de la tarea a realizar y permitiendo una distribución equitativa de la carga de trabajo. Fueron distribuidas entre las siguientes actividades:

- **Plan de Proyecto:** 57 horas
- **Análisis de Requerimientos:** 216 horas
- **Diseño del Sistema:** 348 horas
- **Capacitación del Equipo:** 60 horas
- **Reporte Final TT1:** 110 horas

Sin embargo, al finalizar el semestre de TT1 el equipo completó un total de 422 horas de trabajo, no obstante al ser un proyecto de dos individuos el esfuerzo por persona duplica las horas de trabajo dandonos un total de 844 horas , que distribuidas de la siguiente manera.

- **Plan de Proyecto:** 47 horas
- **Análisis de Requerimientos:** 88 horas

- **Diseño del Sistema:** 175 horas
- **Reporte Final TT1:** 82 horas
- **Investigación y Capacitación:** 30 horas

La Figura 2 demuestra las horas estimadas al inicio del proyecto y su comparación con las horas registradas al finalizar la parte del proyecto trabajada en trabajo terminal I.



Figura 17 Evidencia de las horas estimadas y registradas en el proyecto



Figura 18 Evidencia de las horas estimadas y registradas por etapa del proyecto

La diferencia de horas de 791 a 422 se realizó principalmente a que el equipo de trabajo realizaba diferentes tareas de manera simultaneas, Este total fue significativamente menor que las 791 horas estimadas inicialmente, lo que permitió ahorrar casi la mitad del tiempo planeado. Esta estrategia de trabajo en paralelo fue importante para optimizar el uso del tiempo y asegurar que las actividades se completaran dentro del plazo establecido.

Como se puede observar en la Figura 3, se muestran las actividades realizadas para cada una de las etapas de plan de proyecto, análisis de requerimientos, diseño del sistema y reporte final TT1, así como las horas estimadas y realizadas de cada una.

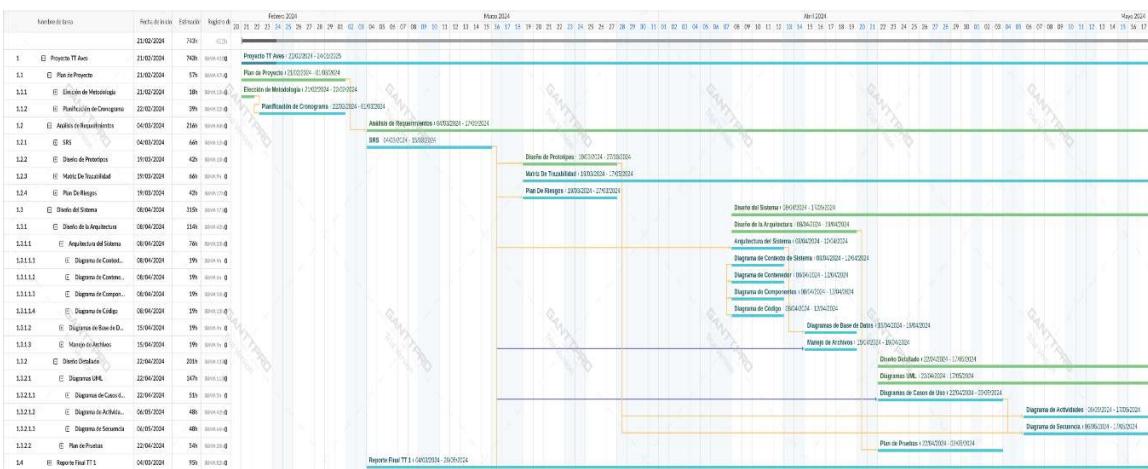


Figura 19 Plan de proyecto de TT1 completo

## 2. Manejo de desviaciones en la ejecución del plan.

Durante la realización del proyecto se encontraron retrasos y desviaciones del plan de proyecto que se tenía establecido inicialmente. Estas desviaciones se presentaron principalmente en la parte de diseño, ya que se tenía contemplado una semana para los diagramas de arquitectura, una semana para los diagramas de base de datos y manejo de archivos, dos semanas para el diagrama de casos de uso y el plan de pruebas y dos semanas extras para los diagramas de actividades y de secuencia, ilustrado por la Figura 4 donde se aprecian la cantidad de horas estimadas y las horas realizadas, así como las semanas en las que se desempeñaron las actividades.



Figura 20 Evidencia de las desviaciones del proyecto

Sin embargo, en la realización de cada uno de los diagramas se percibió que los tiempos previamente contemplados eran erróneos en la mayoría de los diagramas. Puesto que para la arquitectura del sistema fue más extensa de lo esperado y en total tardó dos semanas en completarse, lo que llevó a un desplazamiento de una semana a los demás diagramas. Sin embargo, se pudo recuperar esta semana de atraso al realizar los diagramas de base de datos, manejo de archivos y casos de uso de manera rápida y eficaz.

Otra desviación que se tuvo fue con los diagramas UML, específicamente los diagramas de actividad y secuencia, que en vez de llevarse dos semanas simultáneas se realizaron en semana y media cada uno iniciando con los diagramas de actividad para posteriormente elaborar los diagramas de secuencia. Al desconocer los pasos y los formatos necesarios para un diagrama de secuencia pues no se contaba con la experiencia necesaria, fue necesario emplear más tiempo para la capacitación y los intentos fallidos, hasta conseguir una versión de calidad para describir los procesos del sistema.

Por último, se elaboró el plan de pruebas al finalizar todos los diagramas de diseño, finalizando todo el diseño del sistema para la semana del 19 al 25 de mayo y dar comienzo al reporte final de trabajo terminal 1.

Por otro lado, para la realización del reporte final se había contemplado realizar avances conforme a la finalización de cada bloque de documentos, sin embargo, por cuestión de tiempo y múltiples pruebas y validaciones no se fueron agregando al tiempo y se prefirió agregar al reporte cuando se hubieran completados todos los documentos y se haya validado de nueva cuenta los documentos producidos en conjunto para garantizar la calidad y alineación a los objetivos del proyecto. Esta estimación se ilustra en la Figura 5, donde las etapas de creación y sus versiones se omitieron al esperar a completar los documentos previos y agregarlos al reporte de manera total en la etapa considerada para corrección.

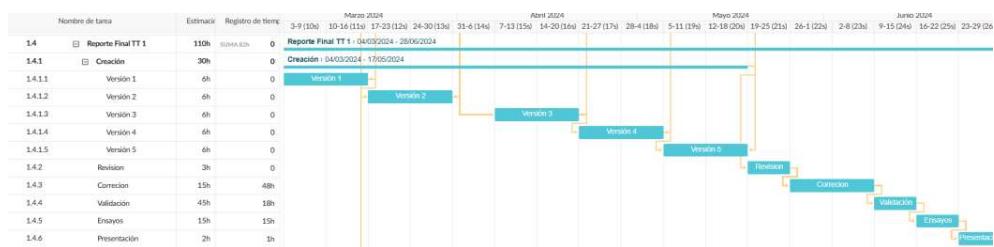


Figura 21 Evidencia de las desviaciones del reporte de proyecto

### **3. Plan de los riesgos del proyecto.**

El manejo de riesgos del proyecto se centró en identificar, evaluar y mitigar los posibles eventos que pudieran afectar negativamente el desarrollo del proyecto. A lo largo del ciclo de vida del proyecto, se implementaron diversas estrategias para gestionar eficazmente los riesgos, con especial atención a aquellos que se detonaron y requirieron acciones específicas para su mitigación, como se pueden ver en el documento de Plan de Riesgos en el Anexo D.

Uno de los principales riesgos que ocurrieron durante el proyecto fue la falta de asistencia en ciertos días de trabajo por parte de algunos miembros del equipo. Este riesgo tenía el potencial de retrasar el progreso del proyecto debido a la ausencia del equipo de trabajo. Para mitigar este riesgo cuando se presentó, se replanificaron las actividades considerando el tiempo disponible y la cantidad de tareas pendientes. Esto implicó una redistribución de las responsabilidades entre los miembros presentes, asegurando que las actividades críticas continuaran sin interrupciones.

Otro riesgo que se presentó fue el conflicto de puntos de vista del proyecto en el equipo de trabajo sobre detalles del diseño y técnicas para realizarlos. Para resolver estos conflictos, se implementó un consenso que involucraba las opiniones del director, asesor y cliente, asegurando que todas las partes interesadas tuvieran la oportunidad de expresar sus preocupaciones y sugerencias. Este enfoque colaborativo facilitó la llegada a un acuerdo común, alineando los objetivos del equipo y mejorando la cohesión y la cooperación entre los miembros.

## **Desarrollo del proyecto**

### **1. Resumen del análisis del sistema.**

En esta sección se presentará un resumen del análisis realizado para el "Sistema para muestreo de aves en la ciudad de Zacatecas". Se incluyen la representación gráfica de los requerimientos de usuario y una síntesis de los requerimientos establecidos en la Especificación de Requerimientos de Software (SRS) mostrado en el Anexo C.

### Representación Gráfica de los Requerimientos de Usuario (Diagrama Conceptual):

El diagrama conceptual es una representación visual de los principales requerimientos del usuario y cómo estos se interrelacionan dentro del sistema. A continuación en la Figura 6 se presenta el diagrama conceptual del sistema:

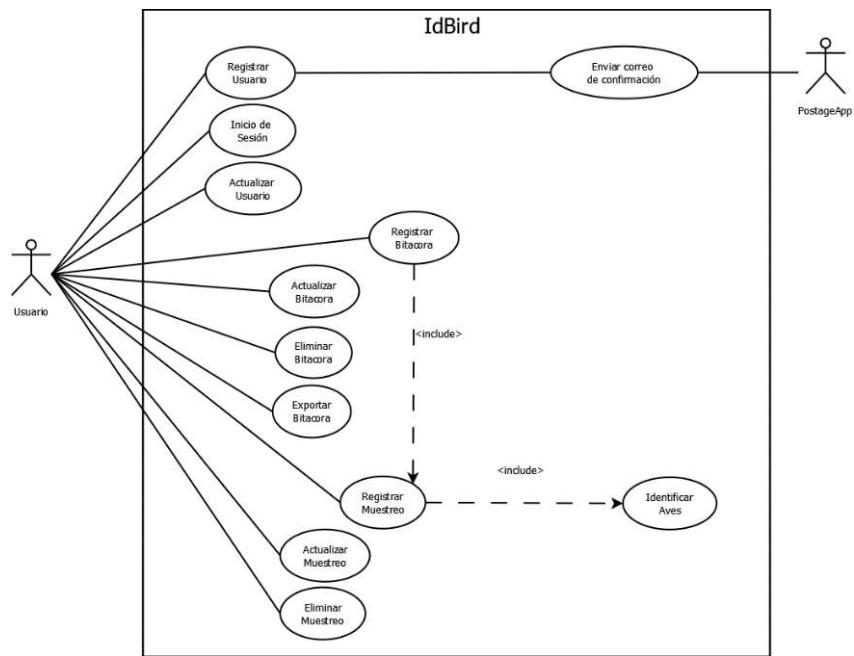


Figura 22 Diagrama de casos de uso

El diagrama describe como el usuario interactúa con el sistema "IdBird" a través de múltiples funcionalidades, tales como registrar usuarios, iniciar sesión, actualizar usuarios, registrar, actualizar y eliminar bitácoras, así como exportar bitácoras. También incluye la funcionalidad de registrar, actualizar y eliminar muestreos, y la identificación de aves. Ademas que PostageApp es el sistema encargado de enviar el correo de confirmación al usuario.

### Los Requerimientos Establecidos (SRS):

El documento de Especificación de Requerimientos de Software (SRS) detalla todos los requerimientos funcionales y no funcionales necesarios para el desarrollo del sistema. A continuación en la Tabla III se presenta un resumen de los principales requerimientos establecidos:

*Tabla Tabla de requerimientos*

ID	Nombre del requerimiento	Tipo de requerimiento
RF-01	Registro, Inicio y Actualización de Usuarios	Funcional
RF-02	Registro, Actualización y Eliminación de Bitácoras de Campo	Funcional
RF-03	Registro, Actualización y Eliminación de Muestreos.	Funcional
RF-04	Identificación de Aves	Funcional
RF-05	Exportación de Bitácoras de Campo	Funcional
RNFD-01	Preprocesamiento de Imágenes	No funcional de desempeño
RNFD-02	Localización en tiempo real	No funcional de desempeño

### **Requerimientos Funcionales:**

- RF-01: Registro y Actualización de Usuarios:  
Permite a los usuarios registrarse y actualizar su información personal.
- RF-02: Registro, Actualización y Eliminación de Bitácoras de Campo:  
Permite a los usuarios crear, actualizar y eliminar bitácoras de observación de aves.
- RF-03: Registro, Actualización y Eliminación de Muestreos:  
Permite a los usuarios registrar, actualizar y eliminar datos de muestreo de aves.
- RF-04: Identificación de Aves:  
Permite identificar aves a partir de imágenes capturadas con la cámara del celular.
- RF-05: Exportación de Bitácoras de Campo:  
Permite exportar las bitácoras de observación de aves para su análisis o archivo.

### **Requerimientos No Funcionales**

- RNFD-01: Preprocesamiento de Imágenes;  
Implica el uso de técnicas de procesamiento de imágenes para mejorar la calidad y precisión de la identificación de aves.
- RNFD-02: Localización en Tiempo Real:  
Permite registrar la ubicación geográfica de las observaciones de aves en tiempo real.

## 2. Diseño del sistema.

### a. Arquitectura del sistema.

Para la representación gráfica de la arquitectura del sistema "IdBird" se ha utilizado el arquitectura c4, que descompone la arquitectura en cuatro niveles de abstracción: Contexto, Contenedores, Componentes y Clases. A continuación, se describen cada una de las imágenes proporcionadas para cada nivel.

#### Diagrama de Contexto:

El diagrama de contexto muestra la interacción del usuario con la aplicación móvil "IdBird". En esta vista de alto nivel, el usuario es la principal entidad interactuante que utiliza la aplicación móvil para el muestreo e identificación de aves en la ciudad de Zacatecas. La aplicación móvil "IdBird" es el sistema principal que proporciona estas funcionalidades al usuario, ilustrado en la Figura 7. En este nivel, se destaca la simplicidad y el enfoque en la interacción principal del usuario con el sistema.

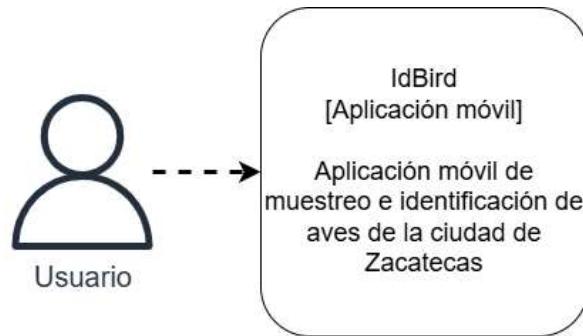


Figura 23 Diagrama de contexto

#### Diagrama de Contenedor:

El diagrama de contenedores descompone el sistema en sus principales contenedores de software, mostrando cómo interactúan entre sí. En esta vista, se identifica la aplicación móvil desarrollada en Kotlin y utilizando el patrón de diseño MVC. Como se puede observar en la Figura 8, la aplicación móvil se comunica con una API de correos desarrollada en Node.js y Express, que a su vez se conecta con el sistema de envío de correos PostageApp. La API de

correo maneja la verificación del registro de usuarios a través del envío de correos electrónicos. La base de datos MongoDB se utiliza para almacenar las credenciales de los usuarios y confirmar su registro. Esta vista destaca la estructura del sistema y la interacción entre sus principales contenedores.

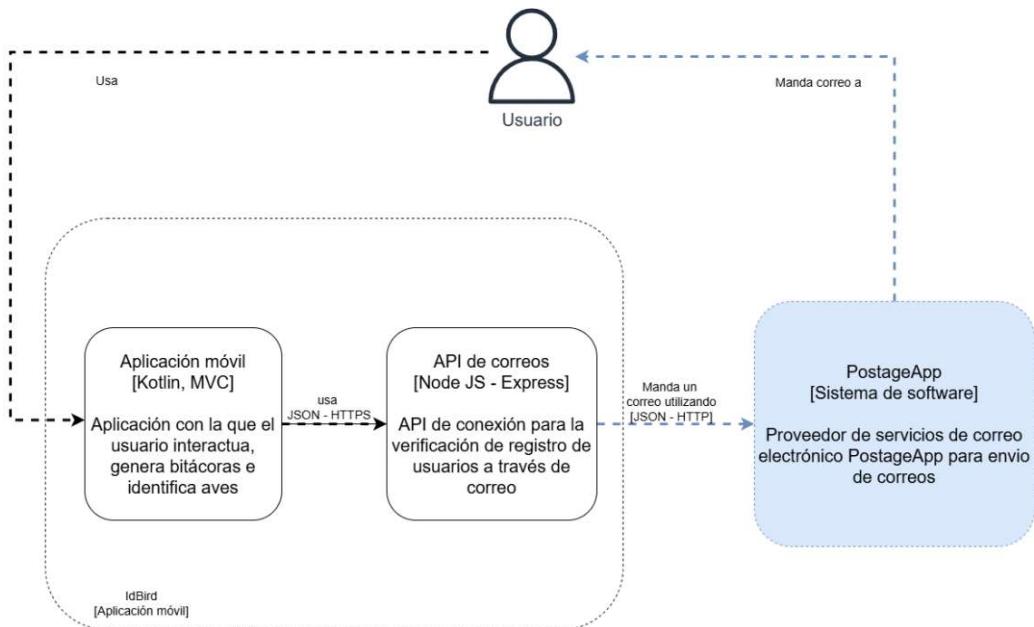


Figura 24 Diagrama de contenedores

### Diagrama de Componentes:

El diagrama de componentes proporciona una visión detallada de los componentes internos de cada contenedor identificado en la vista de contenedores. La Figura 9 detalla cada uno de los componentes de manera visual para una mejor compresión. Para la aplicación móvil, se detallan los componentes MVC: Modelos, Vistas y Controladores. Cada uno de estos componentes maneja aspectos específicos de la aplicación, como la gestión de usuarios, bitácoras y muestreros, así como la identificación de aves. La API de correos está dividida en componentes que manejan la autenticación y el envío de correos, utilizando la base de datos para gestionar la información del usuario. Esta vista permite comprender cómo se organiza el software internamente y cómo los componentes interactúan entre sí para proporcionar las funcionalidades del sistema.

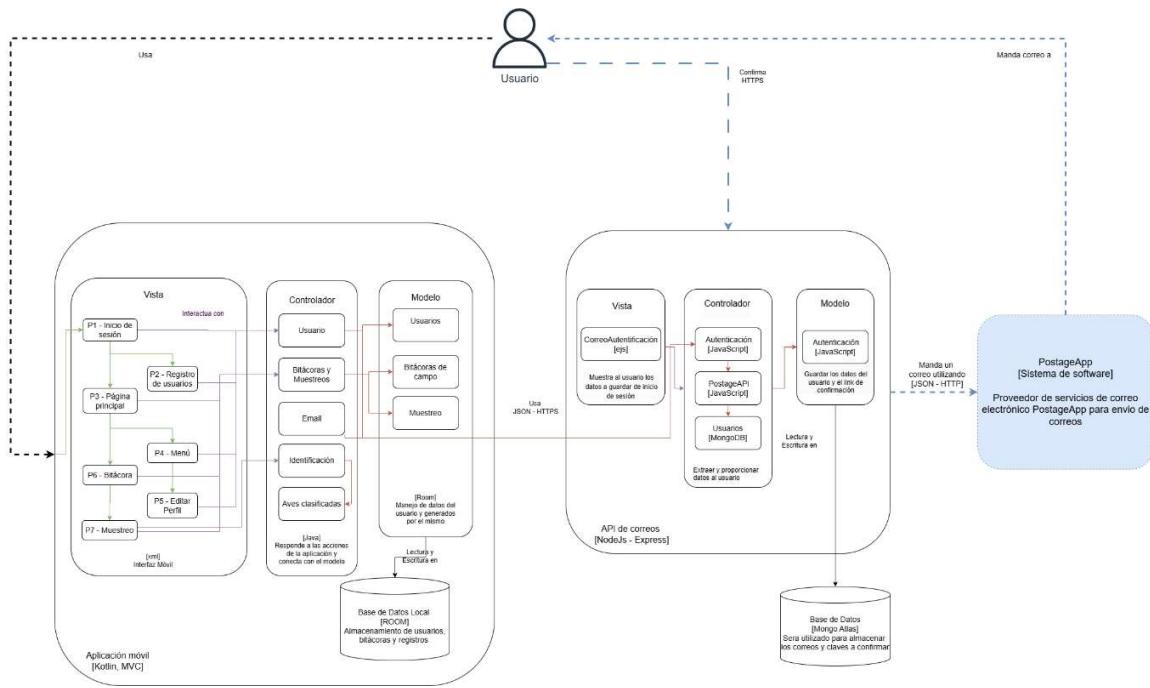


Figura 25 Diagrama de componentes

### Diagrama de Clases:

El diagrama de clases detalla a nivel de código, mostrando las principales clases dentro de cada componente del sistema. La Figura 10 muestra cada una de las clases a usar en el proyecto y detalla sus funcionalidades necesarias. En la aplicación móvil, se destacan clases como Usuario, BitácorasMuestreos, Identificación, y AvesClasificadas. Cada clase tiene atributos y métodos específicos que permiten realizar operaciones como registrar usuarios, gestionar bitácoras y muestras, y clasificar aves a partir de imágenes. En la API de correos, se detallan clases como Autenticación y PostageAPI, que manejan el registro de usuarios y el envío de correos de confirmación. Esta vista proporciona una visión detallada de la implementación y estructura del código, facilitando la comprensión del funcionamiento interno del sistema.

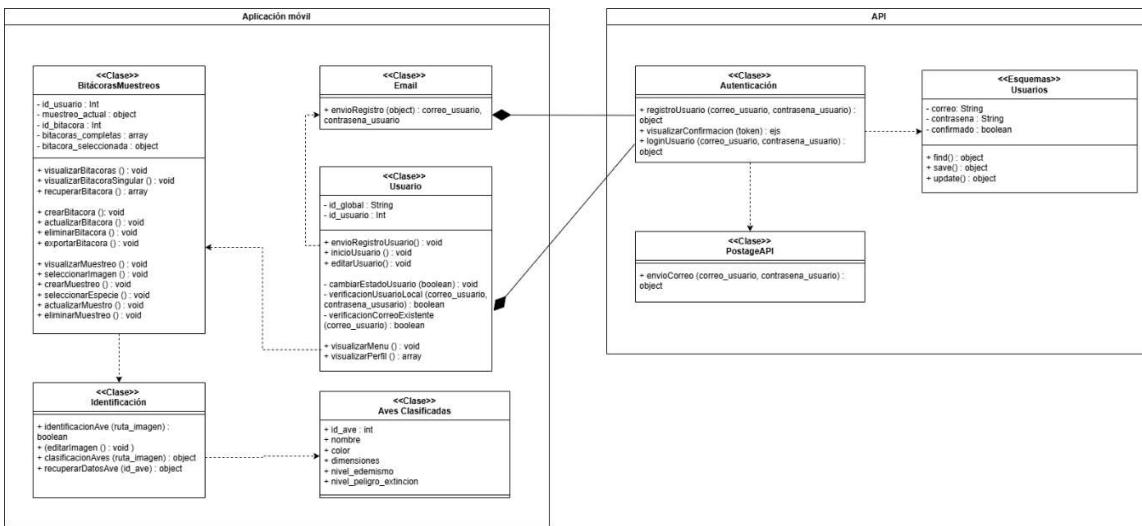


Figura 26 Diagrama de clases

Estos diagramas del arquitectura c4 proporcionan una representación clara y estructurada de la arquitectura del sistema "IdBird", desde la interacción a nivel de usuario hasta la implementación detallada del código. Cada nivel de abstracción ofrece información específica que ayuda a entender cómo se organiza y funciona el sistema en su totalidad.

## b. Matriz de trazabilidad

La matriz de trazabilidad del proyecto es una herramienta esencial que permite rastrear y relacionar los diferentes elementos del proyecto, asegurando que todos los requerimientos están cubiertos por los componentes del sistema y que todas las funcionalidades están debidamente probadas. Este documento establece una conexión clara entre los objetivos del proyecto, los requerimientos, los diagramas de diseño, los componentes, los casos de uso y las pruebas realizadas.

El objetivo principal de la matriz de trazabilidad es garantizar que el sistema desarrollado cumple con todos los requerimientos definidos, desde la fase de diseño hasta la implementación y prueba. La Tabla IV presenta la matriz de trazabilidad, que documenta el seguimiento de los requisitos a lo largo del proyecto.

Tabla X Matriz de trazabilidad

Objetivo particular	Requerimiento	Diseño	Componente	Casos de uso	Prueba	
Registro y Autenticación de Usuarios	RF-01	P1	Usuario Email Autenticación PostageAPI Usuarios	CU-01	EP-01	
		P2			EP-02	
		PE3		CU-02	EP-03	
Gestión del Perfil de Usuario	RF-01	P6	Usuario	CU-03	EP-04	
		P7				
Creación y Gestión de Bitácoras de Campo	RF-02	P3	Bitácoras y Muestreos	CU-04	EP-05	
		P4		CU-05	EP-06	
				CU-06	EP-07	
		P4		CU-11	EP-12	
Manejo de Información de Muestreos	RF-03		Bitácoras y Muestreos	CU-07	EP-08	
				CU-08	EP-09	
	PE1	CU-09		EP-10		
Operaciones con los Muestreos	RF-03	P4	Bitácoras y Muestreos	CU-07	EP-08	
		P5		CU-08	EP-09	
				CU-09	EP-10	
Identificación y Clasificación de las Aves Muestreadas	RF-04	P5	Identificación	CU-10	EP-11	
		PE3				
Cumplimiento de Normativas Ambientales	RF-05	PE1	Bitácoras y Muestreos	CU-10	EP-11	
				CU-11	EP-12	
Cumplimiento de Normativas Ambientales	RF-02	P4	Bitácoras y Muestreos	CU-04	EP-05	
	RF-03	P5		CU-07	EP-08	

### c. Diseño de la base de datos.

El diseño de la base de datos para el sistema "IdBird" se ha basado en un modelo entidad-relación (ER) y un modelo de base de datos que proporcionan una estructura clara para almacenar y gestionar la información de los usuarios, las bitácoras y los muestreos de aves.

#### Tablas de la Base de Datos:

El modelo físico de la base de datos se implementa en tablas que reflejan las entidades y relaciones del modelo ER.

- **Usuarios:** Esta tabla almacena la información personal de los usuarios, como nombre, apellido, correo y contraseña. Cada usuario tiene un identificador único (id\_Usuarios), que se utiliza para asociar sus bitácoras.

- Bitácoras: La tabla de bitácoras almacena detalles de las observaciones realizadas por los usuarios, incluyendo la fecha, hora, lugar y ubicación de cada bitácora. Cada bitácora está vinculada a un usuario específico a través de Usuarios\_id\_Usuarios, garantizando que solo los usuarios propietarios puedan acceder y gestionar sus propias bitácoras.
- Muestreos: Esta tabla almacena información detallada de cada muestreo dentro de una bitácora, como el nombre del muestreo, el nombre coloquial del ave observada, coordenadas, color y dimensiones. Cada muestreo está asociado a una bitácora específica mediante Bitacoras\_id\_Bitacoras, asegurando una relación clara y estructurada entre los datos del muestreo y la bitácora correspondiente.

Como se observa en la Figura 11, se presentan las tablas descritas con sus respectivos atributos, además de sus relaciones entre cada una de ellas.

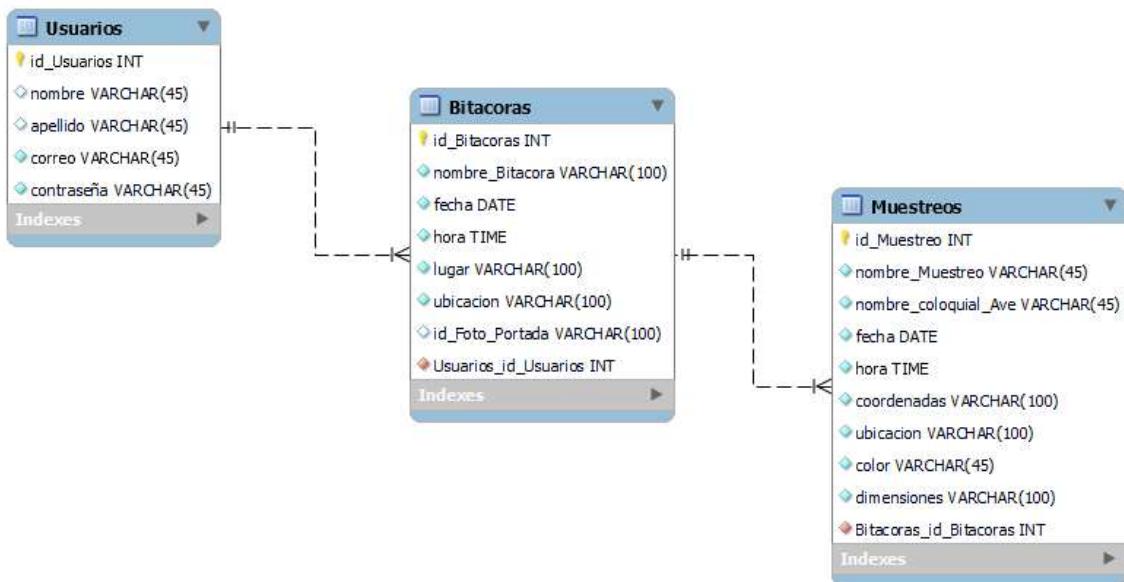


Figura 27 Diagrama de base de datos

## Modelo Entidad-Relación (ER):

El modelo ER, ilustrado en la Figura 12, define las principales entidades del sistema: "Usuarios", "Bitácoras" y "Muestreos", y establece las relaciones entre ellas. Cada usuario puede tener múltiples bitácoras, y cada bitácora puede contener múltiples muestreos. Esto asegura que la información esté organizada de manera jerárquica y lógica, facilitando la gestión y consulta de datos relacionados. La relación uno a muchos entre "Usuarios" y "Bitácoras" y entre "Bitácoras" y "Muestreos" permite mantener la integridad referencial de los datos, asegurando que cada bitácora esté asociada a un usuario y cada muestreo a una bitácora específica (imagen entidad-relación).

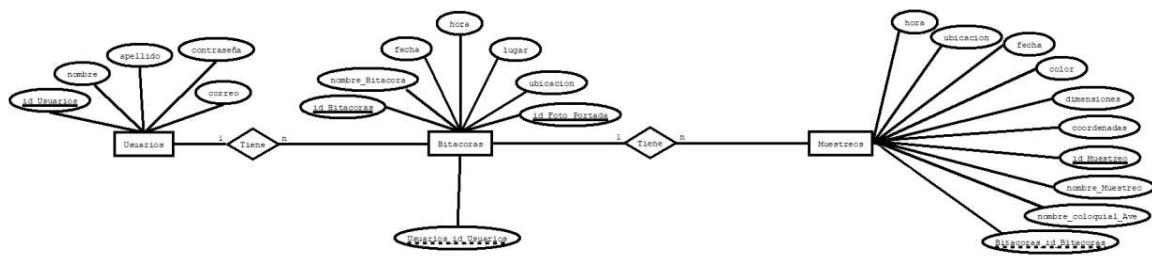


Figura 28 Diagrama de entidad – relación

## d. Manejo de archivos.

En el desarrollo del proyecto "IdBird", se ha implementado un sistema de manejo de archivos para gestionar las bitácoras de campo, los muestreos y los archivos PDF generados por la aplicación. A continuación, se describe la estructura del contenido de los archivos y la ruta relativa de su ubicación dentro del sistema.

### Estructura del Contenido de los Archivos:

La estructura de manejo de archivos está organizada en directorios y subdirectorios para facilitar el acceso y la gestión de los datos generados y utilizados por los usuarios. La estructura se muestra siguiente Figura:

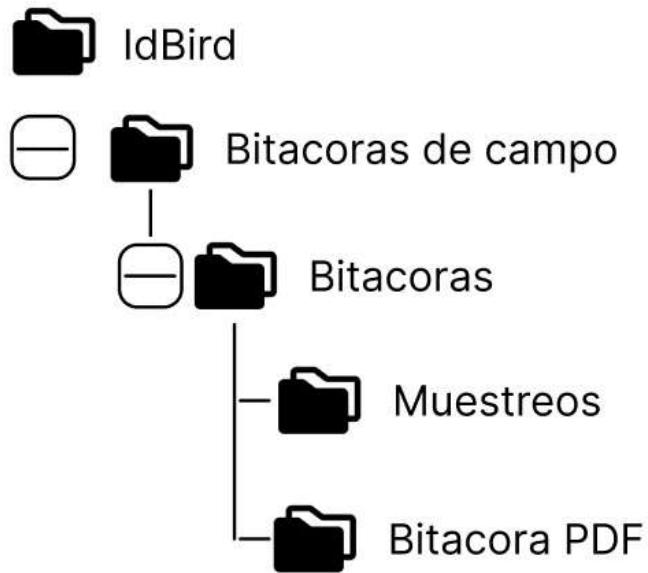


Figura 29 Diagrama de manejo de archivos

- **IdBird**: Directorio raíz de la aplicación que contiene todos los archivos relacionados con las bitácoras de campo.
- **Bitácoras de campo**: Directorio principal que agrupa todas las bitácoras creadas por los usuarios.
- **Bitácoras**: Subdirectorio que almacena todas las bitácoras individuales.
- **Muestreos**: Subdirectorio dentro de cada bitácora que almacena las imágenes subidas o tomadas por el usuario para el muestreo específico.
- **Bitácora PDF**: Subdirectorio dentro de cada bitácora que almacena los archivos PDF generados por la aplicación para cada bitácora.

#### Ruta Relativa de la Ubicación de los Archivos:

Las rutas relativas dentro del sistema de archivos de la aplicación son las siguientes:

##### 1. Bitácoras de Campo:

- Ruta: ./IdBird/Bitácoras de Campo/Bitácoras/
- Este directorio contiene todas las bitácoras creadas por los usuarios.

##### 2. Muestreos:

- Ruta: ./IdBird/Bitácoras de Campo/Bitácoras\_N/Muestreos/

- Este subdirectorio almacena las imágenes relacionadas con los muestreos específicos realizados dentro de cada bitácora.

### **3. Bitácora PDF:**

- Ruta: ./IdBird/Bitacoras de Campo/Bitacoras\_N/Bitacora PDF/
- Este subdirectorio almacena los archivos PDF generados por la aplicación para cada bitácora, proporcionando una versión imprimible y portátil de las bitácoras de campo.

La organización de los archivos en esta estructura jerárquica permite un acceso eficiente y una gestión clara de los datos generados por los usuarios, asegurando que la información esté adecuadamente almacenada y fácilmente recuperable cuando sea necesario.

#### **e. Diagramas de actividad.**

Los diagramas de actividad proporcionan una visualización detallada de los flujos de trabajo y las interacciones del usuario con el sistema. Estos diagramas son esenciales para entender cómo se llevarán a cabo las principales funcionalidades del sistema, desde el registro de usuarios hasta la exportación de bitácoras en formato PDF, puesto que proporcionan una comprensión clara y estructurada de las interacciones del usuario con el sistema, asegurando que todos los procesos críticos estén bien definidos y gestionados de manera eficiente.

Al igual que todos los diagramas de diseño, se encuentran los 5 diagramas en el documento, se muestra uno de los diagramas para ejemplificar.

##### Exportación de Bitácoras de Campo (RF-05)

El diagrama para la exportación de bitácoras, ilustrado en la Figura 14, muestra cómo los usuarios pueden seleccionar una bitácora y exportarla en formato PDF. Este flujo incluye la obtención de la información de la bitácora, la conversión a un formato de reporte y la generación del archivo PDF. La actividad finaliza con la visualización del reporte generado.

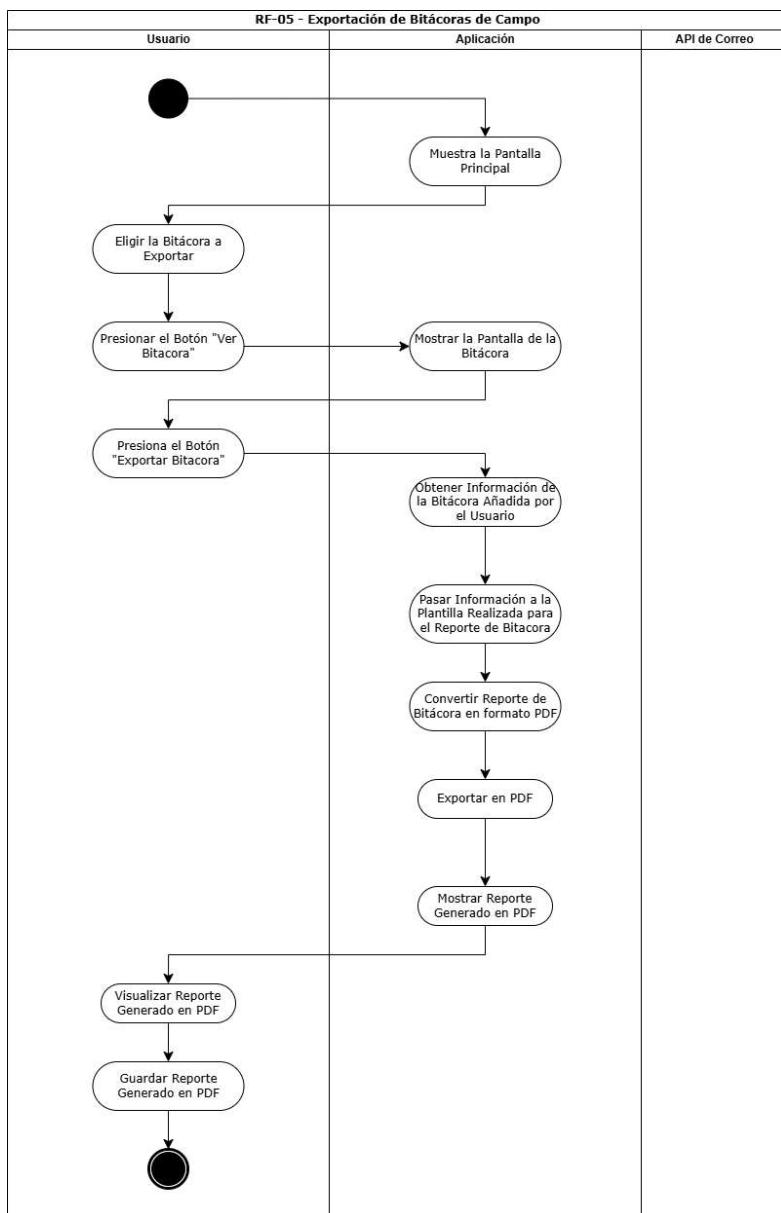


Figura 30 Diagrama de actividad de exportación de bitácoras de campo

## f. Diagramas de secuencia.

Los diagramas de secuencia del proyecto proporcionan una representación detallada de las interacciones entre los diferentes componentes del sistema y los usuarios, describiendo el flujo de mensajes y las actividades involucradas en la realización de tareas específicas. Estos diagramas son esenciales para entender la dinámica del sistema y asegurar que todas las funcionalidades se implementen correctamente.

Los diagramas de secuencia cubren varios escenarios clave del sistema "IdBird", incluyendo el registro e inicio de sesión de usuarios, la actualización de perfiles, la creación y gestión de bitácoras y muestreos, la identificación de aves, y la exportación de bitácoras en formato PDF.

Un ejemplo de estos es el ilustrado en la Figura 15, el diagrama de inicio de sesión detalla el proceso completo desde que el usuario ingresa sus credenciales hasta que el sistema verifica la información y concede acceso. Incluyen la interacción con la API de autenticación y la actualización de los estados de usuario en la base de datos.

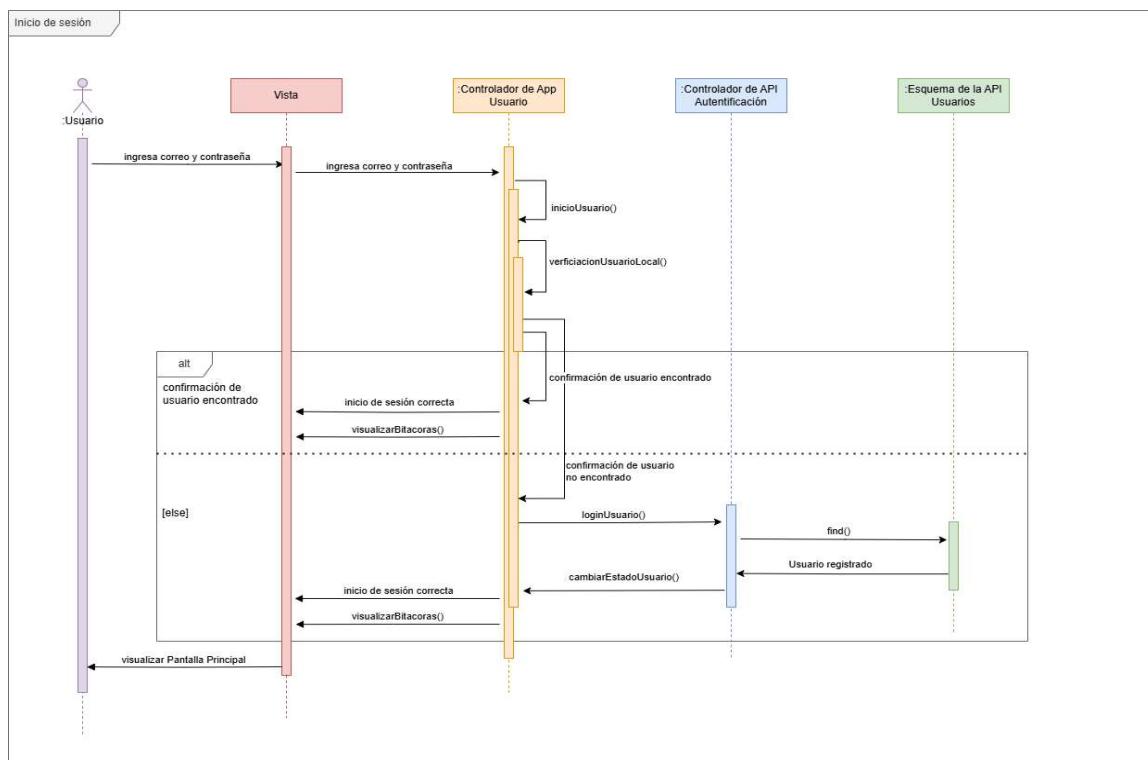


Figura 31 Diagrama de secuencia de inicio de sesión

## **g. Mockups.**

Los mockups presentados para la aplicación IdBird abarcan diversas pantallas clave que forman parte de la funcionalidad integral del sistema. A continuación, se describe brevemente cada una:

1. **Logo de la Aplicación:** Representa la identidad visual de IdBird, diseñado para ser simple y reconocible.
2. **Pantalla de Inicio de Sesión:** Permite a los usuarios ingresar sus credenciales para acceder a la aplicación.
3. **Pantalla de Registro de Usuarios:** Proporciona un formulario para que nuevos usuarios se registren ingresando su nombre, apellidos, correo electrónico y contraseña.
4. **Pantalla Principal (Bitácoras de Campo):** Muestra una lista de bitácoras registradas por el usuario, con opciones para ver detalles, exportar en PDF y crear nuevas bitácoras.
5. **Pantalla de Bitácora:** Detalla los muestreos realizados dentro de una bitácora específica, permitiendo la visualización y creación de nuevos muestreos, así como la exportación de la bitácora en formato PDF.
6. **Pantalla de Muestreo:** Facilita la captura de fotografías para la identificación de aves, mostrando la información relacionada con la especie identificada.
7. **Alertas:** Elementos diseñados para mejorar la experiencia del usuario, notificando cambios o acciones realizadas dentro del sistema.
8. **Pantalla de Confirmación de Usuario Web:** Informa a los usuarios sobre la confirmación de su registro en el sistema después de verificar su correo electrónico.

Para más detalles y especificaciones técnicas de estos y otros mockups, se recomienda consultar el Documento de Diseño, donde se encuentran todos los detalles necesarios para una comprensión completa del diseño implementado a la aplicación. De momento se muestra parte de los mockups realizados en la Figura 16, que contiene el logo principal de la aplicación y las tres primeras pantallas.

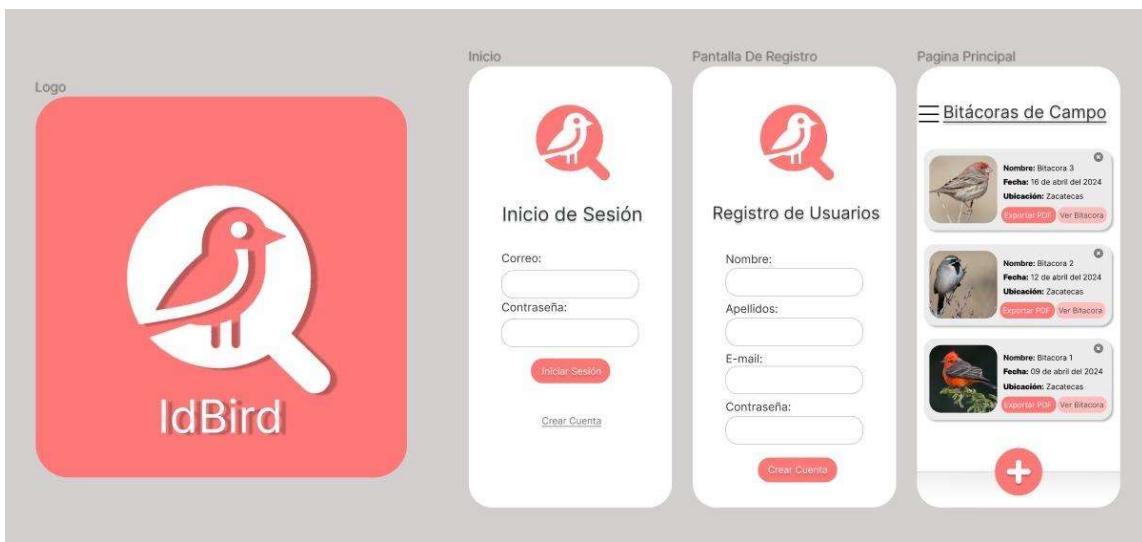


Figura 32 Diseño de prototipos

Todos los diagramas y mockups mostrados, así como todos aquellos creados que forman parte del diseño del sistema, se encuentran en el documento de diseño en el anexo E.

## Análisis de resultados.

Durante la fase de Trabajo Terminal 1 (TT-I) del proyecto " Sistema para muestreo de aves en la ciudad de Zacatecas", las etapas iniciales del ciclo de vida del software para el desarrollo de un sistema, agregando la planificación del proyecto para empezar, se realizó el análisis de requerimientos y el diseño del sistema. A lo largo de estas etapas, se siguieron los planes establecidos y se trabaja conforme al ritmo de trabajo y tareas establecidas, sin embargo también se enfrentaron diversos contratiempos que fueron abordados mediante estrategias efectivas de mitigación y resolución de problemas.

## **Planificación de proyecto**

Durante la fase de planificación, se llevó a cabo una investigación exhaustiva que resultó en la decisión de utilizar la metodología del modelo en V. Aunque inicialmente se había considerado un enfoque ágil debido a la modularidad posible del proyecto, se optó finalmente por el modelo en V debido a su similitud con el modelo en cascada, pero con la ventaja añadida de incluir pruebas continuas en cada una de sus etapas, garantizando un proceso integral y detallado y un software de alta calidad.

Por otro lado, la creación del cronograma resultó ser un proceso más prolongado de lo esperado al iniciar, ya que pasó por múltiples revisiones. En cada revisión, se añadieron diversas tareas adicionales para asegurar la finalización completa y de alta calidad de cada documento y requerimiento. Esta atención al detalle y la rigurosidad en la planificación fueron constantes a lo largo del desarrollo del proyecto para garantizar su éxito.

## **Análisis de Requerimientos**

En la fase de análisis de requerimientos, el levantamiento de estos fue ágil, ya que las reuniones con el cliente siempre fueron breves y concisas. Sin embargo, lo que tomó más tiempo fue el análisis detallado de los requerimientos y la determinación de cómo se implementarían en el proyecto, incluyendo la redacción y aseguramiento de la congruencia de cada parte.

Otro aspecto que complicó el avance fue la determinación del funcionamiento planificado para cada una de las partes del proyecto.

Incluyendo identificar las características preferibles que debían tener las imágenes capturadas para que el modelo pudiera clasificar correctamente las especies de aves, realizando pruebas con diferentes dispositivos móviles de gammas medias a altas y comprobando la calidad de imágenes y la necesidad de distancia y zoom.

Asimismo, fue necesario decidir el lenguaje de programación para la aplicación móvil, asegurando su compatibilidad con los modelos de Redes Neuronales Convolucionales (CNN) y su popularidad en el mercado de aplicaciones móviles actuales; finalmente, se eligió Kotlin por su uso y compatibilidad usual en diversos proyectos distintos.

También se seleccionó un modelo CNN que fuera preciso y ligero para funcionar en un dispositivo móvil; optando por el modelo MobileNet de TensorFlow, previamente probado en un trabajo preliminar “Image Recognition System for Bird Sampling in the City of Zacatecas” [28], por su alta precisión y poca necesidad de almacenamiento y procesamiento ideales para dispositivos móviles.

Además, se planteó una forma enviar de correos electrónicos a través de una API, según los requerimientos del cliente para la confirmación de registro de usuarios a través de correos, a través de la API de correos PostageApp.

### **Diseño del Sistema**

Durante la fase de diseño del sistema, surgieron múltiples sobreestimaciones y subestimaciones que modificaban ligeramente los tiempos establecidos pero gracias a la organización se mantuvo el ritmo de trabajo estable y los tiempos sin muchos cambios.

La parte de la arquitectura, al estar basada en el arquitectura c4, requirió la elaboración de cuatro diferentes diagramas. Entre estos, los diagramas de componentes y de código fueron los más laboriosos, ya que fue en este momento cuando el equipo de trabajo determinó concretamente cómo funcionarían en conjunto las distintas partes de la aplicación.

Asimismo, los diagramas de actividades fueron más numerosos de lo esperado, lo que llevó tiempo adicional. Sin embargo, los diagramas de secuencia fueron los que más tiempo consumieron debido a la falta de familiaridad del equipo con su elaboración. A pesar de esto, estos diagramas fueron verificados múltiples veces para garantizar que se hicieran correctamente.

Por el contrario, los diagramas restantes, como los de casos de uso, de base de datos, manejo de archivos, entre otros, resultaron ser mucho más sencillos y rápidos de lo esperado, lo que permitió recuperar el tiempo sin mayores retrasos y avanzar conforme al cronograma establecido.

## **Contriempos y Resoluciones**

Inicialmente, se contemplaron alrededor de 11 riesgos posibles durante el desarrollo del proyecto. Afortunadamente, solo se presentaron 4 de estos riesgos, que al ser manejados con las medidas de mitigación y prevención adecuadas, no tuvieron un impacto grave en el proyecto.

Uno de los principales contratiempos fue la falta de asistencia en ciertos días de trabajo, lo que se mitigó replanificando las actividades y redistribuyendo las responsabilidades entre los miembros presentes. Esta flexibilidad en la gestión del tiempo permitió que el proyecto continuara avanzando sin interrupciones significativas.

Así mismo, el conflicto de puntos de vista dentro del equipo sobre detalles del diseño y técnicas de implementación se resolvió recurriendo a la opinión del director, el asesor y el cliente del proyecto, logrando un consenso que alineó los objetivos del equipo y mejoró la cohesión y cooperación.

Por ultimo, los contratiempos de sobreestimación y subestimación de los tiempos que requeriría cada actividad fue uno de los mas comunes pero sin grandes impactos ya que continuamente se realizaban planeaciones semanales y por etapas de las actividades necesarias y el tiempo disponible para cada una, por lo que se mantuvo el tiempo sin modificaciones que afectaran gravemente al proyecto.

Finalmente, el análisis de resultados muestra que, a pesar de los contratiempos enfrentados, el equipo del proyecto pudo completar las actividades planificadas con una eficiencia notable, reduciendo casi a la mitad el tiempo estimado inicialmente. La clave del éxito radicó en la replanificación dinámica, la colaboración efectiva y la capacitación continua, asegurando que todas las etapas del proyecto se completaran satisfactoriamente.

## Conclusiones y Recomendaciones

El proyecto "IdBird" ha completado exitosamente su fase de Trabajo Terminal 1 (TT-I), abarcando la planificación del proyecto, el análisis de requerimientos y el diseño del sistema. Durante este primer periodo se logró una comprensión profunda de las necesidades del usuario y la dirección del proyecto, se definieron los objetivos y se establecieron las bases arquitectónicas para el desarrollo del sistema.

A pesar de enfrentar varios contratiempos, como la falta de asistencia y los conflictos de puntos de vista, el equipo demostró una notable capacidad de adaptación y resolución de problemas, logrando completar las tareas con una eficiencia que redujo casi a la mitad el tiempo estimado inicialmente. Este éxito sienta una adecuada base para la siguiente fase del proyecto, asegurando que el equipo esté bien preparado para abordar los desafíos de la siguiente etapa.

No obstante, se destacaron ciertas recomendaciones para un mejor trabajo futuro.

**Mejora en la Gestión del Tiempo:** Durante TT-I, la replanificación y redistribución de tareas fueron cruciales para mitigar la falta de asistencia. En TT-II, se recomienda implementar un sistema de seguimiento del tiempo más riguroso y proactivo. Esto incluiría un registro diario de horas y actividades realizadas, de igual forma ajustar las asignaciones de tareas según sea necesario, asegurando que el proyecto mantenga su curso sin interrupciones significativas.

Se propone el siguiente plan de proyecto para el desarrollo del periodo de trabajo terminal 2, ilustrado en la Figura 17, considerando las etapas de implementación, validación, despliegue del sistema y realización del reporte de TT2, además de horas de capacitación necesaria.

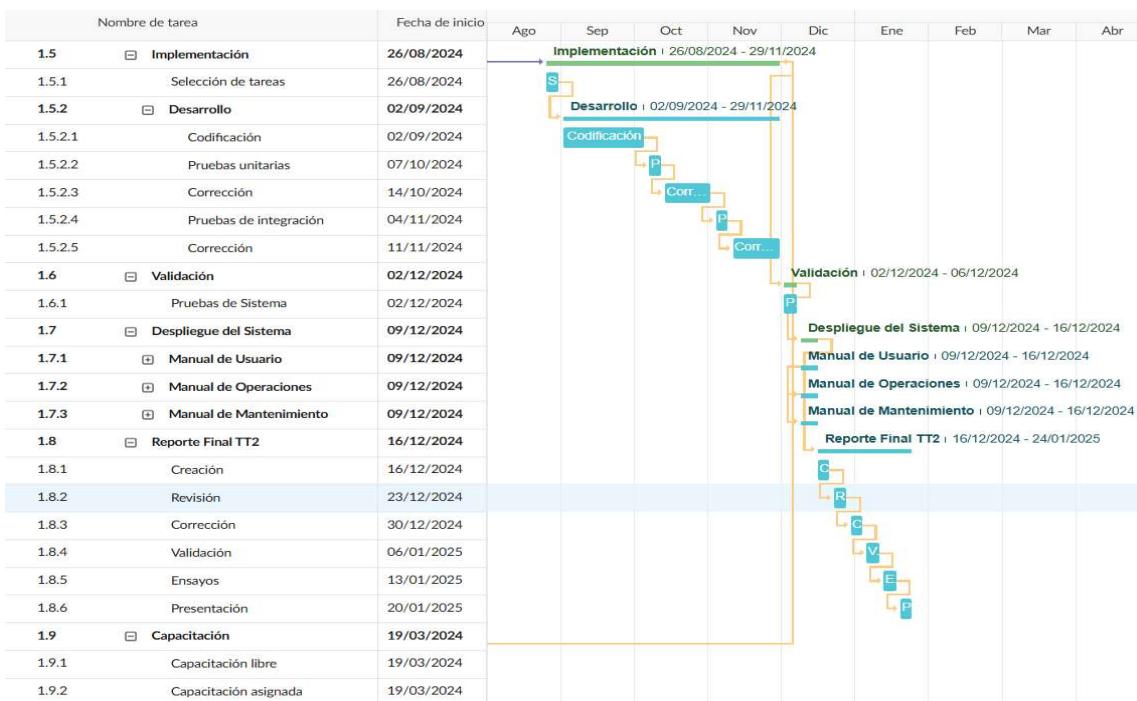


Figura 33 Cronograma propuesto para TT2

**Capacitación Continua y Especializada:** La capacitación en TT-I se realizó de manera continua y adaptativa, lo cual fue efectivo. En TT-II, se recomienda planificar sesiones de capacitación más estructuradas y específicas, centradas en temas claves que se identifiquen como necesarias para el desarrollo. Asegurando que todos los miembros del equipo tengan las habilidades necesarias para realizar cada una de las partes de implementación de manera eficiente.

**Optimización de Recursos y Herramientas:** Aprovechando las lecciones aprendidas en TT-I, se recomienda revisar y optimizar las herramientas y recursos utilizados durante el desarrollo. Evaluar la eficiencia de las herramientas de software, como los lenguajes de programación, modelos CNN para identificación y clasificación de imágenes, uso de softwares externos, etc.; metodologías de trabajo y prácticas de gestión de proyectos puede conducir a una mayor productividad y a la reducción de tiempos de desarrollo en TT-II.

Al implementar estas recomendaciones, el equipo de "Sistema para muestreo de aves en la ciudad de Zacatecas" puede mejorar aún más su eficiencia y efectividad, aspirando que la fase de TT-II se desarrolle de manera fluida y exitosa, cumpliendo con los objetivos establecidos y expectativas del proyecto.

## Fuentes de consulta

- [1] A. G. Navarro-Sigüenza, M. F. Rebón-Gallardo, A. Gordillo-Martínez, A. T. Peterson, H. Berlanga-García, and L. A. Sánchez-González, "Biodiversidad de aves en México," Rev. Mex. Biodiv., vol. 85, no. Suppl, pp. S476-S495, 2014.
- [2] N. Pérez-Valadez, "Adiciones a la avifauna del estado de Zacatecas," Huitzil, vol. 17, no. 2, pp. 175-183, 2016.
- [3] F. Yang, N. Shen, and F. Xu, "Automatic Bird Species Recognition from Images with Feature Enhancement and Contrastive Learning," Appl. Sci., vol. 14, no. 10, p. 4278, 2024.
- [4] J. L. Navarro-Arteaga, J. L. Alanis-Mendez, M. Hernandez-Sanchez, and F. Limon-Salvador, "Bird inventory from Poza Rica, Veracruz, Mexico," Huitzil, vol. 22, no. 1, e604, 2021.
- [5] M. González-Jaramillo, E. Martínez, L. G. Esparza-Olgún, and J. L. Rangel-Salazar, "Actualización del inventario de la avifauna de la Reserva de la Biosfera de Calakmul, península de Yucatán, México: abundancia, estacionalidad y categoría de conservación," Huitzil, vol. 17, no. 1, 2016.
- [6] J. L. Navarro-Arteaga, J. L. Alanis-Mendez, M. Hernandez-Sanchez, and F. Limon-Salvador, "Bird inventory from Poza Rica, Veracruz, Mexico," Huitzil, vol. 22, no. 1, e604, 202
- [7] "New deep learning AI tool helps ecologists monitor rare birds through their songs," ScienceDaily, November 15, 2023.
- [8] F. Yang, N. Shen, and F. Xu, "Automatic Bird Species Recognition from Images with Feature Enhancement and Contrastive Learning," Appl. Sci., vol. 14, no. 10, p. 4278, 2024.
- [9] "Merlin Bird ID – Free, instant bird identification help and guide for thousands of birds," [Online]. Available: <https://merlin.allaboutbirds.org/>.
- [10] eBird Mobile - eBird," [Online]. Available: <https://ebird.org/about/ebird-mobile/>
- [11] "Merlin Bird ID – Free, instant bird identification help and guide for thousands of birds," [Online]. Available: <https://merlin.allaboutbirds.org/>
- [12] eBird Mobile - eBird," [Online]. Available: <https://ebird.org/about/ebird-mobile/>
- [13] "Birdsnap Dataset | Papers With Code," [Online]. Available: <https://paperswithcode.com/dataset/birdsnap>
- [14] "Geoportal del Sistema Nacional de Información sobre Biodiversidad," [Online]. Available: <http://www.conabio.gob.mx/informacion/gis/>
- [15] Information system | Definition, Examples, & Facts | Britannica", Britannica, 2024

- [16] “Características de las aves - Resumen,” ecologiaverde.com. <https://www.ecologiaverde.com/caracteristicas-de-las-aves-1630.html>
- [17] “Características de la cámara del móvil: te explicamos todos los términos,” El Blog de BEEP Informática, Apr. 21, 2021. <https://blog.beep.es/camara-del-movil-caracteristicas-significado-terminos/>
- [18] “Procesado de imágenes digitales,” la.mathworks.com. <https://la.mathworks.com/discovery/digital-image-processing.html>
- [19] “Reconocimiento de patrones (Pattern Recognition),” la.mathworks.com. <https://la.mathworks.com/discovery/pattern-recognition.html>
- [20] C. Kolosky, “What is an Inventory Management System?,” Online Database Software: Use Knack For Custom Online Databases & Systems, Oct. 16, 2023. <https://www.knack.com/blog/what-is-an-inventory-management-system/>
- [21] “Informática Básica: ¿Qué es una aplicación móvil?,” GCFGlobal.org. <https://edu.gcfglobal.org/es/informatica-basica/que-es-una-aplicacion-movil/1/>
- [22] R. S. Yadav, "Improvement in the V-Model," International Journal of Scientific & Engineering Research, vol. 3, no. 2, pp. 1-8, Feb. 2012. Available: <http://www.ijser.org>.
- [23] R. S. Yadav, "Improvement in the V-Model," International Journal of Scientific & Engineering Research, vol. 3, no. 2, pp. 1-8, Feb. 2012. Available: <http://www.ijser.org>.
- [24] C. C. Calderón Tapia, S. Hernández Ríos, H. M. Rincón Bedoya, L. V. Cruz Martínez, C. C. Murillo, and C. A. Gaitán Sánchez, "Protocolo para el Modelamiento de Arquitectura de Software," Gestión de Tecnologías de la Información, Código: GTI-PROT-03, Versión 1, pp. 1-15, Oct. 2021.
- [25] "MVC - MDN Web Docs Glossary: Definitions of Web-related terms," MDN Web Docs, 2024. [Online]. Available: <https://developer.mozilla.org/en-US/docs/Glossary/MVC>. [Accessed: 04-Jun-2024]
- [26] "MVC - MDN Web Docs Glossary: Definitions of Web-related terms," MDN Web Docs, 2024. [Online]. Available: <https://developer.mozilla.org/en-US/docs/Glossary/MVC>. [Accessed: 04-Jun-2024]
- [27] J. Vivanco, “El modelo C4 de documentación para la Arquitectura de Software,” Medium, Jun. 05, 2019. <https://medium.com/@javivanco/el-modelo-c4-de-documentaci%C3%B3n-para-la-arquitectura-de-software-424704528390> (accessed Jun. 04, 2024).
- [28] V. S. Sánchez Lee, A. F. Félix Jiménez, I. I. Belmonte and E. F. Parra González, "Image Recognition System for Bird Sampling in the City of Zacatecas," 2023 12th International Conference On Software Process Improvement (CIMPS), Cuernavaca, Morelos, Mexico, 2023, pp. 126-135, doi: 10.1109/CIMPS61323.2023.10528852. keywords: {Image recognition;Biological system modeling;Urban areas;Neural networks;Birds;Software;Software reliability;birds;image processing;Zacatecas City;pattern recognition;convolutional neural networks}