

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 1 de 134

**Aplicación móvil para la organización e identificación de plantas medicinales
del Pueblo Yanacona del suroriente del departamento del Cauca**



Autores:

Jineth Manzano Díaz

Erwin Omar Solarte Suárez

Facultad de Ingenierías
Corporación Universitaria de Comfacaucá - Unicomfacaucá
Programa de Ingeniería de Sistemas
Popayán, Cauca
2024

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 2 de 134

**Aplicación móvil para la organización e identificación de plantas medicinales
del Pueblo Yanacona del suroriente del departamento del Cauca**



Autores:

Jineth Manzano Díaz

Erwin Omar Solarte Suárez

Directora:

PhD. Vanessa Agredo Delgado

Facultad de Ingenierías
Corporación Universitaria de Comfacaucá - Unicomfacaucá
Programa de Ingeniería de Sistemas
Popayán, Cauca
2024

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 3 de 134

DATOS GENERALES DEL PROYECTO DE GRADO	
Facultad:	Ingeniería
Programa Académico:	Ingeniería De Sistemas
Título del proyecto:	Aplicación móvil para la organización e identificación de plantas medicinales del Pueblo Yanacona del suroriente del departamento del Cauca
Grupo Investigación:	MIND
Línea de Investigación:	Desarrollo de software y usabilidad
Proyecto institucional asociado:	
Fecha de aprobación del perfil:	
Fecha de elaboración de documento de tesis:	Octubre/30/2024
Fecha de radicación:	

Datos Estudiante(s)			
Nombres y apellidos	Cédula de Ciudadanía	Correo electrónico	Número de celular
Jineth Manzano Díaz	1061601879	jinethmanzano@unicomfacaUCA.edu.co	3114161711
Erwin Omar Solarte Suárez	1061540523	erguinsolarte@unicomfacaUCA.edu.co	3023749253

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 4 de 134

Tabla de Contenido

I.	Capitulo	17
1.1.	Introducción	17
1.2.	Planteamiento del problema	18
1.3.	Justificación	21
1.4.	Objetivos.....	26
1.4.1.	Objetivo general.....	26
1.4.2.	Objetivos específicos	26
1.5.	Metodología de investigación.....	27
1.6.	Organización del documento	31
II.	CAPÍTULO.....	32
2.1.	Estado del arte.....	32
2.1.1.	Uso de aplicaciones móviles	33
2.1.2.	Repositorios como herramienta de información	34
2.1.3.	Medicina tradicional	35
2.1.4.	Patrimonio cultural inmaterial.....	36
2.2.	Trabajos relacionados.....	37
2.2.1.	Repositorios para preservar la información.....	37
2.2.2.	Las aplicaciones móviles en los aportes a la salud	38

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 5 de 134

2.2.3. Identificación y procesamiento de imágenes de plantas de cultivo para la industria agrícola 40

III.	Capítulo.....	42
3.1.	Procedimientos para recopilar información	42
3.1.1.	Resultados encuestas a comuneros	43
3.1.2.	Resultado entrevista médico tradicional.....	45
3.1.3.	La sábila	47
3.1.4.	El eucalipto	49
3.1.5.	La hierbabuena.....	51
3.1.6.	La manzanilla.....	53
IV.	Capítulo.....	56
4.1.	Metodología de desarrollo de software	56
4.2.	Selección de la Metodología	57
4.2.1.	Justificación de la Metodología Elegida	57
4.3.	Implementación de la Metodología	61
4.3.1.	Estructura del Equipo.....	61
4.3.2.	Reuniones scrum.....	62
4.4.	Ciclo de vida del desarrollo	63
4.4.1.	Planificación	63
4.4.2.	Requerimientos funcionales.....	63

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 6 de 134

4.4.3.	Requerimientos no funcionales	65
4.4.4.	Toma de fotografías, organización de ellas por carpetas	68
4.4.5.	Diseño	70
4.4.6.	Historias de usuario	73
4.4.7.	Diagrama de secuencia	76
4.4.8.	Diagrama de flujo.....	78
4.4.9.	Diagrama de caso de uso	80
4.5.	Diseño de arquitectura	81
4.5.1.	Componentes de la arquitectura de la aplicación móvil.....	81
4.5.2.	Arquitectura del Modelo de Clasificación VGG16.....	82
4.6.	Implementación	83
4.6.1.	Desarrollo backend	84
4.7.	Desarrollo frontend	92
4.8.	Pruebas	109
V.	CAPÍTULO.....	110
5.1.	Pruebas unitarias del modelo archivo entrenamiento	110
5.1.1.	Interpretación Graficas de entrenamiento	113
5.2.	Pruebas unitarias del modelo archivo predicción.	120
5.3.	Pruebas unitarias del modelo archivo servidor.....	122

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 7 de 134

5.3.1. Pruebas de funcionamiento del modelo con su respectivo api (entrenamiento, predicción, servidor).....	122
5.4. Pruebas unitarias de las interfaces de la aplicación en Android:	126
5.5. Pruebas integrales de funcionamiento backend y frontend	130
5.6. Validación	144
7.2 Análisis heurístico.....	146
5.7. Resultado de evaluación de utilidad de la aplicación	148
VI. Capítulo.....	150
6.1. Conclusiones	150
6.2. Trabajos futuros.....	152
VII. Bibliografía	153

Tabla de Figuras

Figura 1: Sábila.....	47
Figura 2: Eucalipto	49
Figura 3: Hierbabuena	51
Figura 4: Manzanilla.....	53
Figura 5: Sprint	58
Figura 6: sprint 1 recopilación y análisis de información.....	58
Figura 7: sprint 2 recopilación de fotos.....	59
Figura 8: sprint 3 desarrollo backend	59

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 8 de 134

Figura 9: sprint 4 diseño.....	59
Figura 10: sprint 5 desarrollo frontend.....	60
Figura 11: Carpeta de data set plantas	68
Figura 12: Carpeta Eucalipto.....	68
Figura 13: Carpeta Hierbabuena.....	69
Figura 14: Carpeta manzanilla	69
Figura 15: Carpeta sábila	70
Figura 16: Mockups interfaz 1 y 2	71
Figura 17: Mockups interfaz 3 y 4	72
Figura 18: Mockups interfaz 7 ,8 y 9	72
Figura 19: Mockups interfaz 10,11 y 12	73
Figura 20: Diagrama de secuencia.....	78
Figura 21: Diagrama de flujo	80
Figura 22: Diagrama de caso de uso	81
Figura 23: Arquitectura modelo vgg16 [47].....	83
Figura 24: código importación de librerías.....	86
Figura 25: Código modelo entrenamiento	88
Figura 26: Código modelo entrenamiento.	88
Figura 27: Código predicción	89
Figura 28: Código servidor	91
Figura 29: Entorno de Android Studio	93
Figura 30: Creación en Android studio de Activitys	94
Figura 31: Interfaz 1 Android studio	95

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 9 de 134

Figura 32: Interfaz 2 Android studio	96
Figura 33: Importación de librerías de Android studio	97
Figura 34: lógica botones.....	98
Figura 35: Lógica cámara y botón enviar	98
Figura 36: Lógica botón galería.....	99
Figura 37: Permisos cámara / galería	99
Figura 38: Solicitud HTTP POST.....	100
Figura 39: Coroutines.....	101
Figura 40: Respuesta servidor flask.....	102
Figura 41: Contenedor principal	105
Figura 42: Interfaz principal.....	106
Figura 43: Interfaz Menú	106
Figura 44: Interfaz plantas	107
Figura 45: Interfaz lista de enfermedades	107
Figura 46: Interfaz descripción por plantas.....	108
Figura 47: interfaz enfermedades por cada planta	108
Figura 48: Interfaz recetas y usos de cada planta	109
Figura 49: Consola entrenamiento inicio	111
Figura 50: Consola entrenamiento terminado	113
Figura 51: Gráfica entrenamiento 1.....	114
Figura 52: Gráfica entrenamiento 2.....	116
Figura 53: Gráfica entrenamiento 3.....	117
Figura 54: Gráfica entrenamiento 4.....	119

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 10 de 134

Figura 55: Consola predicción.....	121
Figura 56: Consola servidor	122
Figura 57: Servidor 1 corriendo.....	123
Figura 58: Prueba 1 de predicción en servidor	123
Figura 59: Prueba 2 de predicción en servidor	124
Figura 60: Prueba 3 de predicción en servidor	124
Figura 61: Prueba 4 de predicción en servidor	125
Figura 62: Prueba 5 de predicción en servidor	125
Figura 63: Prueba 6 de servidor en corriendo	126
Figura 64: Emulador 1 Android	127
Figura 65: Consola Logcat	127
Figura 66: Emulador 2 Android	128
Figura 67: Emulador 3 Android	129
Figura 68: Emulador 4 Android	129
Figura 69: Emulador 5 Android	130
Figura 70: Emulador 6 Android	130
Figura 71: Navegador 1 anaconda.....	131
Figura 72: Navegador 2 anaconda.....	132
Figura 73: conexión de frontend con backend.....	133
Figura 74: Conexión de servidor establecida	133
Figura 75: Prueba de Conexión servidor con Android	134
Figura 76: consola 1 gradle.....	134
Figura 77: consola 2 gradle.....	135

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 11 de 134

Figura 78: prueba ejecución 1 en móvil.....	135
Figura 79: prueba ejecución 2 en móvil.....	136
Figura 80: prueba ejecución 3 en móvil.....	137
Figura 81: prueba ejecución 4 en móvil.....	138
Figura 82: Figura 105: prueba ejecución 5 en móvil.....	138
Figura 83: prueba ejecución 6 en móvil.....	139
Figura 84: consola servidor ejecutando app desde móvil.....	140
Figura 85: prueba ejecución 7 en móvil.....	141
Figura 86: prueba ejecución 8 en móvil.....	141
Figura 87: prueba ejecución 9 en móvil.....	142
Figura 88: prueba ejecución 10 en móvil.....	143
Figura 89: prueba consola de servidor ejecución 9 en móvil	143
Figura 90: prueba usabilidad usuario 1	145
Figura 91: prueba usabilidad usuario 2	145

INDICE DE TABLAS

Tabla 1: Resultado encuestas a comuneros 1	43
Tabla 2: Resultado encuestas a comuneros 2	44
Tabla 3: Resultado entrevista médico tradicional	45
Tabla 4: Requerimientos funcionales	64
Tabla 5: Requerimientos no funcionales	66
Tabla 6: Historia de usuario 001	74

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 12 de 134

Tabla 7: Historia de usuario 002	74
Tabla 8: Historia de usuario 003	74
Tabla 9: Historia de usuario 004	75
Tabla 10: Historia de usuario 005	75
Tabla 11: Historia de usuario 006	76

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 13 de 134

Agradecimientos

Durante el desarrollo de esta tesis, han sido muchas las personas e instituciones que han brindado su apoyo y orientación, contribuyendo significativamente a su desarrollo y éxito.

En primer lugar, queremos expresar nuestro más sincero agradecimiento al pueblo Yanacona por su apertura, disposición y confianza en este proyecto. Su valiosa participación, conocimientos sobre medicina tradicional y generosidad al compartir sus prácticas ancestrales han sido fundamentales para la creación de esta aplicación móvil. Agradecemos a cada uno de los médicos tradicionales, comuneros y miembros de la comunidad que nos permitieron aprender y entender mejor el legado cultural que desean preservar.

Agradecemos también a la Corporación Universitaria Unicomfacaucá – Unicomfacaucá, por brindarnos el espacio, los recursos y la formación necesaria para llevar a cabo este proyecto. El apoyo de nuestros docentes y tutores fue esencial para superar los desafíos del desarrollo de la aplicación, y sus enseñanzas nos han permitido aplicar metodologías efectivas, innovadoras y pertinentes para lograr una herramienta que esperamos sea de gran utilidad para la comunidad.

Finalmente, un agradecimiento especial a nuestras familias y amigos, quienes, con su paciencia, apoyo constante y motivación, nos han acompañado en cada etapa de este proceso. Su confianza en nosotros fue un pilar importante en la culminación de esta tesis.

A todos, muchas gracias.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 14 de 134

Resumen

Esta tesis presenta el desarrollo de una aplicación móvil cuyo objetivo es organizar e identificar plantas medicinales utilizadas en la medicina tradicional del pueblo Yanacona, una comunidad indígena del suroriente del departamento del Cauca, Colombia. La aplicación está diseñada para servir como un recurso de fácil acceso para miembros de la comunidad, proporcionando información detallada sobre las plantas, incluyendo sus nombres, imágenes de las plantas con las que se trabajó, características y usos terapéuticos. Mediante una interfaz intuitiva y el uso de tecnologías de reconocimiento de imágenes, la app permite a los usuarios identificar plantas a través de fotografías.

Se realizaron pruebas de usuario para validar la usabilidad de la aplicación con miembros de la comunidad Yanacona y usuarios externos, empleando un formato de evaluación heurística. Esta evaluación asigna una puntuación a preguntas basadas en principios heurísticos clave, como principios generales, lenguaje y redacción, rotulación y organización, estructura y navegación, diseño de la interfaz, elementos multimedia, ayuda, accesibilidad y navegación, donde se obtuvo como resultado que se está cumpliendo con los criterios de la evaluación heurística.

Las encuestas de evaluación de utilidad indican que la aplicación es útil debido a que está cumpliendo con el objetivo de organizar e identificar plantas de la medicina tradicional Yanacona, proporcionando información específica sobre cada planta y sus usos. Los usuarios valoraron positivamente la precisión de la funcionalidad de identificación de plantas. Además, la evaluación permitió a los usuarios expresar acuerdos y desacuerdos sobre las funcionalidades e información de la aplicación, así como sugerir mejoras y contenido adicional.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 15 de 134

Este proyecto pretende contribuir a la preservación y transmisión de conocimientos tradicionales, apoyando el interés de la comunidad en proteger y difundir su herencia medicinal de manera digital y accesible.

Palabras clave: Yanacona, medicina tradicional, plantas medicinales, aplicación móvil, preservación cultural, reconocimiento de imágenes.

Astract.

This thesis presents the development of a mobile Application whose objective is to organize and identify medicinal plants used in the traditional medicine of the Yanacona people, an indigenous community in the southeast of the department of Cauca, Colombia. The Application is designed to serve as an easily accessible resource for community members, providing detailed information about the plants, including their names, images of the plants worked with, characteristics and therapeutic uses. Through an intuitive interface and the use of image recognition technologies, the app allows users to identify plants through photographs.

User tests were conducted to validate the usability of the application with members of the Yanacona community and external users, using a heuristic evaluation format. This evaluation assigns a score to questions based on key heuristic principles, such as general principles, language and wording, labeling and organization, structure and navigation, interface design, multimedia elements, help, accessibility and navigation, with the result that the heuristic evaluation criteria are being met.

The usability evaluation surveys indicate that the application is useful because it is fulfilling the objective of organizing and identifying plants of the Yanacona traditional medicine, providing specific information about each plant and its uses. Users positively valued the accuracy of the

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 16 de 134

plant identification functionality. In addition, the evaluation allowed users to express agreements and disagreements about the application's functionalities and information, as well as to suggest improvements and additional content.

This project aims to contribute to the preservation and transmission of traditional knowledge, supporting the community's interest in protecting and disseminating their medicinal heritage in a digital and accessible way.

Key words: Yanacona, traditional medicine, medicinal plants, mobile Application, cultural preservation, image recognition.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 17 de 134

I. Capítulo

1.1. Introducción

Este capítulo fija el marco inicial para el desarrollo del proyecto, abordando los principales aspectos que fundamentan la investigación y el diseño de la aplicación.

En primer lugar, se presenta el planteamiento del problema, en el que se analizan los desafíos y las oportunidades relacionados con la preservación del conocimiento ancestral sobre las plantas medicinales en el contexto del Pueblo Yanacona. Se examinan las amenazas que enfrenta este saber tradicional debido a la modernización y la pérdida de prácticas comunitarias, así como la necesidad de crear herramientas tecnológicas que promuevan su difusión y conservación.

En la justificación, se refiere la importancia de desarrollar una solución digital que no solo permita salvaguardar el conocimiento ancestral del pueblo Yanacona, sino que también lo haga accesible para las nuevas generaciones. La incorporación de la tecnología en este contexto se presenta como una herramienta clave para fortalecer la identidad cultural y asegurar la continuidad de las prácticas medicinales tradicionales.

Seguidamente, se detallan los objetivos del proyecto, tanto generales como específicos, que orientan el desarrollo de la aplicación. Estos objetivos incluyen la identificación y documentación de las plantas medicinales utilizadas por la comunidad Yanacona, así como el desarrollo de una plataforma tecnológica capaz de almacenar, organizar y compartir dicha información de manera intuitiva y eficiente.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 18 de 134

Luego la metodología de investigación, explica los enfoques y las técnicas utilizadas para recopilar información sobre las plantas medicinales y sus usos, así como las estrategias empleadas en el diseño y desarrollo de la aplicación. Se hace énfasis en el proceso participativo con la comunidad Yanacona para garantizar que el producto final refleje fielmente sus conocimientos y necesidades.

Finalmente, se presenta la organización del documento, en la que se describen los capítulos subsiguientes que abordan los distintos aspectos del proyecto, desde el estado del arte y la revisión literaria, hasta la implementación técnica y la validación de la aplicación dentro de la comunidad.

1.2. Planteamiento del problema

El suroccidente andino de Colombia, específicamente el departamento del Cauca, es una de las regiones más diversas del país, habitada por varias comunidades indígenas, entre ellas, el pueblo Yanacona. Los Yanaconas se consolidaron como grupo étnico diferenciado a finales de la década de los ochenta, vinculando su identidad histórica y cultural con la sociedad Inca [1]. Actualmente, el pueblo Yanacona se encuentra distribuido en diversas regiones de Colombia, viviendo bajo la autoridad de los cabildos indígenas [2]. De acuerdo con la organización nacional indígena de Colombia-ONIC, la lengua ancestral (denominada quechua del pueblo Yanacona se encuentra en proceso de recuperación, no obstante, en la actualidad se hace uso del español [3].

Por otro lado, a nivel global, la Organización Mundial de la Salud (OMS) estima que el 80% de la población utiliza plantas medicinales para atender sus necesidades de salud. [4]. No obstante, la medicina tradicional ha perdido relevancia debido a la falta de documentación y

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 19 de 134

transmisión del conocimiento sobre el uso de estas plantas Por esta razón, la OMS promueve estrategias para la conservación y uso de la medicina tradicional, resaltando la necesidad de proteger el patrimonio cultural inmaterial de las comunidades indígenas [5].

Hoy en día el uso de aplicaciones móviles para el reconocimiento y difusión de plantas medicinales en comunidades indígenas de Colombia aún es limitado y está en desarrollo, principalmente debido a barreras de acceso a la tecnología y la falta de integración de conocimientos tradicionales en plataformas digitales. Sin embargo, la necesidad de conservar este conocimiento ha motivado iniciativas que promueven el uso de aplicaciones que identifican plantas y brindan información sobre sus beneficios, recetas tradicionales y aplicaciones medicinales [6].

las aplicaciones también representan una oportunidad para crear una plataforma diferenciada que recoja y respete el conocimiento ancestral, un aspecto crítico dado que muchas de estas comunidades han sido las custodias de la biodiversidad y del conocimiento botánico que hoy se emplea en la medicina tradicional y herbolaria. La Universidad Nacional de Colombia, por ejemplo, ha abordado esta necesidad mediante iniciativas de conservación y divulgación que integran aspectos culturales y científicos, enfatizando la conexión entre biodiversidad y las prácticas tradicionales en comunidades locales [7].

Es por esto que, en este contexto, es fundamental un enfoque diferencial que valore la diversidad natural y cultural, promoviendo la equidad y justicia social en iniciativas relacionadas con la salud y el patrimonio cultural [8]. La medicina tradicional, considerada un conocimiento empírico y ancestral [9], ha sido esencial en la supervivencia de los pueblos indígenas, basada en la relación entre el ser humano y las plantas medicinales [10].

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 20 de 134

Sin embargo, se ha identificado que las prácticas relacionadas con la medicina tradicional del pueblo Yanacona están en riesgo de desaparecer, lo que amenaza su patrimonio oral e inmaterial [11]. En el departamento del Cauca, especialmente en zonas rurales, el acceso a la medicina moderna ha contribuido a la disminución de las prácticas tradicionales, en un contexto de urbanización y cambios en los estilos de vida [12]. Es crucial reconocer la importancia de ambas formas de medicina, integrándose para preservar el conocimiento ancestral sin desatender los avances en salud.

Actualmente, el avance de la tecnología ha permitido el desarrollo de aplicaciones móviles enfocadas en la educación, la información y la promoción de la salud. [13] Sin embargo, no existen aplicaciones específicamente diseñadas para apoyar la conservación de los conocimientos tradicionales de la población Yanacona. Esto nos impulsó a llevar a cabo la investigación y el desarrollo de la aplicación móvil que facilite preservar y organizar el conocimiento ancestral sobre el uso de plantas medicinales es fundamental contar con repositorios digitales que permitan el acceso a este saber y aseguren su transmisión a las futuras generaciones de la comunidad. [14].

De esta problemática se deriva la pregunta de investigación: ¿Cómo se puede apoyar al pueblo Yanacona en el suroccidente del departamento del Cauca, Colombia, en la organización e identificación de las plantas medicinales que forman parte de su medicina tradicional?

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 21 de 134

1.3. Justificación

La pérdida de conocimientos ancestrales en diversas comunidades indígenas es una preocupación creciente, ya que estos saberes tradicionales son fundamentales para la identidad cultural y la sostenibilidad de dichas comunidades. Existen varios factores que contribuyen a la ausencia de estos conocimientos [15].

Uno de los factores es la integración forzada o voluntaria de comunidades indígenas en sociedades dominantes y la migración hacia áreas urbanas, esto puede conducir al abandono de prácticas y tradiciones ancestrales. La devaluación y el desprecio hacia los saberes tradicionales, junto con la ausencia de mecanismos efectivos para su protección, desincentivan la transmisión de estos conocimientos a las nuevas generaciones [15].

El cambio climático y degradación ambiental ataca directamente las prácticas tradicionales relacionadas con la naturaleza, como la medicina herbal y la agricultura sostenible, poniendo en riesgo la supervivencia de estos conocimientos [16].

Con los anteriores factores de pérdida de conocimiento ancestral en las comunidades nacen consecuencias que tienen implicaciones profundas:

En primer lugar, se genera la erosión de la identidad cultural ya que las tradiciones y saberes ancestrales son pilares de la identidad de las comunidades; su pérdida puede generar una desconexión con las raíces culturales y una disminución de la cohesión social. También se genera la amenaza a la biodiversidad, muchos conocimientos tradicionales están vinculados a prácticas de conservación y uso sostenible de recursos naturales; su desaparición puede acelerar la degradación ambiental [17].

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 22 de 134

Para salvaguardar estos saberes diversas organizaciones y comunidades están implementando estrategias, como la documentación y transmisión donde se recopilan y registran prácticas tradicionales en formatos accesibles para las nuevas generaciones, promoviendo su enseñanza en escuelas y comunidades. Así mismo estrategias para el reconocimiento legal y protección de derechos donde se aboga por la creación de normativas que reconozcan y protejan la propiedad intelectual y cultural de las comunidades indígenas, evitando el uso indebido y la apropiación de sus conocimientos [15].

También la promoción del diálogo intercultural, donde se fomenta espacios de intercambio entre conocimientos tradicionales y ciencia occidental puede enriquecer ambas perspectivas y contribuir a la solución de problemas globales [18].

El presente proyecto se centró en el valor de lo que se conoce hoy como medicina tradicional, ya que es un conjunto de saberes y prácticas basadas en las propiedades curativas de las plantas medicinales, cuya transmisión cultural se ha mantenido a través de la tradición oral. Por tanto, es ineludible darse cuenta del debilitamiento de estos usos y costumbres en la medicina tradicional, lo que está afectando de manera negativa a la cultura del cuidado para el tratamiento de malestares y enfermedades desde el saber ancestral [17].

Es importante reconocer que la relación entre la medicina tradicional y la medicina moderna no debe entenderse como un conflicto o una competencia, sino como una oportunidad para el diálogo y hacer de estos tipos de medicina un complemento. Aunque algunas personas puedan sentir que la medicina moderna ha contribuido al debilitamiento de las prácticas tradicionales, esta percepción no implica un rechazo hacia los avances médicos contemporáneos. En lugar de verlas como opuestas, se busca resaltar la necesidad de integrar

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 23 de 134

ambas formas de conocimiento para enriquecer y fortalecer la salud y el bienestar de las comunidades [5].

En un mundo cada vez más digital, la incorporación de herramientas tecnológicas es fundamental para la preservación y promoción de la medicina tradicional ya que permite un acceso rápido y eficiente a la información sobre prácticas y conocimientos ancestrales. Las herramientas tecnológicas son ampliamente utilizadas, lo que facilita la difusión de saberes entre las generaciones más jóvenes y promueve la interacción con la comunidad. Además, pueden incorporar recursos multimedia, como videos y guías interactivas, que enriquecen el aprendizaje y la comprensión de los usos medicinales de las plantas [19].

En la observación del decrecimiento de la medicina tradicional, el propósito de esta investigación fue hacer uso adecuado de herramientas tecnológicas que permitan el fortalecimiento y la conservación de la medicina tradicional para el Pueblo Yanacona, así como de los conocimientos que actualmente existen únicamente en los médicos tradicionales. Es fundamental preservar estos saberes ancestrales, ya que son portadores de una rica herencia cultural y de prácticas medicinales que han sido transmitidas de generación en generación. En este sentido, el enfoque se centra en el pueblo Yanacona, ubicado en el suroccidente del Cauca, asegurando que los beneficiarios directos de esta investigación sean los habitantes de la comunidad, quienes son los guardianes de esta valiosa tradición y su conocimiento sobre el uso de plantas medicinales.

A partir de la información anterior se analizó que la mejor herramienta para fortalecer y conservar la medicina tradicional del Pueblo Yanacona era una aplicación móvil o App (abreviatura de Application), la cual se entiende como “...una herramienta diseñada para

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 24 de 134

desarrollar una función específica en una plataforma concreta: móvil, tablet [20]. Es el equipo tecnológico usado en esta comunidad y esto no solo permite una documentación más efectiva de los saberes tradicionales, sino que también asegura que estas prácticas se mantengan vivas y accesibles, beneficiando así a la comunidad al facilitar la transmisión del conocimiento entre generaciones.

La aplicación móvil funciona a partir de un servidor local, con el cual es posible introducir, organizar, conservar y gestionar los datos relacionados con la diversidad de plantas y sus usos de acuerdo con el conocimiento ancestral. Este conocimiento proviene de la experiencia de los mayores del Pueblo Yanacona y de otras fuentes confiables, lo que garantiza una información completa, fidedigna y útil, en aras de dar continuidad a los procesos de fortalecimiento cultural. Además, se desarrolló una aplicación que permite acceder a esta información de manera instantánea y fluida, facilitando su uso en la vida cotidiana [21].

La herramienta incorpora la funcionalidad de reconocimiento de imágenes para identificar plantas a partir de fotografías capturadas por el usuario. Una vez que la planta es identificada, la aplicación proporciona información detallada sobre su uso medicinal, así como el nombre de la planta, una descripción completa de sus usos, beneficios y formas de consumo. También cuenta con un menú el cual permite consultar un listado de enfermedades las cuales cada una cuenta con sus respectivas plantas que la tratan junto con sus respectivos usos y beneficios. Esto facilita un acceso rápido y eficiente a información crucial sobre la medicina tradicional. Con el desarrollo de la aplicación móvil, es posible acceder a la información sobre el uso de plantas medicinales a partir del uso de un celular. Se considero pertinente que esta herramienta tecnológica fuera de fácil manejo y usable.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 25 de 134

Al interactuar con la aplicación, el usuario tiene la opción de capturar una imagen de una planta medicinal utilizando la cámara de su dispositivo móvil o seleccionar una imagen previamente almacenada en la galería. Una vez que la imagen es procesada, la aplicación despliega información detallada sobre la planta reconocida. Este despliegue incluye sus beneficios medicinales, diversas formas de preparación, y una explicación específica sobre los usos tradicionales dentro del contexto del Pueblo Yanacona. Además, se especifica para qué enfermedades o condiciones se utiliza la planta, de acuerdo con el conocimiento ancestral de la comunidad. Esta funcionalidad no solo facilita el acceso a un valioso saber tradicional, sino que también promueve la continuidad y preservación del uso de plantas medicinales en la vida cotidiana de los Yanaconas “...las aplicaciones móviles se han transformado en el principal canal de comunicación de la sociedad, obteniendo el primer lugar en versatilidad y eficiencia al momento de obtener información importante y fidedigna a cualquier hora del día o de la noche y mantener al usuario actualizado de cualquier tema que este disponga” [22].

La aplicación representa una oportunidad única para unir el valor cultural y tradicional de la comunidad Yanacona y mantener el conocimiento de las personas pertenecientes a esta, considerando que, las plantas medicinales tienen un profundo significado cultural y tradicional en la comunidad, siendo una parte integral de la identidad y la herencia cultural, lo que hace que su preservación sea una prioridad y de igual manera se busque el promover y divulgar el conocimiento referente a esto.

Otros de los factores que nos motivaron a centrar nuestro proyecto de grado en la comunidad Yanacona es su localización geográfica, ya que los pueblos en los que residen se encuentran a escasa distancia de la ciudad. Además, un miembro del equipo que desarrolla este

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 26 de 134

proyecto forma parte de la comunidad Yanacona, y su deseo es preservar y compartir su rico legado cultural a través de la documentación visual para futuras generaciones.

1.4. Objetivos

1.4.1. Objetivo general

Construir una aplicación software para organizar e identificar plantas que son usadas en la medicina tradicional de la comunidad del pueblo Yanacona del suroriente del departamento del Cauca, brindando información propia sobre las plantas y sobre sus usos.

1.4.2. Objetivos específicos

- Determinar a partir de la revisión de la literatura y de los conocimientos de los médicos tradicionales y demás personas conocedoras del pueblo Yanacona del suroriente del departamento del Cauca, un estado del arte relacionado con la clasificación de las plantas usadas en la medicina tradicional y sus respectivos usos actuales.
- Desarrollar una aplicación software que permita identificar una planta usada en medicina tradicional de la comunidad del pueblo Yanacona y que brinde su información y sus posibles usos en dicha medicina tradicional.
- Validar la utilidad de la aplicación software mediante su aplicación y uso en el pueblo Yanacona.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 27 de 134

1.5. Metodología de investigación

El desarrollo del trabajo de grado se llevó a cabo utilizando la metodología de investigación-acción multi-ciclo con bifurcación [23], tal como se había establecido en el documento de anteproyecto. Este enfoque metodológico incluyó tres ciclos fundamentales: conceptual, metodológico y de evaluación, cada uno con actividades clave que contribuyeron al éxito del proyecto.

Durante el ciclo conceptual, se realizó un análisis exhaustivo para caracterizar el conocimiento conceptual sobre la medicina tradicional dentro de la comunidad Yanacona. Este análisis se basó en entrevistas a médicos tradicionales y en la aplicación de encuestas a los comuneros del pueblo Yanacona, ubicado en el suroriente del departamento del Cauca, otra fuente de información fue un libro de medicina tradicional que pertenece a la comunidad Yanacona, el cual documenta detalladamente las plantas medicinales, sus usos terapéuticos y las preparaciones tradicionales. Este texto, considerado un recurso valioso por su contenido culturalmente específico, fue consultado en formato PDF el cual solo se tiene acceso desde la coordinación del programa de salud del pueblo Yanacona, ya que fue elaborado por los propios miembros de la comunidad. La información obtenida permitió no solo incluir la percepción de la medicina tradicional en la comunidad, sino también establecer las bases para la planificación de la investigación, definir los objetivos específicos y de igual forma determinar cada uno de los elementos que se debería incluir en la aplicación.

En el ciclo metodológico se realizó el desarrollo de la aplicación móvil para identificar plantas utilizadas en la medicina tradicional del pueblo Yanacona y proporcionar información detallada sobre sus usos, la cual surge de una necesidad que es la pérdida progresiva del

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 28 de 134

conocimiento ancestral en la comunidad y la ausencia de un sistema centralizado para organizar y consultar dicha información como se explica en la justificación del documento. Para abordar este problema, se realizó un análisis detallado de los requerimientos, el cual permitió seleccionar las tecnologías adecuadas para la implementación.

Lo primero que se realizó fue identificar los requerimientos (capítulo 4) a partir de entrevistas con médicos tradicionales Yanacona y encuestas a miembros de la comunidad. Esto permitió comprender las expectativas y necesidades específicas. Se definieron los requerimientos funcionales donde detallamos que la aplicación debía ser capaz de identificar plantas mediante imágenes utilizando inteligencia artificial, permitir la consulta de información detallada sobre las propiedades, usos y preparaciones de las plantas, ofrecer menús organizados por categorías (enfermedades, plantas, recetas) y funcionar con imágenes representativas de las plantas. Además, se estableció que una parte esencial de la funcionalidad debería ser accesible sin conexión a internet.

También, se definieron los requerimientos no funcionales como la facilidad de uso, considerando usuarios con distintos niveles de alfabetización tecnológica, un desempeño ágil para la identificación de imágenes y navegación, compatibilidad con dispositivos Android accesibles para la comunidad, escalabilidad para permitir futuras mejoras y seguridad para proteger los datos almacenados y enviados.

Con base a los requerimientos, se seleccionaron las tecnologías más adecuadas, utilizando para el reconocimiento de imágenes redes neuronales convolucionales (CNN) entrenadas con TensorFlow y Keras, empleando un modelo preentrenado como VGG16 mediante transferencia de aprendizaje, lo cual garantizó precisión con un conjunto de datos

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 29 de 134

limitado [24]. En el backend, se eligió Flask por su ligereza y facilidad para implementar APIs REST que permitieran la comunicación con la aplicación móvil. Para el frontend, se desarrolló una aplicación en Android Studio con Kotlin. Finalmente, la comunicación entre frontend y backend se implementó mediante una API REST, que permitía enviar imágenes al servidor Flask y recibir las predicciones correspondientes.

El diseño de la interfaz utilizó XML para garantizar una experiencia moderna y flexible. La interfaz se diseñó pensando en la facilidad de navegación, con menús organizados y accesibles, y funcionalidades esenciales priorizadas para la primera versión. Este enfoque aseguró que la aplicación respondiera a las necesidades reales de la comunidad, fomentando la preservación del conocimiento tradicional y mejorando su acceso

Para el desarrollo de esta aplicación se hizo uso de la metodología de desarrollo scrum que permitió dividir el desarrollo en sprints cortos, lo que facilitó la adaptación a los cambios a medida que avanzaba el proyecto. Cada sprint culminaba con la entrega de una versión funcional e incremental de la aplicación, garantizando que se cumplieran los objetivos planteados en cada etapa. Además, la metodología promovió una comunicación constante entre los miembros del equipo, lo que favoreció la rápida identificación de problemas y la implementación de soluciones.

Así mismo con la flexibilidad de Scrum, se pudo enfocar en entregar una aplicación que no solo identificara las plantas de forma precisa, sino que también brindara un acceso intuitivo y organizado al conocimiento ancestral sobre sus usos medicinales.

Para el ciclo de evaluación, se procedió a validar la utilidad de la aplicación mediante su implementación en la comunidad Yanacona. Además, se llevaron a cabo pruebas unitarias para asegurar el correcto funcionamiento de los componentes clave como el archivo de entrenamiento

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 30 de 134

del modelo, el archivo de predicción y el archivo del servidor. Luego, se realizaron pruebas funcionales del modelo en conjunto con su API, abarcando las fases de entrenamiento, predicción y la comunicación con el servidor.

De igual forma, se llevaron a cabo pruebas unitarias para cada una de las interfaces desarrolladas en Android Studio. Al finalizar el desarrollo de la aplicación, se realizó una prueba completa para verificar la navegación dentro del dispositivo y asegurar la coherencia en la experiencia de usuario.

Finalmente, se efectuaron las pruebas integrales uniendo el backend y el frontend para comprobar el correcto funcionamiento de la aplicación en su conjunto, garantizando que todos los componentes operaran de manera sincronizada y eficiente.

La aplicación fue probada directamente en el entorno de la comunidad Yanacona, permitiendo la identificación de cuatro plantas principales que se manejan en la comunidad (manzanilla, eucalipto, hierbabuena y sábila). Estas pruebas confirmaron que la aplicación entregaba la información correcta y resultaba útil para la comunidad, cumpliendo con los objetivos del proyecto.

Sin embargo, durante las pruebas, se evidenció un problema cuando se giraba la pantalla del dispositivo móvil, la aplicación no se adaptaba correctamente al cambio de orientación y dejaba de funcionar como se esperaba. Para solucionar este inconveniente temporalmente se realizó una modificación en el archivo AndroidManifest.xml de Android, deshabilitando la rotación automática de la pantalla. De esta forma, la aplicación se mantiene siempre en orientación vertical, evitando problemas de funcionamiento al girar el dispositivo, quedando como mejora a

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 31 de 134

futuro que se adapte a los giros de la pantalla, esta mejora garantizara una experiencia de usuario más estable y sin interrupciones.

1.6. Organización del documento

Después de este primer capítulo, el documento se organiza en cinco capítulos adicionales que abarcan los aspectos fundamentales del desarrollo de la propuesta de grado:

En el segundo capítulo, se presenta el estado del arte y se realiza una revisión exhaustiva de la literatura relevante, abarcando estudios previos sobre etnobotánica, etnomedicina y tecnología aplicada a la preservación del conocimiento tradicional de la comunidad Yanacona.

En el tercer capítulo, se describe el proceso de revisión de la literatura y la recolección de información sobre las plantas medicinales Yanaconas, utilizando fuentes como libros tradicionales, encuestas comunitarias, y entrevistas con médicos tradicionales.

En el cuarto capítulo, se detalla la metodología Scrum utilizada en el desarrollo de la aplicación, mostrando cada una de las actividades desarrolladas desde la definición de requisitos hasta la implementación, destacando las tecnologías empleadas, como Python y Android.

En el quinto capítulo, se aborda cómo se validó la aplicación con la comunidad Yanacona, analizando la retroalimentación obtenida y evaluando la usabilidad y aceptación de la herramienta.

Finalmente, en el sexto y último capítulo se presentan las conclusiones del trabajo realizado, se sugieren mejoras y futuras investigaciones, y se incluye una bibliografía completa que respalda el proyecto. Además de los capítulos mencionados, se incluyen algunos apéndices que contienen información adicional relevante como cuestionarios y código de programación.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 32 de 134

II. CAPÍTULO

2.1. Estado del arte

El conocimiento sobre las plantas medicinales utilizadas por el pueblo Yanacona ha sido transmitido oralmente durante generaciones, desempeñando un papel crucial en la preservación de la salud y en la conexión cultural con el entorno natural. Sin embargo, con el avance de la medicina moderna y la falta de herramientas tecnológicas adecuadas, este conocimiento enfrenta el riesgo de desaparecer. Esta investigación propone el desarrollo de una aplicación que no solo facilite la conservación de este saber, sino que lo haga accesible a futuras generaciones.

A nivel global, la importancia de las plantas medicinales ha sido reconocida como un recurso clave para los sistemas de salud, particularmente en comunidades rurales. [25] En Colombia, las prácticas etnobotánicas tienen una relevancia significativa debido a la biodiversidad y la riqueza cultural del país. En el contexto Yanacona, las plantas medicinales como el eucalipto, la sábila, la manzanilla y la hierbabuena se utilizan ampliamente para tratar dolencias respiratorias, cutáneas, nerviosas y digestivas, respectivamente. [26]

Aunque existen aplicaciones que permiten la identificación de plantas mediante redes neuronales convolucionales, estas herramientas no incorporan información terapéutica específica ni contextos culturales. La propuesta presentada busca llenar este vacío al desarrollar una aplicación que combine inteligencia artificial con una base de datos culturalmente relevante sobre las plantas medicinales del pueblo Yanacona.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 33 de 134

2.1.1. Uso de aplicaciones móviles

En la actualidad las aplicaciones para la comunicación y la información son de uso cotidiano debido a la facilidad con la que se puede acceder a partir de un dispositivo móvil o un servidor. En ocasiones el uso de estas herramientas es tan constante que no es necesario que los usuarios se encuentren a kilómetros de distancia para comunicarse mediante estas plataformas, puesto que están diseñadas de tal modo que los usuarios se sientan plenamente cómodos e incluso se hallen identificados con ciertos recursos o representaciones nada más como un pasatiempo [21].

En el momento de crear una aplicación móvil es necesario evaluar la pro y la contra a la hora de implementarla, pues lo más conveniente es el uso de software libre, pues por una parte no es necesario pagar ninguna licencia de uso y funciona con todos los dispositivos que operan desde Android [22]. Es por ello por lo que, en el uso de aplicaciones es importante considerar el aspecto de framework, el cual alude al conjunto de herramientas para desarrollar cualquier aplicación. Cabe anotar que toda aplicación que se desarrolle para Android utiliza el mismo conjunto de API (Interfaz de Programación de Aplicaciones) y el mismo framework [21].

Por su parte, las aplicaciones cuentan con diversas posibilidades a la hora de definir un dispositivo para su implementación, los dispositivos pueden ser de comunicación, de computación, de reproducción multimedia, de grabación multimedia y consolas portátiles [22]. No obstante, los teléfonos inteligentes o *Smartphone* cuentan con todas las propiedades mencionadas anteriormente, lo cual debe considerarse como un beneficio y un constante desafío de actualización.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 34 de 134

2.1.2. Repositorios como herramienta de información

Los repositorios digitales son una herramienta importante para el manejo de la información, permitiendo al usuario interactuar, extraer y subir información de acuerdo con sus necesidades [27]. Estas herramientas digitales parten de la iniciativa de los años 60's denominada OA (*Open Access*) que a inicios del siglo XXI se expande de manera exponencial a partir de proyectos con compromiso social avalado por declaraciones de orden internacional que sostienen y perfilan la definición de OA [28].

En el contexto de los repositorios digitales utilizados en el proyecto de la aplicación móvil para la identificación de plantas medicinales del pueblo Yanacona, no solo se ofrece la posibilidad de almacenar y acceder a información relevante sobre las plantas y sus usos medicinales, sino que también se integra la tecnología de reconocimiento de imágenes. Esta tecnología permite a los dispositivos móviles analizar visualmente los elementos presentes en una imagen o fotografía.

El reconocimiento de imágenes es una tecnología que permite a los dispositivos móviles identificar y comprender visualmente los elementos presentes en una imagen o fotografía. En el entorno del proyecto de aplicación móvil para la organización e identificación de plantas medicinales del Pueblo Yanacona, el reconocimiento de imágenes se refiere a la capacidad del sistema para analizar una imagen de una planta y determinar automáticamente de qué planta se trata, así como proporcionar información relevante sobre sus usos medicinales y características [29].

En términos más técnicos, el proceso de reconocimiento de imágenes implica la utilización de algoritmos y modelos de aprendizaje automático (*machine learning*), como redes

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 35 de 134

neuronales convolucionales (CNN), para analizar patrones visuales en las imágenes. Estos algoritmos son entrenados previamente con una gran cantidad de imágenes etiquetadas para que puedan aprender a distinguir entre diferentes objetos [29], en este caso, plantas medicinales.

En el entorno del proyecto lo que se busca es que cuando un usuario capture o cargue una imagen de una planta a través de la aplicación móvil, el sistema utilice el reconocimiento de imágenes para comparar las características visuales de la planta en la imagen con las que ha aprendido previamente. Una vez que la planta sea identificada con éxito, el sistema proporcionará información sobre los usos medicinales, propiedades y detalles específicos de esa planta en particular, basándose en la base de datos o el conocimiento almacenado en la aplicación.

2.1.3. Medicina tradicional

De acuerdo con la OMS, se considera que la medicina tradicional es “... el pilar principal de la prestación de servicios de salud, o su complemento” [30], por tanto, la Medicina Tradicional y Complementaria (MTC) cuenta con tres definiciones que se considera importante clarificar, a saber: 1. Medicina Tradicional: posee una antiquísima historia y se considera la suma de todos y cada uno de los conocimientos, capacidades y prácticas basados en teorías, creencias o concepciones, experiencias o vivencias propias de las distintas culturas de las cuales se ha hecho uso para preservar la salud, y prevenir, diagnosticar, mejorar o tratar enfermedades en la integridad del ser humano. 2. Medicina Complementaria: también se conoce como medicina alternativa, la cual constituye una amplia gama de prácticas para la atención en salud, que no hacen parte ni de la medicina tradicional ni de la medicina convencional de ningún país, y tampoco están integradas al sistema de salud dominante en la territorialidad, y 3. Medicina

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 36 de 134

Tradicional y complementaria: esta fusiona tanto los términos de lo tradicional y lo complementario en la medicina abarcando productor, prácticas profesionales [30].

2.1.4. Patrimonio cultural inmaterial

Según la UNESCO, el patrimonio inmaterial o el patrimonio vivo hace referencia a todas y cada una de las prácticas, expresiones, saberes o técnicas que son transmitidos por las diversas comunidades de generación en generación y que tienen una incidencia en su cotidianidad y pervivencia [31], por lo cual constituye una memoria colectiva.

Ante lo cual es importante acuñar que existe una lista Representativa de Patrimonio Cultural Inmaterial, el cual es un

...mecanismo de salvaguardia de manifestaciones y prácticas culturales relevantes para las comunidades o colectividades, las cuales junto con instituciones públicas, organizaciones civiles y entidades privadas se comprometen a salvaguardar, documentar, investigar, promover, fomentar, transmitir y revitalizar dichas manifestaciones, buscando su continuidad en el tiempo como referentes de identidad para las actuales y futuras generaciones [32].

Dentro de la lista representativa de patrimonio cultural se encuentra el médico tradicional, quien es una persona que practica formas de medicina y cuidado de la salud basadas en conocimientos, técnicas y métodos transmitidos a lo largo de generaciones en comunidades y culturas específicas. Estos practicantes suelen estar profundamente arraigados en las tradiciones culturales y en el conocimiento ancestral de su comunidad. Sus enfoques para diagnosticar, tratar y prevenir enfermedades pueden diferir significativamente de las prácticas médicas convencionales occidentales [33] . De igual forma, pueden emplear una variedad de métodos, que incluyen el uso de plantas medicinales, rituales, terapias manuales, técnicas de

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 37 de 134

energía y otros recursos naturales. Sus prácticas a menudo se basan en creencias espirituales y una comprensión holística de la salud, que considera tanto los aspectos físicos como los emocionales y espirituales de la persona [33].

En muchas culturas, los médicos tradicionales desempeñan un papel fundamental en la atención médica comunitaria y son respetados como guardianes del conocimiento ancestral sobre la salud y el bienestar. Sin embargo, es importante reconocer que la efectividad y seguridad de las prácticas de los médicos tradicionales pueden variar, y algunas comunidades han trabajado para integrar estas prácticas con enfoques médicos convencionales para garantizar una atención de salud integral y segura [33].

2.2. Trabajos relacionados

2.2.1. Repositorios para preservar la información

En el trabajo: **Importancia de los repositorios para preservar y recuperar la información**, hace una profundización en la importancia de los Repositorios correspondientes al Sistema Nacional de Salud en Cuba, para lo cual fue necesario el análisis de algunos elementos con el *Open Acces* (OA) y las plataformas más utilizadas para la creación de repositorios como el CWIS (Campus Wide Information System) dejando ver las ventajas y desventajas de esta herramienta [34].

El trabajo: **Repositorios de información como recurso de comunicación académica de universidades nacionales argentinas**, está orientado por el *Open Acces* (OA) como recurso de comunicación académica de sus producciones. En su metodología, el estudio consta de: 1. Descripción de tendencias internacionales del movimiento OA, 2. Descripción del panorama actual de los repositorios vinculados a las instituciones universitarias argentinas, 3. Identificación

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 38 de 134

de los repositorios de información. Los aportes de este trabajo radican en la construcción de un estado de la cuestión sobre el avance del OA en Argentina y reflexiona sobre cómo las Tecnologías de Información y Comunicación (TIC) favorecieron el acceso a la información libre y gratuita a partir del movimiento OA [35].

El trabajo: **Repositorios de acceso abierto en las instituciones de educación superior en México,» *Información, cultura y sociedad*** estuvo basado en el análisis de once repositorios de las Instituciones de Educación Superior en México, haciendo uso de la metodología SCOT que permitió ver las ventajas de incluir la producción académica en repositorios, también pudo verse cada uno de los desafíos tanto a nivel institucional como a nivel personal. Por ende, aparece OpenDOAR como uno de los directorios con mayor acogida a nivel mundial debido a la cantidad y calidad de recursos alojados en los repositorios de OA. Se concluye que deben existir políticas y procedimientos que garanticen la calidad de la información y la adecuada capacitación del personal involucrado para el mantenimiento de cada repositorio, considerando pertinente un equipo de trabajo holístico [36].

2.2.2. Las aplicaciones móviles en los aportes a la salud

El trabajo: **Aplicaciones móviles en salud: usos frecuentes y potenciales riesgo**, da cuenta, mediante investigación documental, de un contexto general para entender el surgimiento del uso de las aplicaciones móviles, su utilidad en las distintas poblaciones, los beneficios que generan en el manejo clínico y los principales riesgos asociados a su uso en Colombia. Incluso se pudo mostrar que en el surgimiento de COVID-19 se usaron aplicaciones como CoronApp y GABO. También se muestra que la población con mayor uso de aplicaciones es la materno-perinatal y se sugiere que puede ser un nicho con resultados favorables. No obstante, respecto

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 39 de 134

a los dilemas éticos y riesgos asociados al uso de aplicaciones, está a falta de regulación y control relacionados con los contenidos divulgados. Por tanto, es necesario que haya una normativa robusta respecto a la seguridad de datos y de calidad en las aplicaciones relacionadas con el cuidado de la salud disponibles en el mercado colombiano [37]

La investigación: **Aplicaciones Médicas Móviles: definiciones, beneficios y ries**, aporta una perspectiva general, a partir de una revisión documental, del uso de aplicaciones médicas, las definiciones de los dispositivos médicos, los estudios referentes a su uso, los potenciales riesgos y las regulaciones existentes en territorio colombiano. El estudio muestra que las principales dificultades en el uso de estas nuevas tecnologías al área de salud, radica en la dificultad para garantizar la idoneidad de los contenidos por lo cual requiere una actitud crítica en cuento a sus contenidos y a su uso [38].

El trabajo de investigación: **Aplicaciones móviles relacionadas con la salud. Un estudio sobre las aplicaciones con funcionalidad para el recordatorio de la toma de Medicamentos**, aborda el uso de aplicaciones para el manejo de la toma de medicamentos en España, en los cuales se realiza un análisis clasificando las aplicaciones por características: características generales de cada aplicación y características propias de cada aplicación para evaluar sus funciones mediante bases de datos especializadas como PubMed, ResearchGate Scientific Network, Scopus, IEEE Xplore, Web of Science (WOS) y Google Scholar. El estudio arrojó que ninguna de las aplicaciones cumple con características deseables. No obstante, se recomienda el desarrollo de diseños generales que estén basados en el usuario a fin de desarrollar aplicaciones móviles para optimizar la adherencia a los medicamentos [39].

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 40 de 134

El trabajo: **aceptación tecnológica de aplicaciones móviles para el servicio de salud a domicilio** tiene como enfoque el comprender la importancia de las aplicaciones para los servicios de atención en salud móvil en España, por tanto, mediante una investigación mixta se realiza un análisis de confiabilidad, una prueba de multicolinealidad y análisis inferencial. Se señala que hay una idoneidad para la continuidad de los servicios de salud móvil a partir de distintas aplicaciones, y que es importante para investigaciones futuras hacer uso del modelo extendido de UTAUT2 [40].

2.2.3. Identificación y procesamiento de imágenes de plantas de cultivo para la industria agrícola

En el trabajo Procesamiento de imágenes para reconocimiento de daños causados por plagas en el cultivo de Begonia semperflorens Link & Otto (flor de azúcar) se aplica el procesamiento digital de imágenes que es una herramienta ampliamente utilizada en la automatización de procesos industriales, dado a que presenta confiabilidad, eficacia y rapidez en el procesamiento.

En este trabajo se presentan los resultados obtenidos en el reconocimiento de plagas utilizando la visión de máquina por computador como elemento de diagnóstico. La captura de las imágenes se realizó por medio de un agente robótico aéreo (drone) equipado con una cámara, lo que permitió capturar las imágenes del estado de las hojas de un cultivo de la planta conocida como ‘flor de azúcar’ (Begonia semperflorens). Estas imágenes fueron procesadas utilizando técnicas de visión de máquina con el fin de identificar el posible ataque de plagas en el cultivo.

Las técnicas utilizadas corresponden a filtros morfológicos, difuminado gaussiano y filtrado HSL. Como resultado principal de este trabajo se detectaron perforaciones de hojas

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 41 de 134

ocasionadas por el ataque de plagas, específicamente babosas, caracoles, arañas rojas y minadoras [41].

En el trabajo Software para identificar y localizar diversas variedades de plantas de quinua en el distrito de Querocoto se realizó una búsqueda en los repositorios de datos de la comunidad científica, para luego analizarlos en busca de patrones que los identifique como parte de las diversas variedades de quinua. Esta investigación apoyó a la comunidad científica ligada a la investigación en agricultura a encontrar variedades de plantas de quinua con características que les permita mejorar y conservar el recurso genético de *Chenopodium Quínoa* [42].

El proyecto tiene como objetivo automatizar el proceso de identificación y localización de las plantas de quinua. Para la localización, se empleó el GPS, mientras que el proceso de identificación se basó en el análisis de imágenes fotográficas mediante técnicas de reconocimiento de patrones. El objetivo general es desarrollar un software capaz de identificar y localizar diversas variedades de la planta de quinua, utilizando procesamiento de imágenes para lograr una identificación precisa y eficiente [42].

En el trabajo Detección de enfermedades en plantas de tomate a través del análisis computacional de sus hojas, se buscó aportar con la detección temprana de enfermedades en las plantas de dichas especies, las cuales se está trabajando en solucionar 4 de las enfermedades que afectan en las plantaciones de tomate. La identificación se puede dar a través de las partes de la planta como son, las hojas, el tallo o los frutos almacenando en una data set de imágenes de hojas y tallos enfermas [43].

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 42 de 134

III. Capítulo

3.1. Procedimientos para recopilar información

Para llevar a cabo la recolección de información sobre las plantas medicinales utilizadas por la comunidad del pueblo Yanacona, se realizó una revisión de la literatura para asegurar la exactitud y transcendencia de los datos recopilados. La revisión, inicialmente, incluyó el análisis de textos académicos, artículos científicos, y libros especializados en etnobotánica y medicina tradicional, con un enfoque particular en las prácticas medicinales utilizadas por el pueblo Yanacona.

Una de las principales fuentes de información fue un documento PDF sobre medicina tradicional elaborado por los comuneros de la comunidad Yanacona, quienes tienen conocimientos profundos sobre plantas medicinales. Este texto documenta detalladamente las plantas, sus usos terapéuticos, beneficios y recetas de preparaciones tradicionales. A cargo de los médicos tradicionales de la comunidad, el documento garantiza que la información recopilada esté alineada con las prácticas y saberes ancestrales, lo que lo convierte en un recurso valioso y culturalmente específico.

Además de la revisión de la literatura, se llevaron a cabo encuestas estructuradas dentro de la comunidad Yanacona. Estas encuestas fueron diseñadas para obtener información directa de los miembros de la comunidad sobre las plantas que utilizan, los métodos de preparación, y los contextos en los que aplican estas medicinas tradicionales. Los datos obtenidos a través de las encuestas proporcionaron un enfoque contemporáneo sobre el uso de las plantas y complementar la información extraída de la literatura.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 43 de 134

3.1.1. Resultados encuestas a comuneros

Tabla 1:

Resultado encuestas a comuneros 1

Nombre del comunero	Resguardo	Conocimiento de plantas medicinales si/no	plantas conocidas	preparación	medio para adquirir conocimiento
Sebastián Jiménez	Yanacona	si	Manzanilla, sábila	Infusiones, emplastos	Tradición familiar
Gabriela Velazco	Yanacona	no	Eucalipto, hierbabuena	cataplasmas, infusiones, té, vaporización	abuelos
Edwin Omen	Yanacona	si	Jengibre, apio, manzanilla	Infusiones, té	padre
Arley Anacona	Yanacona	si	Manzanilla, hierbabuena, eucalipto, sábila, ruda	en infusiones, té, emplastos	madre
Diana mamian	Yanacona	si	Manzanilla, ruda	Infusiones, té	Tradición familiar
José uní	Yanacona	si	Sabila, eucalipto, pronto alivio	Emplastos, ungüento, infusión, vaporización	abuelos
Mary Narváez	Yanacona	si	Cedrón, sábila, hierbabuena, manzanilla	Té, ungüento, infusión	abuelos
Eulices Palechor	Yanacona	si	Eucalipto, cedrón, hierbabuena	Ungüento, té, infusión, vaporización	Tradición familiar
Viviana Anacona	Yanacona	si	Ruda, manzanilla, sábila, eucalipto, cedrón	Infusión, emplastos, té	Tradición familiar

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 44 de 134

Tabla 2:

Resultado encuestas a comuneros 2

Nombre del comunero	¿Le gustaría una app para conocer más de su cultura?	¿Qué le gustaría ver en la app?	Formatos para aprender sobre la medicina tradicional	¿Qué tan familiarizado(a) está con el uso de aplicaciones móviles?	¿En qué dispositivos le gustaría usar la aplicación (teléfono, tableta, computadora)?
Sebastián Jiménez	si	Usos terapéuticos de plantas y fotos detalladas.	Fotos, texto y videos	mucho	teléfono
Gabriela Velazco	si	Recetas y usos de las plantas	Fotos y texto	mucho	teléfono
Edwin Omen	si	Plantas que no conozco	Fotos y videos	mucho	teléfono
Arley Anacona	si	Usos terapéuticos de plantas y fotos detalladas.	Textos y fotos	mucho	teléfono
Diana Mamian	si	Paso a paso e como usar las plantas medicinales	Videos, fotos y texto	medio	teléfono
José uní	si	Ver enfermedades que se pueden tratar con plantas medicinales	Videos y fotos	medio	teléfono
Mary Narváez	si	Recetas y usos	Texto y fotos	bajo	teléfono
Eulices Palechor	si	Enfermedades y plantas con sus recetas	Texto y fotos	bajo	teléfono

Viviana Anacona	si	Imágenes de las plantas y sus recetas de uso	Videos, texto y fotos	bajo	teléfono
-----------------	----	--	-----------------------	------	----------

3.1.2. Resultado entrevista médico tradicional

Tabla 3:

Resultado entrevista médico tradicional

Nombre médico tradicional	Motivo para aprender sobre la medicina tradicional	plantas más comunes que conoce	Propiedades curativas	Técnicas de preparación	nivel de conocimiento sobre la medicina tradicional en las personas en la actualidad
Aracely Narváez	Oportunidad de aprender de un mayor, y su curiosidad de niña por aprender e investigar.	Sabila, eucalipto, hierbabuena. Acetaminofén, ruda, manzanilla, menta, jengibre, tomillo, limoncillo, cedrón, sauco	Respiratorias, calmantes, para sacar fríos, para dolores musculares, heridas, inflamaciones, dolores estomacales, fiebres, tos.	Infusión, vaporización, té, baños. Paños, ungüentos.	, considero que el conocimiento sobre la medicina tradicional ha disminuido con el paso del tiempo, especialmente entre las generaciones más jóvenes. Aunque algunos aún valoran y practican estos saberes, la influencia de la medicina moderna y la urbanización ha hecho que muchas

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 46 de 134

	personas se alejen de las prácticas ancestrales.
--	--

En las tablas presentadas se detalla la información recopilada a partir de las encuestas realizadas a los comuneros y las entrevistas con un médico tradicional de la comunidad Yanacona. Este análisis permitió identificar las plantas medicinales más comunes, sus principales usos y la forma en que se adquirió el conocimiento sobre la medicina tradicional. A partir de esta información, se seleccionaron las cuatro plantas más representativas: manzanilla, sábila, eucalipto y hierbabuena, como base para el desarrollo del proyecto. Adicionalmente, se documentaron recetas y usos tradicionales asociados a estas plantas, así como sugerencias sobre el contenido que los usuarios esperaban encontrar en la aplicación. (Para profundizar en los resultados de las encuestas y entrevistas, se incluye información complementaria en el capítulo 3 de los anexos del documento.)

El desarrollo de la aplicación se centró en estas cuatro plantas debido a su relevancia cultural y terapéutica dentro de la comunidad Yanacona. Asimismo, su accesibilidad en la región facilitó la recolección de imágenes necesarias para entrenar el sistema de clasificación por redes neuronales convolucionales integrado en la aplicación. Este proceso aseguró que el modelo fuera robusto y estuviera basado en representaciones visuales precisas y contextualizadas de las plantas, garantizando un alto nivel de eficacia en el reconocimiento dentro de un entorno real y culturalmente significativo.

A pesar de la rica diversidad de plantas medicinales en la región, muchas otras especies se encuentran en zonas más remotas, como áreas selváticas o de páramo, lo que hace que su

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 47 de 134

acceso sea más complicado y estacional. Estas condiciones dificultan la obtención de un número adecuado de imágenes de dichas plantas para construir un conjunto de datos apropiado para el funcionamiento. Por este motivo, se decidió priorizar las cuatro plantas mencionadas, ya que su disponibilidad constante permite un proceso más eficiente y garantiza la precisión del reconocimiento dentro de la aplicación.

A continuación, se describen sus características, usos y beneficios, información que fue obtenida de las entrevistas con los médicos y repositorios de plantas aromáticas y medicinales.

3.1.3. La sábila



Figura 1: Sábila

Esta planta es conocida también como aloe vera, es una planta que se cultiva en todo el mundo por sus propiedades medicinales y cosméticas. Es conocida por sus hojas gruesas y carnosas que contienen un gel transparente con numerosos beneficios para la salud y la piel.

Algunas características:

- La sábila puede alcanzar hasta 60-100 cm de altura

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 48 de 134

- Las hojas son gruesas, de color verde grisáceo, con bordes dentados. El interior de las hojas contiene un gel transparente.
- Produce flores tubuliformes de color amarillo a naranja en racimos, que aparecen en un tallo floral erecto. La floración ocurre en verano.

Usos:

- El gel de sábila se puede consumir en pequeñas cantidades como complemento alimenticio, aunque no es común en la cocina cotidiana.
- La sábila es famosa por sus propiedades regenerativas y cicatrizantes. El gel se aplica tópicamente para aliviar quemaduras, heridas, irritaciones y dermatitis. También tiene propiedades laxantes cuando se consume en forma de jugo, ayudando a tratar problemas digestivos como el estreñimiento.
- El gel de sábila se utiliza ampliamente en productos para el cuidado de la piel y el cabello debido a sus propiedades hidratantes y antiinflamatorias. Es común en cremas, lociones y champús para mejorar la hidratación y la salud de la piel y el cabello.

Beneficios:

- La sábila es conocida por sus propiedades hidratantes, cicatrizantes y antiinflamatorias. Su uso regular puede ayudar a mejorar la salud de la piel, acelerar la curación de heridas y quemaduras, y promover una digestión saludable. Además, sus propiedades antioxidantes pueden contribuir a la salud general.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 49 de 134

Cultivo:

- La sábila es una planta resistente que se adapta bien a condiciones áridas y cálidas. Prefiere un suelo bien drenado y una ubicación soleada. Se puede propagar fácilmente a través de sus hijuelos o por semillas. Requiere riego moderado y es importante evitar el exceso de agua, ya que esto puede causar pudrición de las raíces. La planta es de bajo mantenimiento y puede crecer en macetas o directamente en el suelo. [44]

3.1.4. El eucalipto



Figura 2: Eucalipto

Es una planta que se caracteriza por sus hojas lanceoladas, de un color verde grisáceo, que contienen aceites esenciales muy apreciados por sus propiedades medicinales. Las hojas jóvenes son más redondeadas y ricas en compuestos volátiles, mientras que las hojas adultas son más largas y delgadas.

Características:

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 50 de 134

- El eucalipto puede variar en tamaño desde arbustos pequeños hasta árboles que alcanzan alturas de más de 60 metros.
- Las hojas del eucalipto son largas, estrechas y lanceoladas. En su etapa juvenil suelen ser redondeadas, pero con la madurez se vuelven alargadas. Tienen una textura cerosa que las protege de la pérdida de agua.
- La corteza del eucalipto varía entre especies, pero puede ser lisa, fibrosa o escamosa. En algunas especies, se desprende en capas, dejando un tronco liso y de colores claros.
- El eucalipto es conocido por su fuerte aroma característico, derivado del aceite esencial presente en sus hojas.
- Las flores del eucalipto no tienen pétalos visibles y están rodeadas de numerosos estambres, lo que les da una apariencia esponjosa. El fruto es una cápsula leñosa que contiene las semillas.

Usos:

- El eucalipto es conocido por su capacidad para aliviar problemas respiratorios. Se utiliza en forma de infusiones, vaporizaciones y aceites esenciales para el tratamiento de resfriados, bronquitis, asma y congestión nasal. Además, tiene propiedades antisépticas, por lo que se emplea para desinfectar heridas y mejorar la cicatrización. También es eficaz como repelente natural de insectos.

Beneficios:

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 51 de 134

- Ayuda a despejar las vías respiratorias al actuar como expectorante, facilitando la expulsión de pulgas.
- Sus aceites esenciales tienen propiedades antimicrobianas, útiles en infecciones leves.
- Alivia dolores musculares y articulares cuando se aplica en forma de ungüento o aceite esencial.
- Su aroma se usa en aromaterapia para reducir el estrés y mejorar la concentración. [44].

3.1.5. La hierbabuena



Figura 3: Hierbabuena

Es una planta conocida por sus hojas aromáticas y sus múltiples usos en la cocina y la medicina tradicional. Originaria de la región mediterránea, es reconocida por sus hojas de color verde intenso, lanceoladas y con bordes dentados, que emiten un aroma fresco y mentolado.

Características:

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 52 de 134

- La hierbabuena puede alcanzar hasta 60 cm de altura.
- Las hojas son de un verde brillante, con un toque rugoso y un aroma característico. Son utilizadas frescas o secas en diversas preparaciones.
- Produce pequeñas flores de color púrpura o blanco, que florecen en racimos durante el verano.

Usos:

- En la cocina, la hierbabuena se utiliza para dar sabor a una variedad de platos, incluyendo ensaladas, salsas y bebidas como el mojito. Su sabor fresco y mentolado es apreciado en la gastronomía de muchas culturas.
- En la medicina tradicional, la hierbabuena se emplea para aliviar problemas digestivos como la indigestión, los gases y los cólicos. También se usa en infusiones para calmar dolores de cabeza y problemas respiratorios.
- El aceite esencial de hierbabuena se usa en aromaterapia para promover la relajación y reducir el estrés.

Beneficios:

- La hierbabuena es conocida por sus propiedades antiinflamatorias, antimicrobianas y digestivas. Su uso regular puede ayudar a mejorar la digestión, aliviar el malestar estomacal y proporcionar un efecto refrescante.

Cultivo:

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 53 de 134

- La hierbabuena se cultiva fácilmente en huertos caseros y jardineras. Prefiere un suelo bien drenado y una ubicación soleada o parcialmente sombreada. La planta se propaga a través de esquejes o semillas, y su crecimiento es vigoroso, lo que a menudo requiere podas regulares para mantener su forma. [44].

3.1.6. La manzanilla



Figura 4: Manzanilla

Es una planta conocida por sus flores blancas con centros amarillos y sus propiedades medicinales. Se valora principalmente por sus flores, que se utilizan en infusiones y extractos para aprovechar sus beneficios para la salud.

Características:

- La manzanilla puede alcanzar hasta 60 cm de altura.
- Las hojas son de color verde claro, finamente divididas y con una textura suave.
- Produce flores pequeñas de color blanco con un centro amarillo, organizadas en capítulos que recuerdan a las margaritas. Florece durante la primavera y el verano.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 54 de 134

Usos:

- Aunque no se usa ampliamente en la cocina, las flores secas de manzanilla se utilizan para preparar infusiones y tés.
- La manzanilla es famosa por sus propiedades calmantes y digestivas. Se usa en infusiones para aliviar problemas digestivos como la indigestión, los gases y los cólicos. También tiene propiedades sedantes y puede ayudar a calmar el insomnio y la ansiedad. Además, se utiliza en compresas para aliviar dolores oculares y problemas de la piel.
- El aceite esencial de manzanilla se usa en aromaterapia para promover la relajación y reducir el estrés.

Beneficios:

- La manzanilla es conocida por sus propiedades antiinflamatorias, antiespasmódicas y calmantes. Su uso regular puede ayudar a mejorar la digestión, reducir el estrés y promover un sueño reparador. Además, tiene propiedades antioxidantes que pueden contribuir a la salud general.

Cultivo:

- La manzanilla se cultiva fácilmente en jardines y huertos. Prefiere un suelo bien drenado y una ubicación soleada. Se puede propagar a través de semillas o esquejes, y la planta requiere cuidados mínimos. Su crecimiento es vigoroso, y se debe podar regularmente para mantener su forma y promover una floración continua. [44].

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 55 de 134

En conjunto, la combinación de la revisión de la literatura, la consulta de un libro tradicional específico, las encuestas a la comunidad y las entrevistas con médicos tradicionales permitió construir una base de datos y culturalmente relevante sobre las plantas medicinales del pueblo Yanacona. Esta información fue esencial para el desarrollo de la aplicación móvil, asegurando que reflejara fielmente los conocimientos y prácticas de la comunidad.

Si bien la aplicación actualmente reconoce cuatro plantas fundamentales la sábila, hierbabuena, manzanilla y eucalipto, es importante destacar que existen datos adicionales sobre otras plantas medicinales, recolectados durante el proceso de investigación. Debido a las limitaciones técnicas y de acceso geográfico, solo se incluyeron estas cuatro plantas en la primera versión de la aplicación. Sin embargo, la información sobre otras plantas ha sido recopilada y está documentada en los anexos, con el objetivo de ampliar la base de datos en futuras actualizaciones. Una base de datos con solo cuatro plantas no sería suficiente para reflejar toda la riqueza del conocimiento Yanacona, por lo que se contempla la inclusión progresiva de más plantas en el futuro.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 56 de 134

IV. Capítulo

4.1. Metodología de desarrollo de software

La metodología de desarrollo de software es una herramienta fundamental en el desarrollo de proyectos que proporciona un marco organizado para guiar cada etapa de este proceso. En el presente proyecto, que comprende tanto la investigación sobre las plantas medicinales utilizadas por el pueblo Yanacona como el desarrollo de una aplicación móvil, la metodología elegida desde la parte de creación de software fue Scrum [45], una de las más renombradas y aptas en el ámbito de la gestión ágil de proyectos.

La metodología Scrum permitió dividir el desarrollo del proyecto en iteraciones cortas y manejables, conocidas como "sprints", donde cada sprint se enfoca en alcanzar objetivos específicos. Esta flexibilidad es decisiva, ya que el proyecto no solo involucra el desarrollo de una aplicación móvil, sino también una investigación etnobotánica sobre las plantas medicinales, sus beneficios, usos, preparaciones y las enfermedades que pueden tratar.

Utilizando Scrum, se busca que el proyecto avance de manera ordenada y adaptativa, permitiendo anexar cambios y mejoras en función de los hallazgos de la investigación o las necesidades de los usuarios. De esta forma, se pretende que tanto los objetivos de investigación como los de desarrollo tecnológico se cumplan de manera efectiva, obteniendo como producto resultante una aplicación móvil que, a través del reconocimiento de plantas medicinales, proporcione información detallada sobre sus usos, preparaciones y beneficios.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 57 de 134

4.2. Selección de la Metodología

4.2.1. Justificación de la Metodología Elegida

Se tuvo en cuenta la metodología Scrum para el desarrollo de este proyecto basándose en las características específicas de la investigación y desarrollo tales como iteración continua, retroalimentación constante, flexibilidad y gestión del equipo que se llevaron a cabo. Scrum es una metodología ágil que facilita la gestión de proyectos complejos y en continua evolución, lo cual es especialmente adecuado para este proyecto que integra la investigación sobre plantas medicinales del pueblo Yanacona con el desarrollo de una aplicación móvil. [46]

A continuación, las principales características de la metodología Scrum:

Inicialmente, la adaptabilidad y flexibilidad de la metodología para realizar la investigación sobre plantas medicinales, sus usos, beneficios, preparaciones y las enfermedades que pueden tratar es un proceso que puede requerir ajustes continuos a medida que se descubren nuevos datos o se refine la comprensión de la información. En este caso, la adaptabilidad y flexibilidad se logró mediante encuestas a los comuneros, lecturas de un libro elaborado por los médicos tradicionales de la comunidad Yanacona, el cual está en formato PDF, y entrevistas a dichos médicos tradicionales. Todo esto se detalla en el capítulo 3.

El enfoque incremental es una característica que garantiza que el proyecto progrese en pequeños pasos controlables, lo que es ideal para que el proyecto que combina tanto la investigación académica como el desarrollo tecnológico. Se realizan seis sprint donde cada uno termina con un producto funcional o un avance significativo, garantizando que tanto la investigación como el desarrollo de la aplicación se completen en fases manejables y revisables.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 58 de 134

A continuación, se muestran imágenes de los sprints realizados utilizando jira para ello:

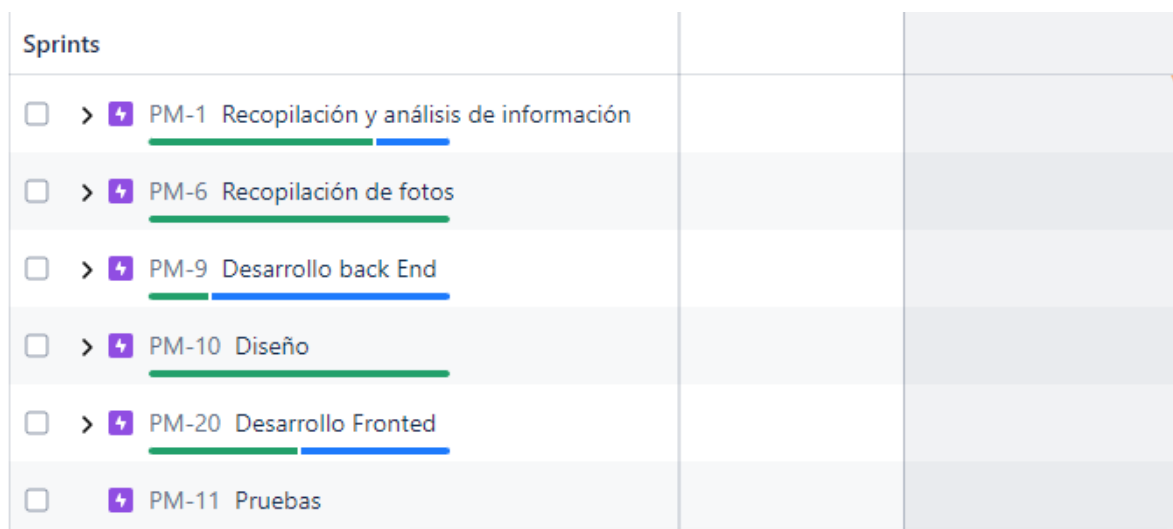


Figura 5: Sprint

En las siguientes figuras se muestran cada sprint con sus respectivas tareas indicadas ya realizadas y el encargado de realizarlas:

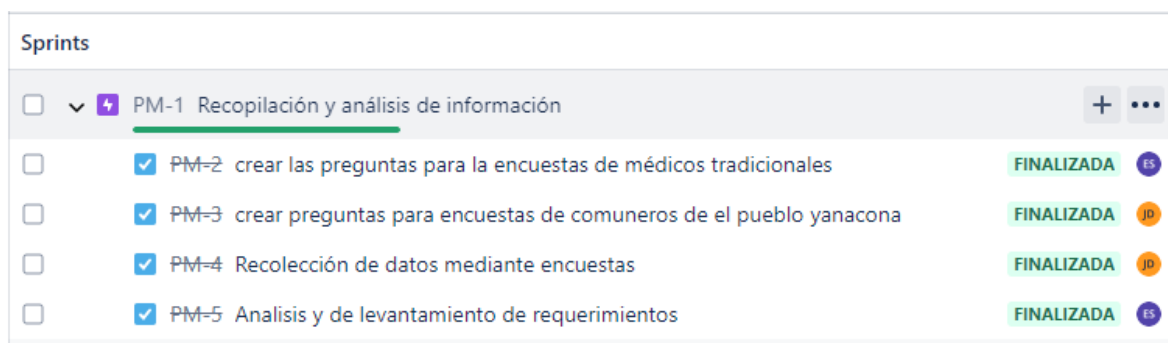


Figura 6: sprint 1 recopilación y análisis de información

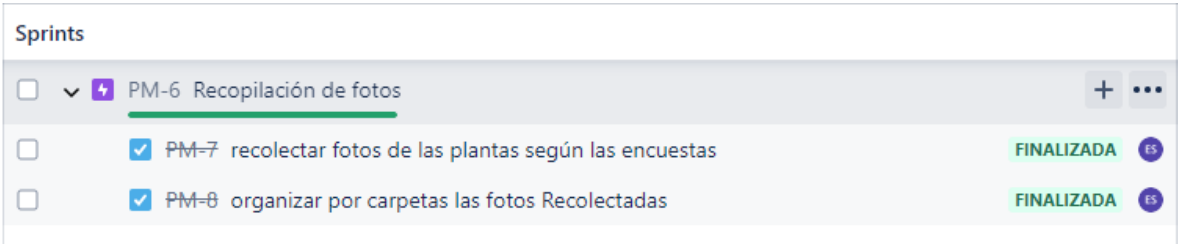


Figura 7: sprint 2 recopilación de fotos

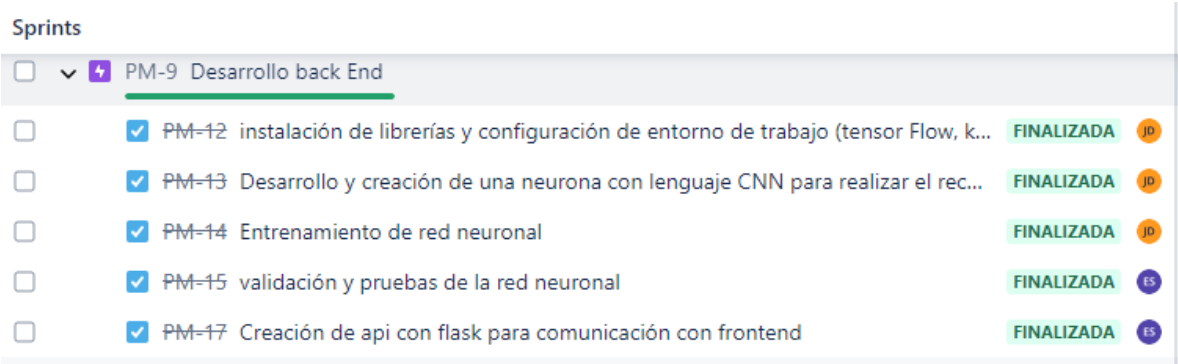


Figura 8: sprint 3 desarrollo backend

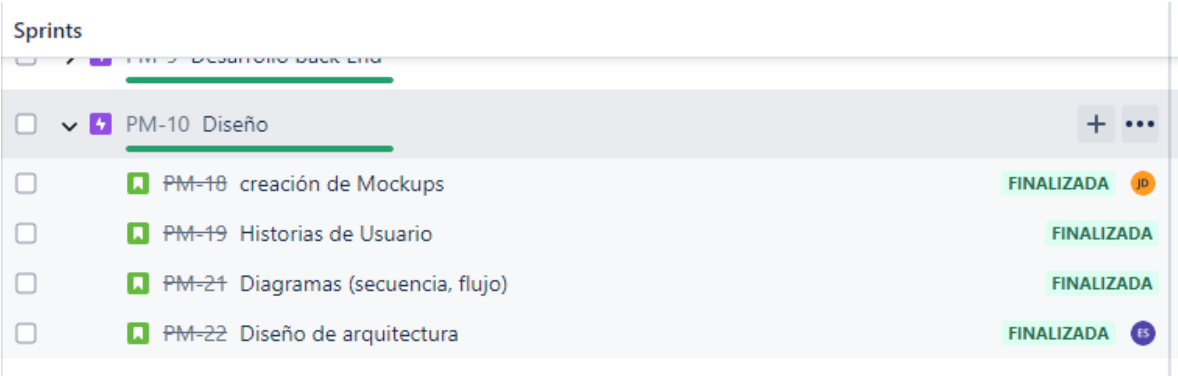


Figura 9: sprint 4 diseño

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 60 de 134

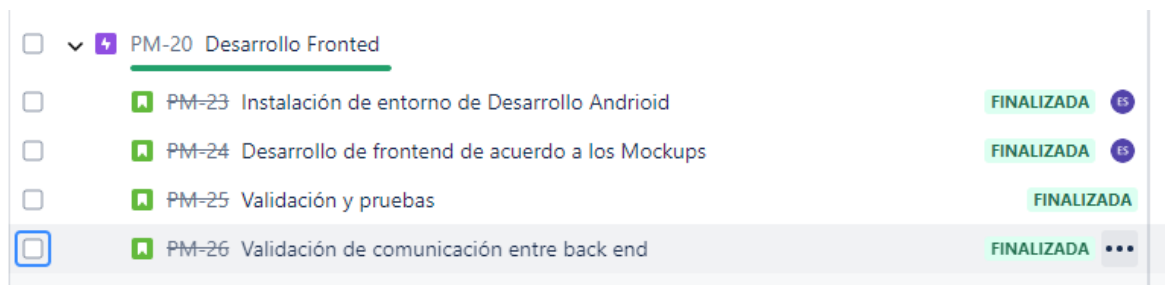


Figura 10: sprint 5 desarrollo frontend

La colaboración y retroalimentación continua es una característica que promueve una ceñida colaboración entre los miembros del equipo de desarrollo, así como con los usuarios finales. En este caso, la colaboración con la comunidad Yanacona es importante para validar la determinación e importancia de la información sobre las plantas medicinales. La metodología Scrum facilita la incorporación de la retroalimentación de la comunidad de manera continua, asegurando que la aplicación final no solo sea técnicamente sólida, sino también culturalmente apropiada y útil para los usuarios.

Para garantizar una organización y coordinación dentro del equipo de desarrollo, se implementaron varias prácticas clave. Se emplearon herramientas como Jira para organizar y seguir el progreso de las tareas, lo que ayudó a mantener la transparencia en el estado del proyecto y facilitó la asignación de responsabilidades. Esto aseguró que cada miembro del equipo conociera su rol y las prioridades del proyecto.

El equipo se reunía periódicamente para discutir el progreso, identificar posibles obstáculos y planificar las tareas del día. Estas reuniones fomentaron una comunicación clara y fluida entre los miembros, lo que permitió resolver problemas rápidamente y asegurarse de que todos estuvieran alineados con los objetivos del sprint.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 61 de 134

Cada ciclo de desarrollo se organizó en sprints de corta duración, donde se definieron objetivos claros y alcanzables. Al final de cada sprint, se entregaban versiones funcionales de la aplicación para su evaluación y mejora, lo que permitió avanzar de manera incremental. Esta combinación de herramientas y prácticas colaborativas fue fundamental para el éxito del proyecto, asegurando una implementación ágil y efectiva.

El objetivo principal del proyecto fue construir una aplicación software para organizar e identificar plantas que son usadas en la medicina tradicional de la comunidad del pueblo Yanacona del suroriente del departamento del Cauca, brindando información propia sobre las plantas y sobre sus usos, esta metodología se enfocó en la entrega continua de valor a los usuarios finales, asegurando que desde las primeras fases del desarrollo, la aplicación comience a ofrecer funcionalidades útiles, que fueron mejoradas y ampliadas en las iteraciones sucesivas.

4.3. Implementación de la Metodología

4.3.1. Estructura del Equipo

El equipo de desarrollo para este proyecto se organizó en tres roles principales, de acuerdo con la metodología Scrum:

Erwin Suárez como Scrum Master fue responsable de guiar al equipo en la implementación de Scrum, asegurando que se siguieran las prácticas y principios, facilitó las reuniones, ayudó a resolver impedimentos y se encargó de mantener el enfoque del equipo en los objetivos del sprint.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 62 de 134

Jineth manzano como Product Owner representó la voz de los usuarios finales y la comunidad Yanacona, asegurando que los requisitos y necesidades de estos stakeholder se reflejaran en el desarrollo de la aplicación. Este rol fue determinante para la priorización de las historias de usuario y para garantizar que el desarrollo del producto estuviera alineado con las expectativas de la comunidad y los objetivos del proyecto.

Development Team estuvo compuesto por un desarrollador Erwin solarte y una investigadora etnobotánica Jineth manzano. El desarrollador se encargó de la creación de la aplicación móvil, incluyendo la funcionalidad de reconocimiento de plantas y la integración del modelo de clasificación de plantas medicinales. La investigadora proporciono los contenidos detallados sobre las plantas, asegurando que la información sobre sus usos, beneficios y preparaciones fuera precisa y relevante.

4.3.2. Reuniones scrum

Durante el proyecto, se realizaron varias reuniones para garantizar la comunicación efectiva y el progreso constante:

Al inicio de cada sprint, se realizó una reunión de planificación en la que se discutieron los objetivos del sprint, se seleccionaron las historias de usuario a trabajar y se definieron las tareas específicas a realizar.

Los Daily Standup llevaron a cabo todos los días, permitiendo que cada miembro del equipo compartiera su progreso, los obstáculos encontrados y los planes para el día siguiente. Esto ayudó a mantener la transparencia y permitió la resolución rápida de problemas.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 63 de 134

Los Sprint Review, se realizó una revisión al final de cada sprint donde se presentó el trabajo completado de cada uno de ellos para así realizar un feedback. Esto fue fundamental para asegurar que el desarrollo del proyecto se mantuviera alineado con las expectativas de la comunidad Yanacona.

4.4. Ciclo de vida del desarrollo

4.4.1. Planificación

- Creación de las preguntas para médicos tradicionales del pueblo Yanaconas (Especificado en el capítulo 3).
- Creación de encuestas de preguntas para los comuneros del pueblo Yanacona (Especificado en el capítulo 3).
- Recolección y análisis de los datos obtenidos por las encuestas a los comuneros y las entrevistas a los médicos tradicionales (Especificado en el capítulo 3).
- Análisis y levantamiento de requerimientos.

4.4.2. Requerimientos funcionales

Se presentan los requerimientos funcionales de la aplicación móvil, los cuales incluyen las principales acciones que los usuarios podrán realizar, tales como botones para la captura o selección de imágenes, navegar en el menú, la visualización de información sobre sus usos y beneficios, y la navegación entre las distintas interfaces. Estos requerimientos son clave para garantizar que el sistema cumpla con los objetivos planteados y ofrezca una experiencia de usuario eficiente y efectiva.

Tabla 4:
 Requerimientos funcionales

DESARROLLO DE APLICACIÓN MOVIL						
REQUISITOS FUNCIONALES						
Nombre de la aplicación:						
Aplicación móvil para la organización e identificación de plantas medicinales del Pueblo Yanacona del suroriente del departamento del Cauca						
ID. Requisito	REQUE RIMIENT O	DESCRIPCION	PRIO RIDA D	COMPORTA MIENTO ESPERADO	ENTRADA	SALIDAS
RF-01	Botón cámara	Activa la cámara del dispositivo para capturar una imagen de la planta que se desea identificar Dirige al usuario a un	ALTA	la aplicación debe permitir la activación de la cámara del dispositivo móvil	fotografía	predicción
RF-02	Botón "Menú Principal	menú con las opciones "Plantas" y "Enfermedades".	ALTA	la aplicación debe permitir ir al menú de la app	menú: plantas, enfermedades	botones plantas, enfermedades
RF-03	Botón galería	La aplicación accede a la galería del dispositivo para seleccionar una foto de la planta.	ALTA	la aplicación debe permitir el acceso a la galería del dispositivo móvil y seleccionar la imagen a clasificar	galería del móvil	selección de imagen a clasificar
RF-04	Procesa miento de Imagen	La imagen capturada se envía al backend, donde la CNN procesa la imagen para	ALTA	la app debe permitir el envío de la imagen capturada o seleccionada	imagen.	clasificaci ón

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 65 de 134

RF-05	Resultado del Reconocimiento:	identificar la planta. La aplicación muestra el nombre de la planta identificada, junto con una opción para ver más información sobre la misma.	ALTA	al modelo de clasificación. la app debe mostrar el nombre de la planta clasificada.	planta clasificada	imagen, recetas y usos
RF-06	Opción "Plantas"	Muestra una lista de plantas disponibles en la base de datos, cada una con una imagen en miniatura.	ALTA	la app debe permitir ver la opción de plantas en la interfase	menú plantas	imagen y nombre de las plantas que se encuentran en la app
RF-07	Opción "Enfermedades"	Muestra una lista de enfermedades que pueden ser tratadas con plantas medicinales.	ALTA	la app debe permitir ver la opción de enfermedades en la interfase	menú enfermedades	nombre de las enfermedades a tratar con plantas medicinales de la app
RF-08	botón de Usos y recetas	Presenta una descripción detallada de los usos medicinales de la planta	ALTA	la app debe permitir ver los usos y recetas de las plantas medicinales en la app	botón recetas.	descripción detallada de la receta y usos.
RF-09	Búsqueda Avanzada:	Permite buscar plantas o enfermedades por nombre, usos o síntomas.	ALTA	la app debe permitir la búsqueda de plantas	consulta	imágenes, usos y recetas.

4.4.3. Requerimientos no funcionales

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 66 de 134

Se detallan los aspectos relacionados con el rendimiento, la usabilidad, tiempo de respuesta, la disponibilidad, la escalabilidad, compatibilidad de la aplicación móvil. Estos requerimientos son fundamentales para asegurar que la aplicación funcione de manera rápida, segura y confiable, brindando una experiencia fluida y satisfactoria, incluso bajo diferentes condiciones de uso y carga. Además, garantizan que el sistema sea fácilmente actualizable y adaptable a futuros cambios o mejoras.

Tabla 5:

Requerimientos no funcionales

ID. Requisito	REQUERIMIENTO	DESCRIPCION	PRIORIDAD
RNF-01	Usabilidad	La interfaz debe ser fácil de usar, con botones claramente etiquetados y flujos de usuario lógicos. Los usuarios deben poder navegar por la aplicación sin necesidad de un manual.	ALTA
RNF-02	Tiempo de Respuesta	La aplicación debe tener tiempos de respuesta rápidos al procesar solicitudes de reconocimiento de imágenes y al cargar información sobre las plantas medicinales, para proporcionar una experiencia de usuario fluida y sin demoras.	ALTA
RNF-03	Disponibilidad	La aplicación debe estar disponible para su uso en cualquier momento y lugar, incluso en áreas con conectividad limitada a internet.	MEDIA
RNF-04	Seguridad de los Datos	Se debe garantizar la seguridad de los datos del usuario, implementando medidas de cifrado y protección contra accesos no autorizados para proteger la información personal y médica almacenada en la aplicación.	MEDIA
RNF-05	Escalabilidad	La aplicación debe ser capaz de manejar un aumento en el número de usuarios y en la cantidad de datos almacenados a medida que crece su popularidad y su base de usuarios.	ALTA

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 67 de 134

RNF-06	Compatibilidad:	La aplicación debe ser compatible con una amplia gama de dispositivos móviles y sistemas operativos para llegar a la mayor cantidad posible de usuarios	ALTA
RNF-07	Rendimiento	La aplicación debe tener un rendimiento óptimo en términos de velocidad y eficiencia, incluso cuando se ejecuta en dispositivos móviles con recursos limitados como memoria y capacidad de procesamiento.	MEDIA
RNF-08	Adaptabilidad	La aplicación debe ser adaptable a diferentes tamaños de pantalla y resoluciones, garantizando una experiencia de usuario consistente en dispositivos móviles de diferentes marcas y modelos.	BAJA
RNF-09	Mantenimiento	Se debe proporcionar un plan de mantenimiento regular para garantizar que la aplicación siga funcionando correctamente, corrigiendo errores, actualizando contenido y realizando mejoras según sea necesario.	ALTA
RNF-10	Accesibilidad	La aplicación debe ser accesible para usuarios con discapacidades visuales, motoras o cognitivas, cumpliendo con las pautas de accesibilidad establecidas, como las pautas de la WCAG (Web Content Accessibility Guidelines).	MEDIA
RNF-11	Compatibilidad con Versiones Anteriores	La aplicación debe ser diseñada para ser compatible con una amplia gama de dispositivos móviles y sistemas operativos, asegurando que los usuarios con dispositivos más antiguos puedan seguir disfrutando de sus funcionalidades. La compatibilidad se extiende desde Android 5.0 (Lollipop) hasta la última versión disponible, Android 14 (Upside Down Cake), garantizando un acceso inclusivo y eficiente para todos los usuarios, independientemente de la antigüedad de sus dispositivos.	ALTA
RNF-12	Tolerancia a Fallos	La aplicación debe ser capaz de recuperarse automáticamente de posibles fallos del sistema, minimizando así cualquier interrupción en la experiencia del usuario.	BAJA

4.4.4. Toma de fotografías, organización de ellas por carpetas

- Realizar la toma de fotografías de plantas según el resultado de las encuestas, organizar por carpetas las fotografías recopiladas.
- Se crean cuatro carpetas con los nombres de las plantas a trabajar en el modelo de clasificación eucalipto, manzanilla, hierbabuena y sábila.

Creación de dataset de plantas medicinales:

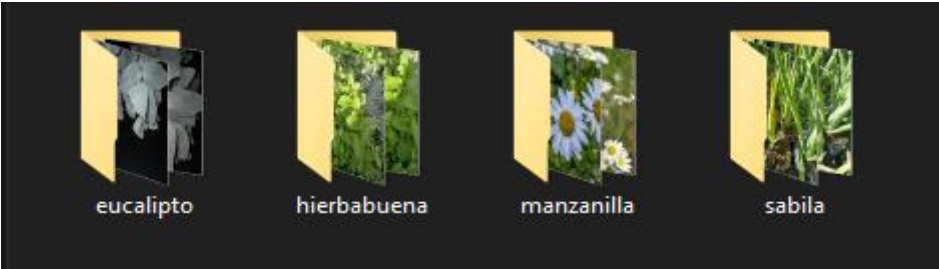


Figura 11: Carpeta de data set plantas

Acá se muestra el contenido de cada carpeta:

Eucalipto:

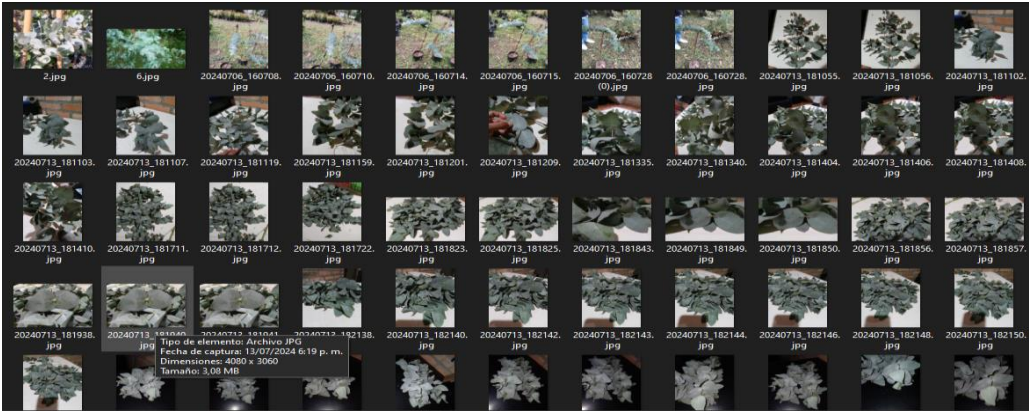


Figura 12: Carpeta Eucalipto

Hierbabuena:

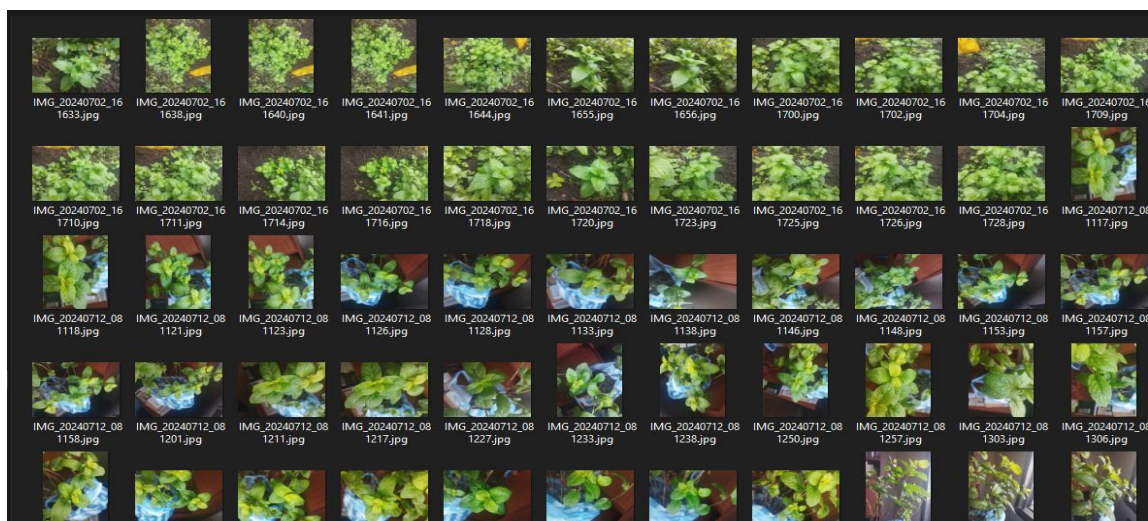


Figura 13: Carpeta Hierbabuena

Manzanilla:

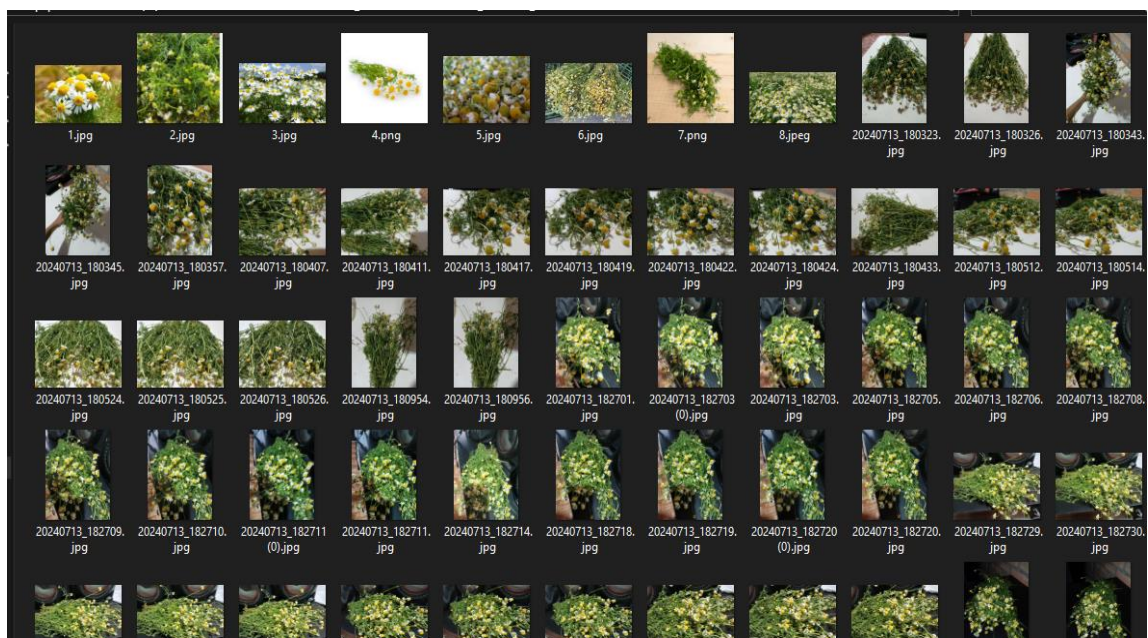


Figura 14: Carpeta manzanilla

Sábila:

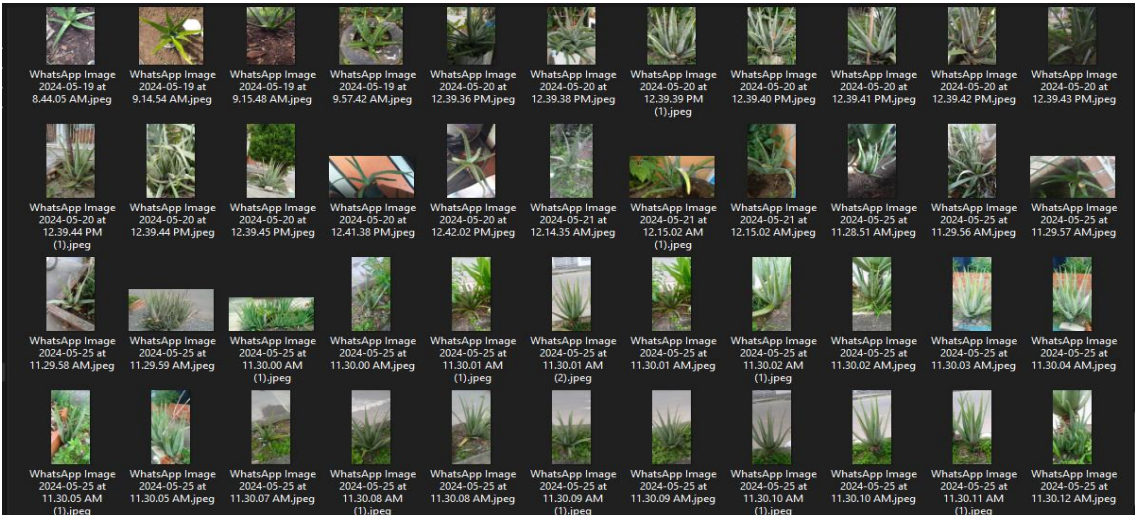


Figura 15: Carpeta sábila

4.4.5. Diseño

4.4.5.1. Creación de Mockups.

Durante el desarrollo de la aplicación, fue fundamental visualizar y validar el diseño de la interfaz antes de su ejecución definitiva. Para ello, se realizaron prototipos de media fidelidad (Mid-Fidelity), que permiten representar de manera más detallada la estructura y funcionalidad de la aplicación sin llegar a los niveles de precisión visual de los prototipos de alta fidelidad.

Los mockups fueron creados con el fin de simular cómo interactuarían los usuarios con la aplicación, brindando una visión clara de la disposición de los elementos, la navegación entre pantallas, y la interacción básica sin comprometer demasiado tiempo en los detalles gráficos. Esta etapa fue clave para recibir retroalimentación temprana y asegurar que el flujo de la aplicación correspondiera a las expectativas del usuario.

A continuación, se presentan los mockups que muestran el diseño de la interfaz y las principales funcionalidades del sistema.

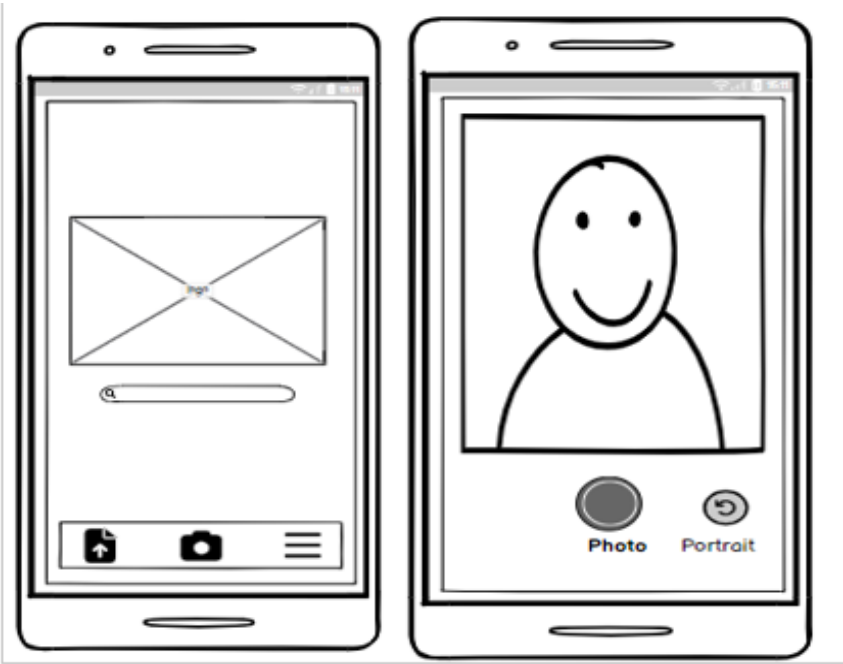


Figura 16: Mockups interfaz 1 y 2

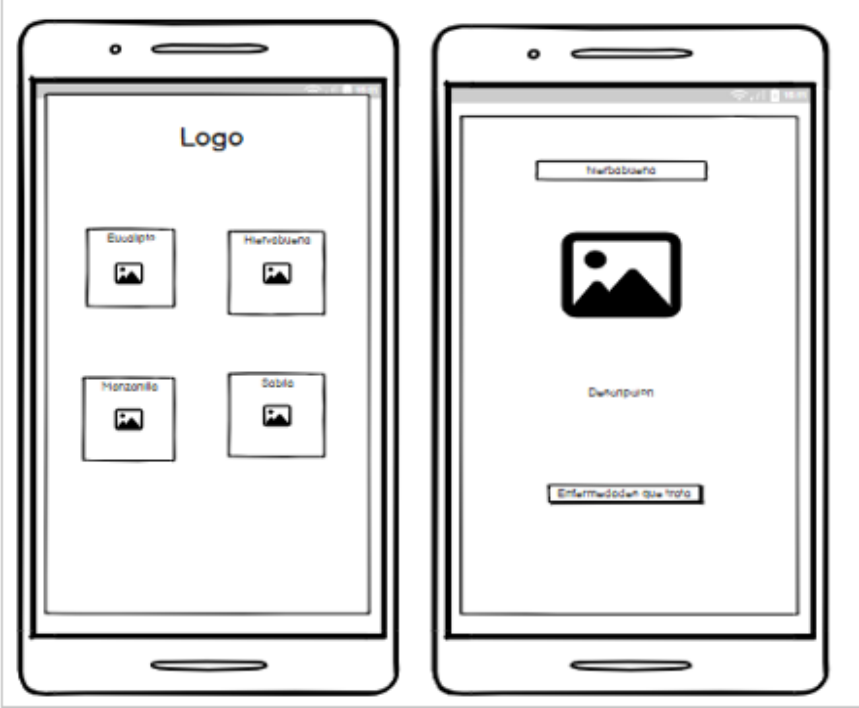


Figura 17: Mockups interfaz 3 y 4



Figura 18: Mockups interfaz 7 ,8 y 9

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 73 de 134

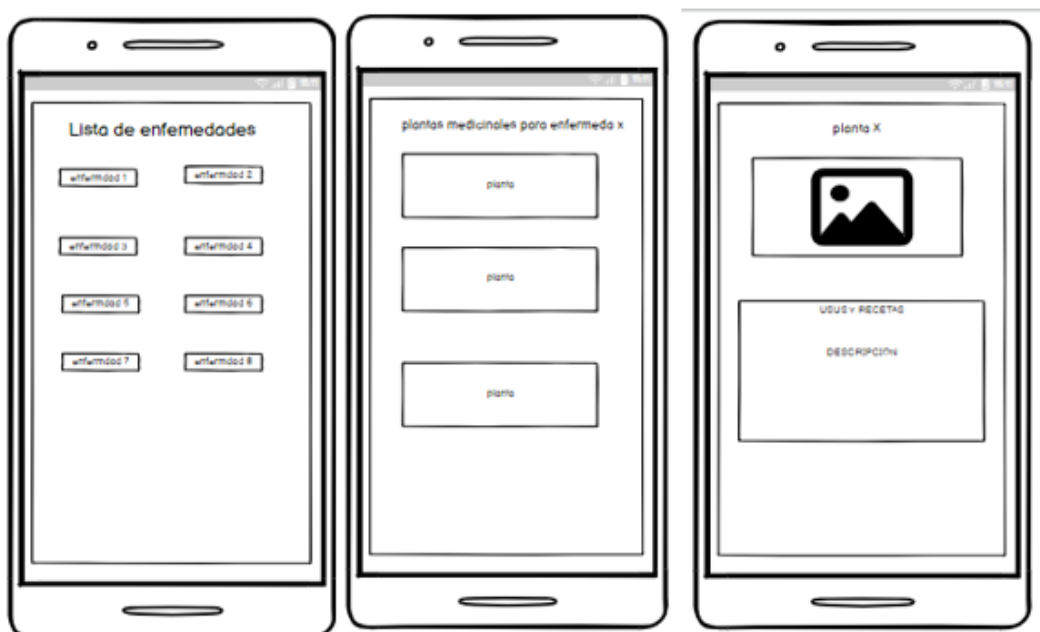


Figura 19: Mockups interfaz 10,11 y 12

4.4.6. Historias de usuario

Se presentan las historias de usuario, planteadas a partir de los requerimientos funcionales identificados para la aplicación móvil. Cada historia de usuario describe una interacción del sistema desde la perspectiva de los usuarios, como la funcionalidad de cada botón, la obtención de información detallada sobre los usos, beneficios y características de las plantas que contiene la aplicación y la capacidad de navegar entre las diferentes secciones de la aplicación.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 74 de 134

Tabla 6:

Historia de usuario 001

HISTORIA DE USUARIO	FECHA
Historia De Usuario No: 001	
Nombre: Botón Cámara	Prioridad: Alta
Descripción: el usuario al iniciar la aplicación observará un botón el cual llevará a la cámara del móvil.	
Criterios de Aceptación: 1. El usuario, debe acceder a la cámara desde el botón cámara que está ubicado en la interfaz principal. 2. El sistema debe permitir a la cámara capturar la fotografía. 3. La cámara debe mostrar las opciones de aprobar fotografía o eliminar y tomar una nueva	

Tabla 7:

Historia de usuario 002

HISTORIA DE USUARIO	FECHA
Historia De Usuario No: 002	
Nombre: botón galería	Prioridad: Alta
Descripción: el usuario al iniciar la aplicación observará un botón el cual llevará a la galería del móvil	
Criterios de Aceptación: 1. El usuario, debe acceder a la galería desde el botón galería que está ubicado en la interfaz principal. 2. El sistema debe permitir el acceso a la galería. 3. El sistema debe permitir seleccionar la imagen a clasificar.	

Tabla 8:

Historia de usuario 003

HISTORIA DE USUARIO	FECHA
Historia De Usuario No: 003	
Nombre: botón menú	Prioridad: Alta

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 75 de 134

Descripción: el usuario al iniciar la aplicación observará un botón menú que permite el acceso a la navegación dentro de la aplicación.

Criterios de Aceptación:

1. El usuario, debe acceder al menú de la app desde el botón menú que está ubicado en la interfaz principal.
2. El sistema debe permitir el acceso al menú de la app
3. El usuario debe seleccionar en el menú las opciones plantas o enfermedades dependiendo de su consulta.

Tabla 9:

Historia de usuario 004

HISTORIA DE USUARIO	FECHA
Historia De Usuario No: 004	
Nombre: botón enviar	Prioridad: Alta
Descripción: el usuario al seleccionar la planta desde la galería o la captura de ella desde la cámara el sistema activa un nuevo botón en la interfaz principal el cual permite enviar la imagen al servidor para la clasificación de ella.	
Criterios de Aceptación:	
1. El usuario, debe acceder a la cámara o galería para seleccionar la imagen a clasificar. 2. El sistema debe permitir después de la selección de la imagen activar el botón enviar. 3. El usuario debe presionar el botón enviar para el envío de la imagen seleccionada al servidor y la clasifique.	

Tabla 10:

Historia de usuario 005

HISTORIA DE USUARIO	FECHA
Historia De Usuario No: 005	
Nombre: opción plantas	Prioridad: Alta
Descripción: el usuario al ingresar al menú tendrá las opciones de ver las plantas medicinales que contiene la aplicación y navegar en ellas para ver sus usos y recetas	

Criterios de Aceptación: 1. El usuario, debe acceder a la opción de plantas medicinales. 2. El sistema debe permitir ver las plantas con las que cuenta la aplicación. 3. El sistema debe permitir el acceso a las plantas para ver su información detalladamente. 4. El sistema debe permitir ver la receta y uso de la planta que se seleccione.
--

Tabla 11:
 Historia de usuario 006

HISTORIA DE USUARIO	FECHA
Historia De Usuario No: 006	
Nombre: opción enfermedades	Prioridad: Alta
Descripción: el usuario al ingresar al menú tendrá la opción de ver las diferentes enfermedades que se tratan con plantas medicinales	
Criterios de Aceptación: 1. El usuario, debe acceder a la opción de enfermedades. 2. El sistema debe permitir ver el menú de las diferentes enfermedades que se tratan con las plantas medicinales con las que cuenta la aplicación. 3. El sistema debe permitir el acceso a las enfermedades para ver su información detalladamente. 4. El sistema debe permitir ver las plantas que trata la enfermedad seleccionada. 5. El sistema debe permitir ver las rectas y uso de la planta para tratar la enfermedad seleccionada.	

4.4.7. Diagrama de secuencia

El diagrama de secuencia representa el flujo de interacción entre el usuario, la aplicación móvil, y el servidor, mostrando cómo se lleva a cabo el proceso de reconocimiento de plantas medicinales utilizando una imagen y cómo el usuario puede acceder a la información de las plantas de manera manual, sin necesidad de una imagen.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 77 de 134

El usuario puede capturar una imagen utilizando la cámara del dispositivo o seleccionar una imagen de la galería, la aplicación envía la imagen a un servidor mediante una API para su procesamiento.

La imagen es enviada al servidor a través de una API REST, en el servidor, el modelo previamente entrenado procesa la imagen para identificar la planta medicinal

El servidor utiliza el modelo de clasificación de imágenes basado en aprendizaje automático para analizar la imagen y determinar a qué planta corresponde, una vez identificada, el servidor envía la respuesta con el nombre de la planta clasificada de vuelta a la aplicación móvil.

La aplicación recibe la respuesta del servidor, mostrando al usuario una pantalla con la información detallada de la planta identificada, incluyendo nombre de la planta, características (como propiedades medicinales y descripción), enfermedades que trata, recetas para preparar remedios con la planta, navegación manual mediante el botón de menú:

Alternativamente, el usuario puede prescindir del uso de imágenes y acceder manualmente a la información a través de un botón de Menú permite explorar las plantas medicinales disponibles o navegar por una lista de enfermedades.

En el caso de seleccionar una enfermedad, la aplicación mostrará las plantas medicinales que sirven para tratarla, junto con las recetas correspondientes. Si el usuario selecciona una planta, se desplegará la información completa de dicha planta, incluyendo sus características, los padecimientos que puede tratar, y las recetas relacionadas.

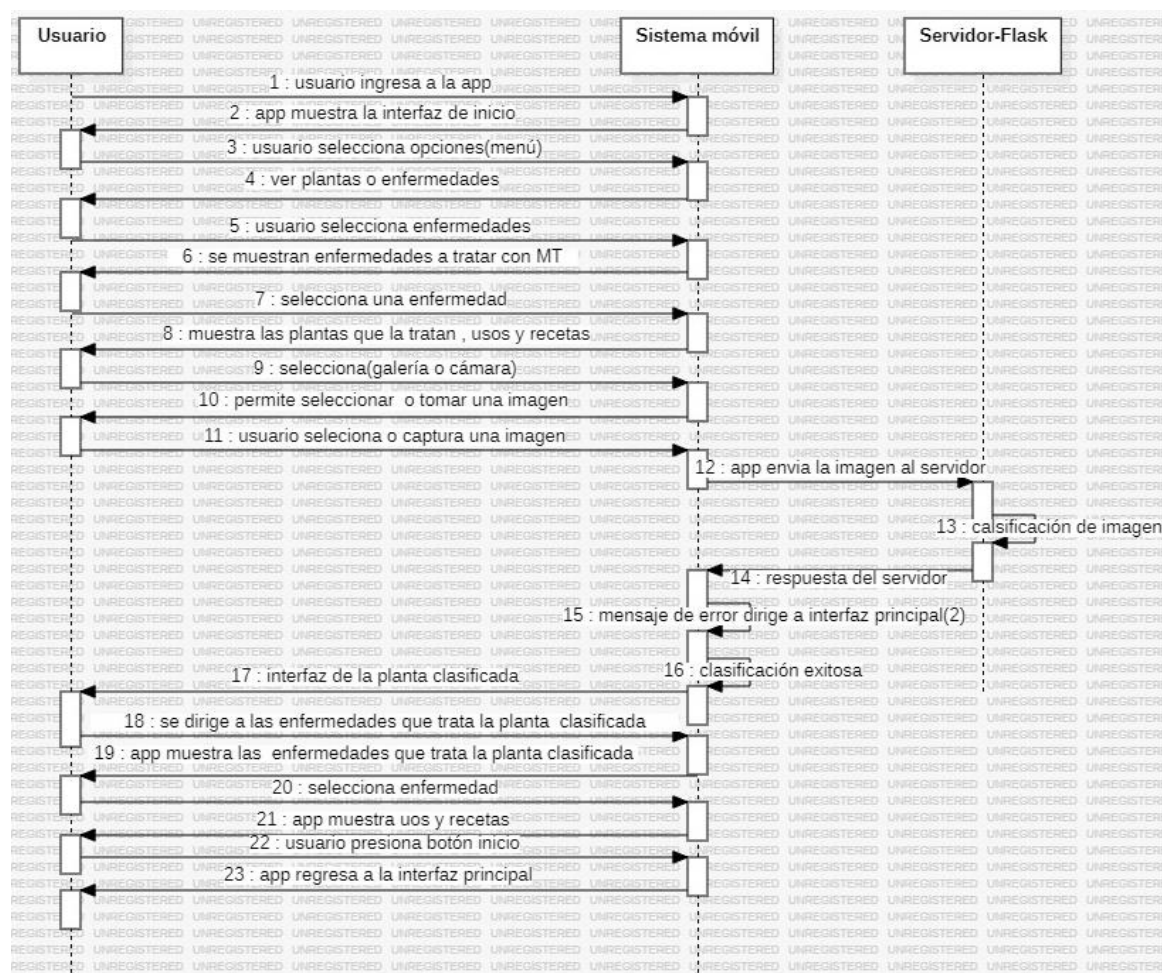


Figura 20: Diagrama de secuencia

4.4.8. Diagrama de flujo

Este diagrama de flujo representa el funcionamiento completo de la aplicación móvil, comenzando con las interacciones que el usuario puede realizar a través de los botones Cámara, Galería y Menú.

Cuando el usuario interactúe con los botones de Cámara o Galería, selecciona o captura una imagen de una planta medicinal. Esta imagen es enviada mediante una API al modelo de clasificación, el cual ha sido previamente entrenado para reconocer las plantas. Tras el proceso

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 79 de 134

de clasificación, la aplicación recibe el resultado con el nombre de la planta identificada y redirige al usuario a una interfaz específica de dicha planta. En esta interfaz, el usuario podrá acceder a información detallada de la planta, como su descripción, las enfermedades que trata, las recetas medicinales relacionadas y su imagen correspondiente.

Adicionalmente, si el usuario decide interactuar mediante el botón de Menú, tiene dos opciones principales de explorar las plantas medicinales o revisar las enfermedades tratadas por las plantas.

Al seleccionar la opción de plantas, se muestra un listado completo de las plantas disponibles en la aplicación. El usuario puede seleccionar cualquier planta para visualizar su descripción, imagen, las enfermedades que trata y las recetas para su preparación y uso medicinal.

Por otro lado, si el usuario opta por revisar el listado de enfermedades, podrá observar una lista de enfermedades las cuales al seleccionar una de ellas en particular, la aplicación mostrará la planta medicinal que se utiliza para tratarla, junto con su imagen, una descripción detallada de sus características y las recetas necesarias para su tratamiento.

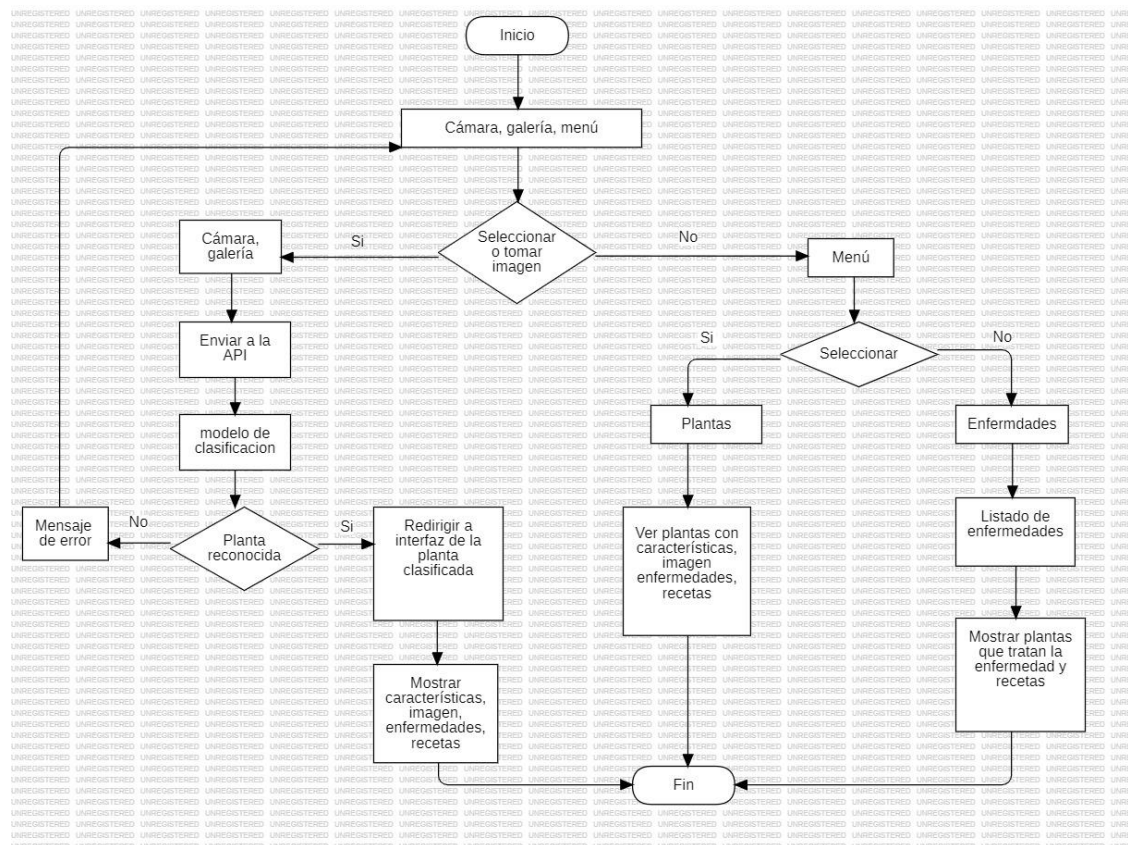


Figura 21: Diagrama de flujo

4.4.9. Diagrama de caso de uso

Este diagrama de caso de uso muestra cómo el usuario puede interactuar con la aplicación para acceder a la información de las plantas de dos maneras: seleccionando una planta manualmente o a través de una clasificación mediante imágenes.

Cuenta con dos actores el primero es el usuario, que se representado por la figura del lado izquierdo, este es el actor principal que interactúa con la aplicación, accediendo a la funcionalidad de consulta de plantas y enfermedades, y enviando imágenes para su clasificación.

El segundo actor es el servidor Flask que se Representa por la figura del lado derecho, este es el segundo actor que recibe la imagen enviada por el usuario, ejecuta el modelo de clasificación y devuelve la información de la planta reconocida a la aplicación móvil.

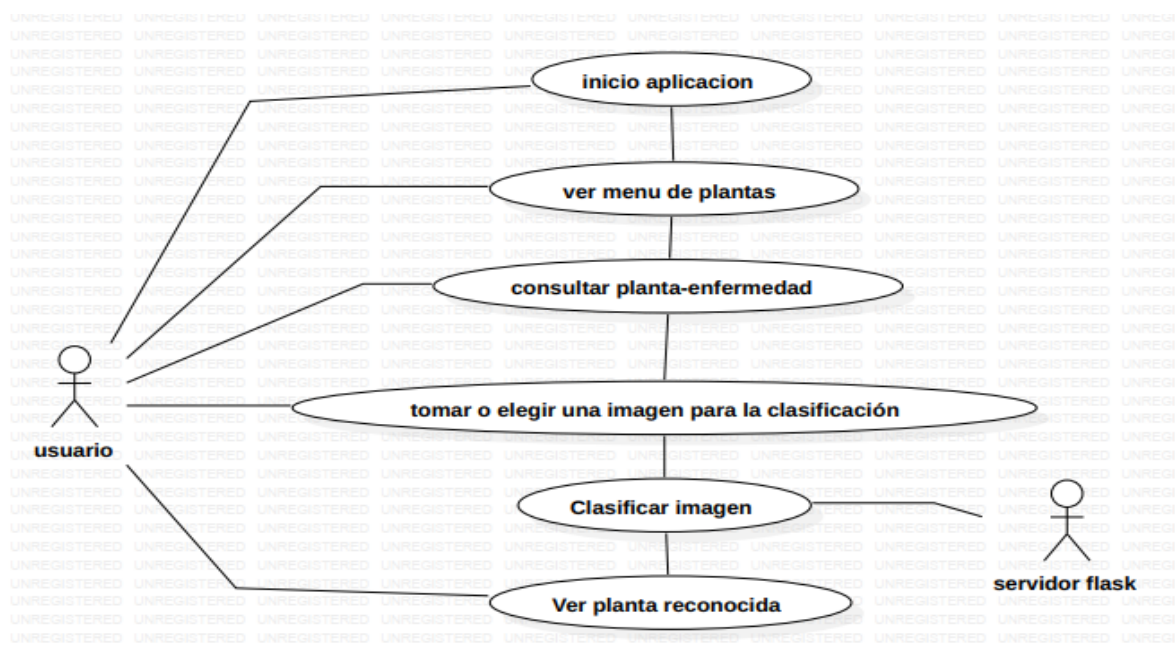


Figura 22: Diagrama de caso de uso

4.5. Diseño de arquitectura.

La arquitectura de la aplicación está basada en un modelo cliente-servidor, donde la aplicación móvil en Android actúa como el cliente, y un servidor Flask procesará las imágenes para clasificar las plantas medicinales. Este enfoque permite delegar tareas intensivas de procesamiento al servidor, manteniendo la aplicación móvil ligera y eficiente para el usuario final.

4.5.1. Componentes de la arquitectura de la aplicación móvil

En la app el cliente es (Aplicación Móvil en Android) la cual presenta una interfaz amigable para el usuario, donde este puede tomar o seleccionar una imagen de una planta usando la cámara o la galería. También permite la navegación manual a través de un menú para ver la lista de plantas o enfermedades tratadas por ellas, proporcionando información como descripciones, recetas y características de las plantas.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 82 de 134

Una vez que el usuario selecciona o captura una imagen, esta es enviada al servidor a través de una API. La aplicación móvil actúa como cliente al realizar solicitudes HTTP al servidor Flask, luego de recibir la respuesta del servidor, la aplicación navega hacia la interfaz de la planta clasificada, donde se muestra la información asociada como características, imagen, enfermedades tratadas y recetas.

El Servidor Flask recibe la imagen enviada desde la aplicación móvil y a través de un modelo de inteligencia artificial entrenado (como una red neuronal de clasificación de imágenes), el servidor identifica la planta medicinal representada en la imagen y una vez que la imagen es procesada, el servidor devuelve la clasificación al cliente junto con los datos asociados a la planta identificada.

4.5.2. Arquitectura del Modelo de Clasificación VGG16

Este modelo de clasificación se utilizó para entrenar el backend de nuestra aplicación. Al ser un modelo preentrenado, observamos una oportunidad valiosa para agilizar tanto el proceso de desarrollo como el de entrenamiento y reconocimiento de imágenes en el proyecto. La utilización de un modelo preentrenado como VGG16 nos permitió aprovechar los pesos y características previamente aprendidas, ajustándolo a nuestras necesidades específicas de clasificación de plantas medicinales. Esto no solo aceleró el desarrollo, sino que también mejoró la precisión del reconocimiento en comparación con entrenar un modelo desde cero.

A continuación, se detalla su arquitectura, junto con una imagen que ilustra su estructura:

El modelo VGG16 se compone de bloques de capas convolucionales, que tienen 2 o 3 convoluciones. Las capas convolucionales de la red VGG16 son en su mayor parte filtros de tamaño 3×3 con un tamaño de paso de 1, y las capas de agrupación son de 2 × 2 con un tamaño

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 83 de 134

de paso de 2, esta configuración es estándar y se aplica a cualquier implementación básica del modelo. [47]

Una pila de capas convolucionales es seguida por tres capas totalmente conectadas: las dos primeras tienen 4096 neuronas cada una, la tercera realiza una clasificación, la cual consta de 1000 neuronas y, por lo tanto, contiene 1000 canales (uno para cada clase). La capa final es la capa softmax. La configuración de las capas totalmente conectadas es la misma en todas las redes [47].

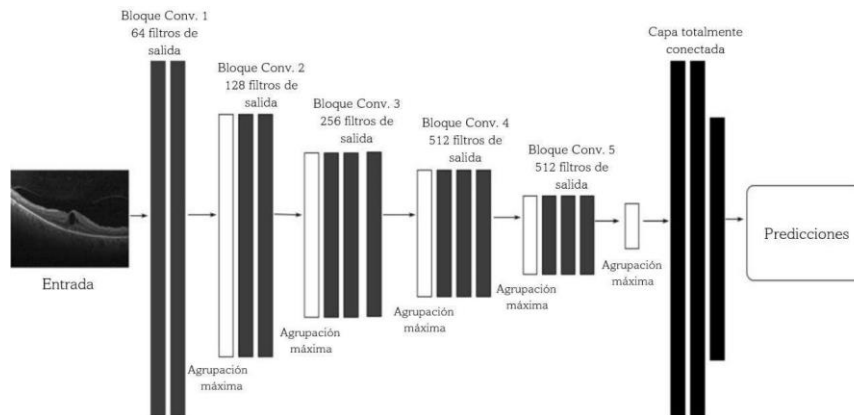


Figura 23: Arquitectura modelo vgg16 [47].

4.6. Implementación

La implementación de la aplicación se llevó a cabo dividiendo el trabajo en dos partes, el backend y el frontend, cada una con un rol específico y complementario en la funcionalidad general del sistema.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 84 de 134

Iniciamos por el backend que fue desarrollado para manejar el procesamiento de datos, incluyendo la implementación de un modelo de red neuronal para la clasificación de imágenes. Utilizando Flask como marco principal, se creó una API capaz de recibir imágenes desde la aplicación móvil, procesarlas a través de un modelo preentrenado (como VGG16), y devolver una clasificación precisa de las plantas medicinales. Esta arquitectura robusta asegura que los usuarios puedan obtener resultados rápidos y precisos basados en las imágenes enviadas.

Por el lado del frontend, se desarrolló en Android, es la interfaz que interactúa directamente con el usuario. A través de una experiencia intuitiva y amigable, la aplicación permite capturar o seleccionar imágenes, consultar información sobre plantas medicinales, visualizar sus características, y acceder a recetas y tratamientos. La integración con el backend se realiza mediante solicitudes HTTP, lo que garantiza una comunicación fluida entre la aplicación móvil y el servidor.

Las dos partes del sistema trabajan en conjunto para brindar una solución completa, desde la captura de imágenes hasta la clasificación y presentación de la información, garantizando que el usuario tenga acceso a un conocimiento profundo de las plantas medicinales de manera rápida y sencilla.

4.6.1. Desarrollo backend

El primer paso en el desarrollo backend consistió en la instalación de los entornos de desarrollo necesarios, como Visual Studio Code, el lenguaje de programación Python, y el entorno de Anaconda. Una vez configurados estos entornos, se procedió a descargar y configurar las librerías requeridas: TensorFlow, Keras, NumPy, y otras dependencias dentro del entorno de Anaconda.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 85 de 134

El desarrollo del modelo de clasificación se estructuró en tres archivos principales:

Un archivo de entrenamiento responsable de entrenar el modelo de clasificación con el conjunto de datos de plantas medicinales. A través de este proceso, el modelo aprende a reconocer las características de las plantas para poder realizar predicciones precisas en el futuro.

Un archivo de predicción encargado de tomar una imagen ingresada por el usuario, ya sea capturada o seleccionada, y pasarla al modelo para su clasificación. Al final del proceso, el sistema devuelve el nombre de la planta reconocida. En este proyecto, el modelo ha sido entrenado para reconocer cuatro plantas: sábila, hierbabuena, eucalipto y manzanilla.

Y por último el archivo del servidor que actúa como una API, con una principal función es recibir imágenes de plantas medicinales, procesarlas utilizando el modelo de red neuronal, y devolver la predicción de la planta identificada.

A continuación, se describen los fragmentos de código implementado en el desarrollo del modelo CNN (**ENTRENAMIENTO**)

Lo primero que se realiza es la codificación del archivo **ENTRENAMIENTO** el cual va a hacer el encargado de entrenar el modelo de red neuronal convolucional basado en VGG16 para la clasificación de imágenes en cuatro clases diferentes, utilizando técnicas de aumento de datos y regularización para mejorar la generalización y prevenir el sobreajuste. Al finalizar, se generan gráficos para evaluar el rendimiento del modelo y se guarda el modelo entrenado junto con sus pesos.

En la imagen se muestran las librerías que se usan para el desarrollo del archivo de entrenamiento del modelo de clasificación de imágenes.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 86 de 134

```
import os
os.environ['KMP_DUPLICATE_LIB_OK'] = 'True'
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras import optimizers
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dropout, Flatten, Dense, Activation
from tensorflow.keras.applications import VGG16
from tensorflow.keras.callbacks import EarlyStopping
from tensorflow.keras import backend as K
from tensorflow.keras.regularizers import l2
```

Figura 24: código importación de librerías

A continuación, fragmento de código para el desarrollo y creación de una neurona con lenguaje CNN para realizar el reconocimiento de imágenes (en el documento anexo descripción de línea por línea de código).

```
# Limpiar cualquier sesión de Keras existente
K.clear_session()
```

```
# Directorios para los datos de entrenamiento y validación
data_entrenamiento = 'Cnn_Medicina_Tradicional/data/entrenamiento'
data_validacion = 'Cnn_Medicina_Tradicional/data/validacion'
```

```
# Parámetros
n_epocas = 50
longitud, altura = 150, 150
tamaño_batch = 28
```

```
# Contar el número de imágenes en cada directorio
num_img_entrenamiento = sum([len(files) for r, d, files in os.walk(data_entrenamiento)])
num_img_validacion = sum([len(files) for r, d, files in os.walk(data_validacion)])
```

```
# Calcular steps_per_epoch y validation_steps
pasos_epoca = num_img_entrenamiento // tamaño_batch
pasos_validacion = num_img_validacion // tamaño_batch
```

```
nclases = 4
tasa_aprendizaje = 0.0005 #0001 es el principal
```

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 87 de 134

```

# Aumentación de datos para el entrenamiento
entrenamiento_datagen = ImageDataGenerator(
    rescale=1. / 255,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True,
    rotation_range=40,
    width_shift_range=0.2,
    height_shift_range=0.2,
    brightness_range=[0.8, 1.2]
)

# Generadores para los datos de entrenamiento y validación
entrenamiento_generador = entrenamiento_datagen.flow_from_directory(
    data_entrenamiento,
    target_size=(altura, longitud),
    batch_size=tamaño_batch,
    class_mode='categorical'
)

validacion_generador = test_datagen.flow_from_directory(
    data_validacion,
    target_size=(altura, longitud),
    batch_size=tamaño_batch,
    class_mode='categorical'
)

# Cargar el modelo preentrenado VGG16 + capas superiores
base_model = VGG16(weights='imagenet', include_top=False, input_shape=(altura, longitud, 3))

# Congelar las capas del modelo base VGG16
for layer in base_model.layers:
    layer.trainable = False

# Crear el modelo superior
top_model = Sequential()
top_model.add(Flatten(input_shape=base_model.output_shape[1:]))
top_model.add(Dense(512, activation='relu', kernel_regularizer=l2(0.001)))
top_model.add(Dropout(0.5))
top_model.add(Dense(nclases, activation='softmax'))

# Combinar el modelo base y el modelo superior
model = Sequential()
model.add(base_model)
model.add(top_model)

```

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 88 de 134

```
# Compilar el modelo
model.compile(
    loss='categorical_crossentropy',
    optimizer=optimizers.Adam(learning_rate=tasa_aprendizaje),
    metrics=['accuracy']
)

# Early stopping para prevenir el sobreajuste
early_stopping = EarlyStopping(monitor='val_loss', patience=10, restore_best_weights=True)

# Entrenar el modelo y almacenar el historial de entrenamiento
history = model.fit(
    entrenamiento_generador,
    steps_per_epoch=pasos_epoca,
    epochs=n_epocas,
    validation_data=validacion_generador,
    validation_steps=pasos_validacion,
    callbacks=[early_stopping]
)

# Guardar el modelo y los pesos
target_dir = './modelo/'
if not os.path.exists(target_dir):
    os.mkdir(target_dir)
model.save(os.path.join(target_dir, 'modeloplantas.keras'))
model.save_weights(os.path.join(target_dir, 'pesos.weights.h5'))
```

Figura 25: Código modelo entrenamiento

El siguiente fragmento de código es la Implementación para mostrar las gráficas de perdida y precisión del modelo, graficas que nos permiten observar el comportamiento de aprendizaje del modelo de clasificación después de realizar el proceso de entrenamiento.

```
# Importar matplotlib
import matplotlib.pyplot as plt

# Graficar la precisión y la pérdida del modelo
plt.figure(figsize=(12, 4))

# Precisión
plt.subplot(1, 2, 1)
plt.plot(history.history['accuracy'], label='Precisión de entrenamiento')
plt.plot(history.history['val_accuracy'], label='Precisión de validación')
plt.title('Precisión del modelo')
plt.xlabel('Épocas')
plt.ylabel('Precisión')
plt.legend()

# Pérdida
plt.subplot(1, 2, 2)
plt.plot(history.history['loss'], label='Pérdida de entrenamiento')
plt.plot(history.history['val_loss'], label='Pérdida de validación')
plt.title('Pérdida del modelo')
plt.xlabel('Épocas')
plt.ylabel('Pérdida')
plt.legend()

# Ajustar el layout y mostrar las gráficas
plt.tight_layout()
plt.show()
```

Figura 26: Código modelo entrenamiento.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 89 de 134

A continuación, el fragmento de código del archivo (**PREDICCIÓN**) el cual a la hora de ejecutarlo carga el modelo previamente entrenado, para luego realiza la predicción de la imagen dada.

```
import numpy as np
from tensorflow.keras.preprocessing.image import load_img, img_to_array
from tensorflow.keras.models import load_model
longitud, altura = 150, 150
modelo = './modelo/modeloplantillas.keras'
pesos_modelo = './modelo/pesos.weights.h5'
cnn = load_model(modelo)
cnn.load_weights(pesos_modelo)

def predict(file):
    x = load_img(file, target_size=(longitud, altura))
    x = img_to_array(x)
    x = np.expand_dims(x, axis=0)
    array = cnn.predict(x)
    result = array[0]
    answer = np.argmax(result)
    if answer == 0:
        print("pred: Eucalipto ")
    elif answer == 1:
        print("pred: Hiervabuena")
    elif answer == 2:
        print("pred: Manzanilla")
    elif answer == 3:
        print("pred: Sabila")

    return answer

predict('Cnn_Medicina_Tradicional/sabila23.jpg')
```

Figura 27: Código predicción

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 90 de 134

A continuación, el desarrollo de un api con flask para comunicación con el frontend, fragmento de código del archivo (**SERVIDOR**)

```
def model_predict(img_path, model):
    # Cargar y preprocesar la imagen
    img = load_img(img_path, target_size=(width_shape, height_shape))
    x = img_to_array(img)
    x = np.expand_dims(x, axis=0) # Añadir la dimensión de batch
    x /= 255.0 # Normalizar si se hizo en el entrenamiento

    # Realizar la predicción
    preds = model.predict(x)
    return preds

@app.route('/', methods=['GET'])
def index():
    # Página principal
    return render_template('index.html')

@app.route('/predict', methods=['POST'])
def upload():
    if request.method == 'POST':
        # Obtiene el archivo del request
        f = request.files['file']

        # Graba el archivo en ./uploads
        basepath = os.path.dirname(__file__)
        upload_path = os.path.join(basepath, 'uploads')

        # Verifica si la carpeta uploads existe, si no, la crea
        if not os.path.exists(upload_path):
            os.makedirs(upload_path)
```

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 91 de 134

```

# Ruta completa del archivo
file_path = os.path.join(upload_path, secure_filename(f.filename))
f.save(file_path)

# Predicción
preds = model_predict(file_path, model)

# Obtener el nombre de la clase con la mayor probabilidad
class_idx = np.argmax(preds)
result = names[class_idx]

# Eliminar la imagen después de la predicción
os.remove(file_path)

# Enviar el resultado de la predicción como JSON
return jsonify({'prediction': result})

return None

if __name__ == '__main__':
    app.run(debug=False, threaded=False)

from flask import Flask, request, render_template, jsonify
import os
import numpy as np
from tensorflow.keras.preprocessing.image import load_img, img_to_array
from tensorflow.keras.models import load_model
from werkzeug.utils import secure_filename

# Parámetros del modelo
width_shape, height_shape = 150, 150
names = ['eucalipto', 'hiervabuena', 'manzanilla', 'sabila']

# Definimos una instancia de Flask
app = Flask(__name__)

# Paths del modelo preentrenado y los pesos
MODEL_PATH = './modelo/modeloplantas.keras'
WEIGHTS_PATH = './modelo/pesos.weights.h5'

# Cargamos el modelo preentrenado
model = load_model(MODEL_PATH)

# Cargamos los pesos si están en un archivo separado
model.load_weights(WEIGHTS_PATH)

print('Modelo y pesos cargados exitosamente. Verificar http://127.0.0.1:5000/)

```

Figura 28: Código servidor

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 92 de 134

El proceso de entrenamiento, validación y pruebas de los fragmentos de código anteriormente mencionados en el capítulo 5 se mostrarán los resultados de la ejecución de cada uno detalladamente.

4.7. Desarrollo frontend

El Desarrollo de frontend se realiza de acuerdo con los Mockups establecidos previamente, se desarrolló la aplicación móvil en Android Studio utilizando Kotlin como lenguaje, la aplicación buscando ofrecer una experiencia de navegación intuitiva con una interfaz principal que presenta tres botones clave, uno para acceder al menú donde podrás ver las opciones de plantas o enfermedades que la app contiene, otro para el acceso a la cámara y un último para acceder galería.

En la opción de plantas, los usuarios pueden visualizar imágenes, usos y recetas de las plantas disponibles dentro de la aplicación (sábila, manzanilla, hierbabuena y eucalipto) En la opción de enfermedades, se muestra cómo tratar cada una con plantas medicinales, incluyendo sus recetas. Además, al seleccionar una imagen desde la cámara o galería, se activa un botón el cual permite el envío al servidor backend, desarrollado en Flask, que contiene un modelo de clasificación de dichas plantas. Este modelo permite identificar la planta en la imagen y redirigir al usuario directamente a la interfaz correspondiente de la planta reconocida. La interacción entre la aplicación y el servidor Flask se realiza de forma eficiente, permitiendo un flujo dinámico y rápido.

En el desarrollo de la aplicación de medicina tradicional se utilizaron varios archivos Activity, cada uno con su correspondiente archivo XML. En los archivos Activity se codifica la lógica que permite la interacción entre las interfaces, gestionando la navegación y funcionalidad

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 93 de 134

de la aplicación. Por otro lado, en los archivos XML se define el diseño de la interfaz de usuario, especificando el contenido visual, como el diseño de la disposición de los elementos, los colores, las imágenes y los textos utilizados en la aplicación.

La siguiente imagen muestra la interfaz de trabajo de Android Studio. En la parte izquierda, se encuentra el panel del proyecto, que muestra todos los archivos y módulos de la aplicación en desarrollo. En la parte superior, se observa una barra de herramientas con opciones para ejecutar, depurar la aplicación, y seleccionar un emulador, ya sea virtual o un dispositivo físico conectado. En el área central y derecha, se visualiza el editor de código, donde se pueden ver y editar los archivos que forman parte del proyecto.

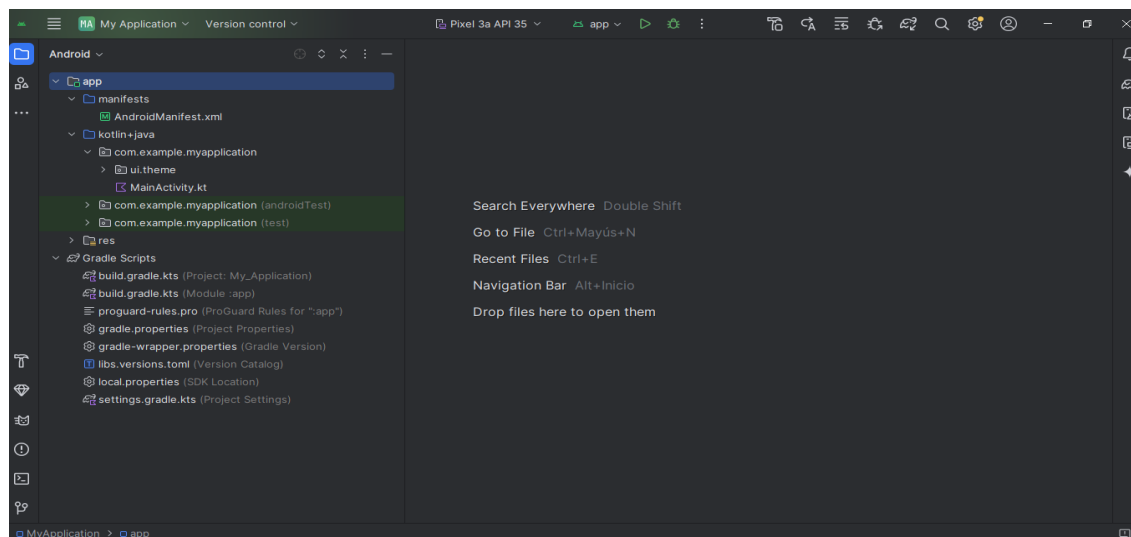


Figura 29: Entorno de Android Studio

La siguiente imagen muestra el proceso para crear una nueva Activity junto con su respectivo archivo XML. Este procedimiento se sigue de la misma manera para todas las Activities que se necesiten crear en la aplicación. Las opciones resaltadas en azul son las que se seleccionan para la creación de las nuevas Activities.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 94 de 134

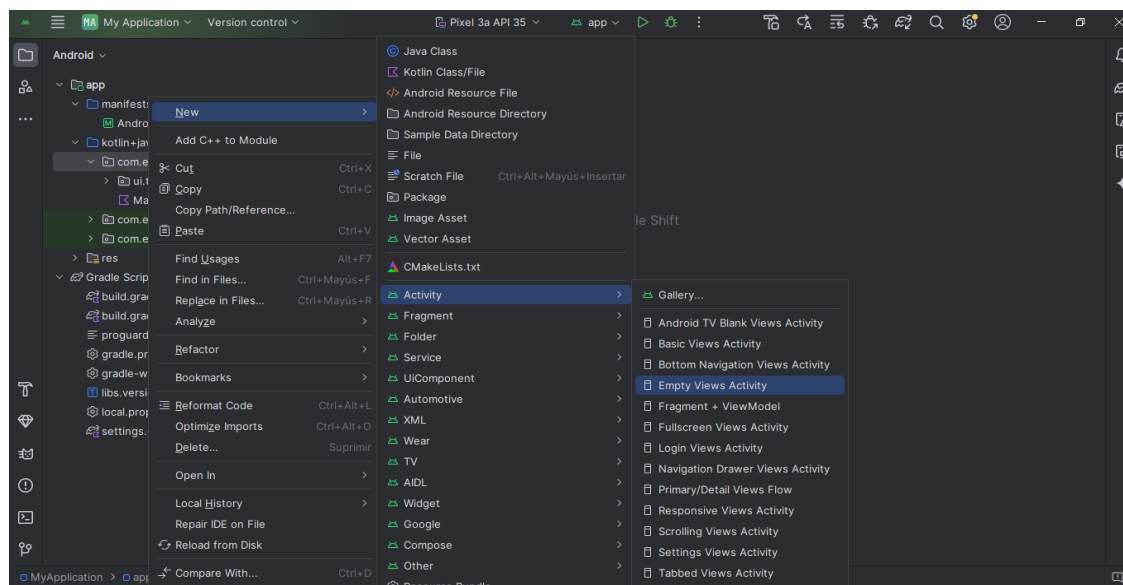


Figura 30: Creación en Android studio de Activitys

Al seleccionar este proceso, se permite asignar un nombre a la Activity; en nuestro caso, le hemos puesto "Inicio", donde implementaremos la lógica para controlar la interfaz principal de la aplicación. Además, nos ofrece la opción de elegir el lenguaje de programación con el que trabajaremos, y en este caso hemos seleccionado Kotlin.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 95 de 134

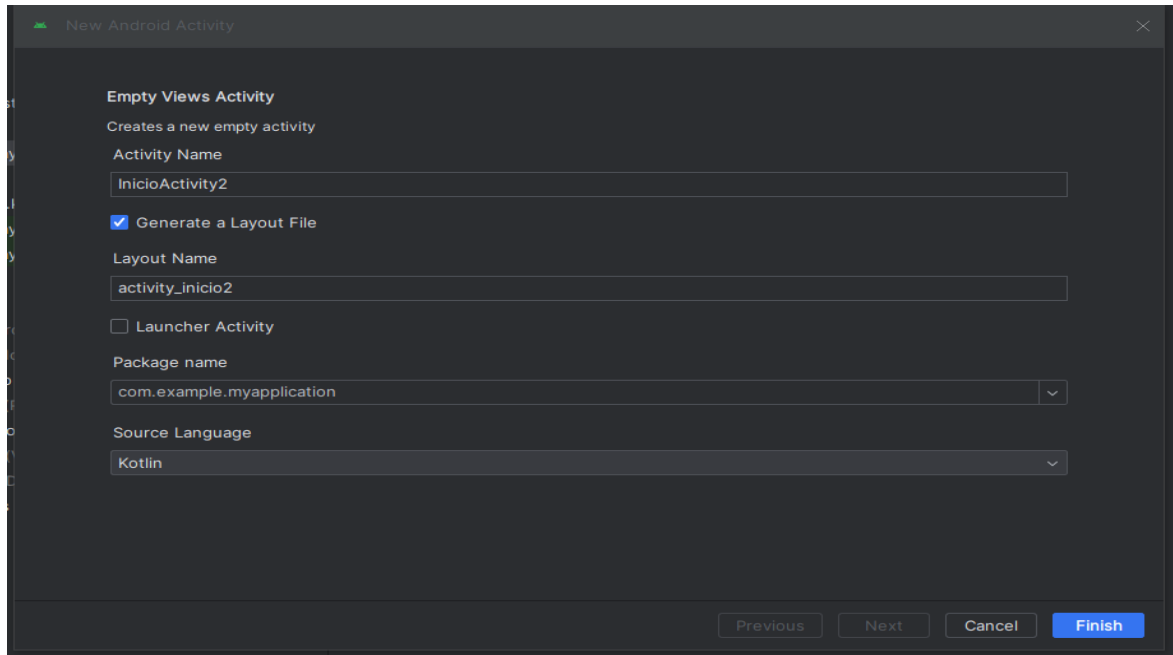


Figura 31: Interfaz 1 Android studio

Una vez finalizado el proceso anterior, Android Studio genera los nuevos archivos correspondientes y nos proporciona el espacio para comenzar a desarrollar las funcionalidades que necesitamos. Se pueden observar tanto el archivo InicioActivity en la sección de código, donde implementaremos la lógica, como el archivo de diseño layout o XML, donde configuraremos la interfaz visual de la actividad.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 96 de 134

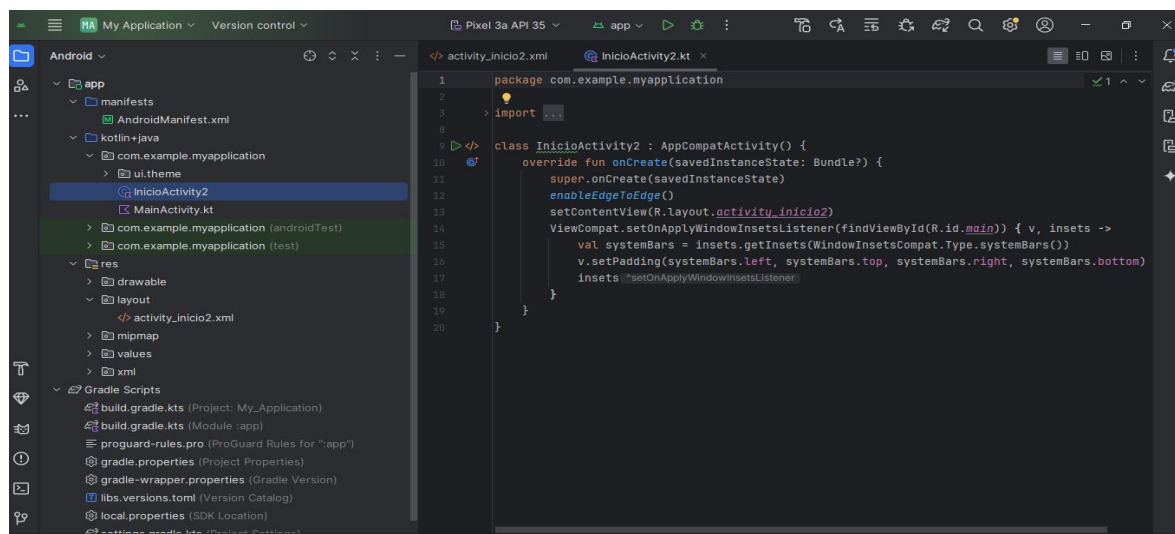


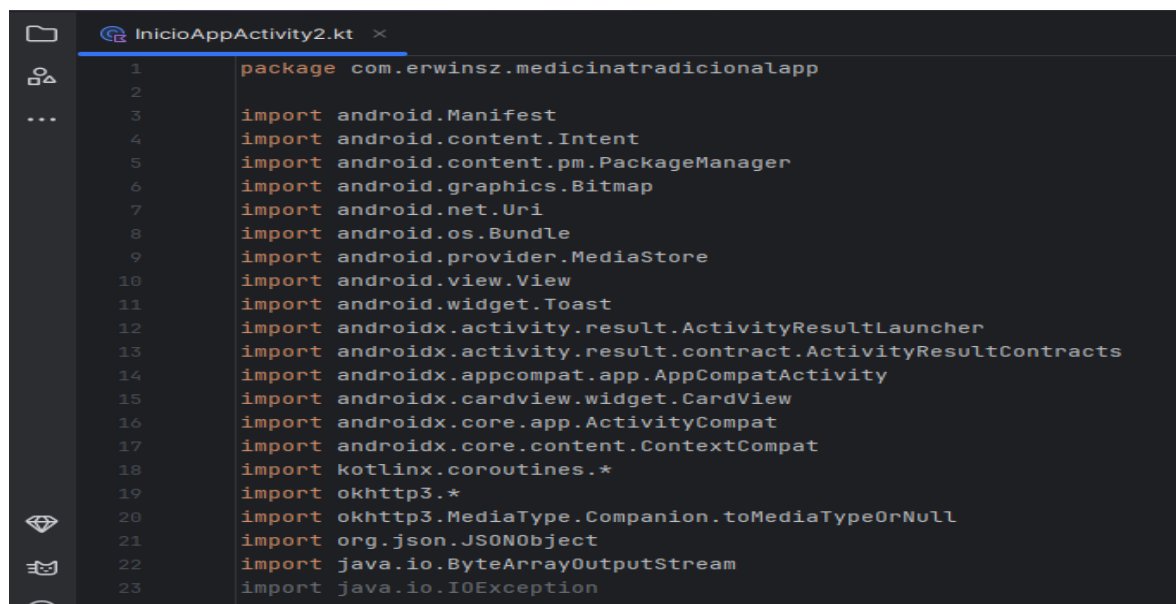
Figura 32: Interfaz 2 Android studio

A continuación, se presentan los archivos previamente creados, que incluyen la lógica necesaria para la conexión y navegación entre las interfaces, así como el código correspondiente para el diseño de las mismas. En el archivo de la Activity, se implementa la funcionalidad que permite la interacción entre las distintas pantallas de la aplicación, mientras que en el archivo XML, se define el diseño visual, especificando la disposición de los elementos, colores, y otros aspectos estéticos de la interfaz.

Para cada interfaz que se quiera realizar se maneja los mismos tipos de archivo y modelo de codificación

Este conjunto de librerías permite que la aplicación capture imágenes desde la cámara o la galería, las procese y las envíe a un servidor, todo gestionando permisos y evitando que la interfaz se bloquee durante operaciones de larga duración.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 97 de 134



```

1 package com.erwinsz.medicinatradicionalapp
2
3 import android.Manifest
4 import android.content.Intent
5 import android.content.pm.PackageManager
6 import android.graphics.Bitmap
7 import android.net.Uri
8 import android.os.Bundle
9 import android.provider.MediaStore
10 import android.view.View
11 import android.widget.Toast
12 import androidx.activity.result.ActivityResultLauncher
13 import androidx.activity.result.contract.ActivityResultContracts
14 import androidx.appcompat.app.AppCompatActivity
15 import androidx.cardview.widget.CardView
16 import androidx.core.app.ActivityCompat
17 import androidx.core.content.ContextCompat
18 import kotlinx.coroutines.*
19 import okhttp3.*
20 import okhttp3.MediaType.Companion.toMediaTypeOrNull
21 import org.json.JSONObject
22 import java.io.ByteArrayOutputStream
23 import java.io.IOException

```

Figura 33: Importación de librerías de Android studio

A continuación, se presenta el código que inicializa los botones de la interfaz gráfica para que puedan ser utilizados en la lógica del programa, permitiendo posteriormente implementar acciones como abrir la cámara, galería, menú, o enviar datos.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 98 de 134

```
class InicioAppActivity2 : AppCompatActivity() {

    private lateinit var cameraLauncher: ActivityResultLauncher<Intent>
    private lateinit var galleryLauncher: ActivityResultLauncher<Intent>
    private lateinit var selectedBitmap: Bitmap
    private lateinit var btnEnviar: CardView
    private lateinit var btnMenu: CardView
    private lateinit var btnCamara: CardView
    private lateinit var btnGaleria: CardView

    companion object {
        private const val REQUEST_CAMERA_PERMISSION = 1
        private const val REQUEST_STORAGE_PERMISSION = 2
    }

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_inicio_app2)

        // Inicializa los botones
        btnEnviar = findViewById(R.id.btnEnviar)
        btnMenu = findViewById(R.id.btnmenu)
        btnCamara = findViewById(R.id.btncamara)
        btnGaleria = findViewById(R.id.btngaleria)
    }
}
```

Figura 34: lógica botones

El siguiente código gestiona la lógica para tomar una foto, mostrar el botón "Enviar" cuando se captura una imagen, y desactivar otros botones después de tomar dicha foto.

```
// Oculta el botón "Enviar" al iniciar la actividad
btnEnviar = findViewById(R.id.btnEnviar)
btnEnviar.visibility = View.GONE

// Configura el launcher para la cámara
cameraLauncher = registerForActivityResult(ActivityResultContracts.StartActivityForResult()) { result ->
    if (result.resultCode == RESULT_OK) {
        val photo = result.data?.extras?.get("data") as? Bitmap
        photo?.let { it: Bitmap
            selectedBitmap = it
            // Muestra el botón "Enviar" después de tomar la foto
            btnEnviar.visibility = View.VISIBLE
            // Desactiva otros botones
            disableOtherButtons()
        }
    }
}
```

Figura 35: Lógica cámara y botón enviar

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 99 de 134

```
// Configura el launcher para la galería
galleryLauncher = registerForActivityResult(ActivityResultContracts.StartActivityForResult()) { result ->
    if (result.resultCode == RESULT_OK) {
        val selectedImageUri: Uri? = result.data?.data
        selectedImageUri?.let { uri ->
            val bitmap = MediaStore.Images.Media.getBitmap(this.contentResolver, uri)
            selectedBitmap = bitmap
            // Muestra el botón "Enviar" después de seleccionar una imagen
            btnEnviar.visibility = View.VISIBLE
            // Desactiva otros botones
            disableOtherButtons()
        }
    }
}

// Configura los botones
btnMenu.setOnClickListener { it: View!
    val intent = Intent( packageContext: this, NavegacionActivity2::class.java)
    startActivity(intent)
}
```

Figura 36: Lógica botón galería

La siguiente imagine muestra el código encargado de gestionar la respuesta del usuario a las solicitudes de permisos y realizar la acción adecuada (abrir la cámara o galería) si los permisos son concedidos.

```
Manejo de permisos
erride fun onRequestPermissionsResult(requestCode: Int, permissions: Array<out String>, grantResults: IntArray) {
    super.onRequestPermissionsResult(requestCode, permissions, grantResults)
    when (requestCode) {
        REQUEST_CAMERA_PERMISSION -> {
            if (grantResults.isNotEmpty() && grantResults[0] == PackageManager.PERMISSION_GRANTED) {
                Toast.makeText( context: this, text: "Permiso de cámara concedido", Toast.LENGTH_SHORT).show()
                openCamera()
            } else {
                Toast.makeText( context: this, text: "Permiso de cámara denegado", Toast.LENGTH_SHORT).show()
            }
        }
        REQUEST_STORAGE_PERMISSION -> {
            if (grantResults.isNotEmpty() && grantResults[0] == PackageManager.PERMISSION_GRANTED) {
                Toast.makeText( context: this, text: "Permiso de almacenamiento concedido", Toast.LENGTH_SHORT).show()
                openGallery()
            } else {
                Toast.makeText( context: this, text: "Permiso de almacenamiento denegado", Toast.LENGTH_SHORT).show()
            }
        }
    }
}
```

Figura 37: Permisos cámara / galería

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 100 de 134

El siguiente código permite tomar una imagen, la convierte a un archivo, la envía al servidor Flask usando una solicitud HTTP POST y maneja la respuesta del servidor (el nombre de la planta reconocida gracias al reconcomiendo de la imagen). Las coroutines en la aplicación permiten realizar una solicitud de red a un servidor Flask, procesar el resultado y luego actualizar la interfaz de usuario con los datos recibidos, todo sin interrumpir al usuario mientras interactúa con la aplicación.

Las coroutines en Kotlin son funciones que pueden suspender su ejecución en un punto determinado y reanudarla más tarde, lo que permite manejar operaciones asincrónicas de manera eficiente sin bloquear el hilo principal. Son una parte fundamental de la programación asíncrona en Kotlin, especialmente útiles para realizar tareas como acceder a bases de datos, llamar a APIs, procesar grandes cantidades de datos, o cualquier otra tarea que pueda llevar tiempo y que no debería interrumpir la experiencia del usuario [48].

```

Función para enviar la imagen al servidor Flask usando coroutines
private fun sendImageToServer(bitmap: Bitmap) {
    // Inicia una coroutine en el scope de Main (UI thread)
    CoroutineScope(Dispatchers.Main).launch { this: CoroutineScope
        try {
            // Cambia a un thread de IO para realizar la operación de red
            withContext(Dispatchers.IO) { this: CoroutineScope
                val client = OkHttpClient()

                // Convierte el Bitmap a un archivo .jpg
                val stream = ByteArrayOutputStream()
                bitmap.compress(Bitmap.CompressFormat.JPEG, quality: 100, stream)
                val imageBytes = stream.toByteArray()

                val requestBody = MultipartBody.Builder()
                    .setType(MultipartBody.FORM)
                    .addFormDataPart(name: "file", filename: "plant.jpg", RequestBody.create("image/jpeg".toMediaTypeOrNull(), imageBytes))
                    .build()
            }
        }
    }
}

```

Figura 38: Solicitud HTTP POST

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 101 de 134

```

val request = Request.Builder()
    .url("http://192.168.156.16:5000/predict")
    .post(requestBody)
    .build()

// Ejecuta la solicitud y espera la respuesta
val response = client.newCall(request).execute()

if (response.isSuccessful) {
    val responseBody = response.body?.string()
    println("Respuesta del servidor: $responseBody")
    responseBody?.let { it: String
        val jsonResponse = JSONObject(it)
        val plantName = jsonResponse.getString( name: "plant")
        println("Nombre de planta recibido: $plantName")

        // Vuelve al hilo principal para mostrar el resultado y navegar a la actividad correspondiente
        withContext(Dispatchers.Main) { this: CoroutineScope
            Toast.makeText(context: this@InicioAppActivity2, text: "Planta reconocida: $plantName", Toast.LENGTH_SHORT).show()
        }
    }
}

```

Figura 39: Coroutines

El siguiente código permite que, tras recibir la respuesta del servidor Flask, la aplicación navegue automáticamente a la actividad correspondiente según la planta identificada. Además, maneja errores y excepciones de manera adecuada para garantizar una buena experiencia de usuario.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 102 de 134

```

// Navegar a la actividad correspondiente
when (plantName.toLowerCase()) {
    "sabila" -> startActivity(Intent( packageContext: this@InicioAppActivity2, sabilaActivity2::class.java))
    "eucalipto" -> startActivity(Intent( packageContext: this@InicioAppActivity2, EucaliptoActivity2::class.java))
    "hierbabuena" -> startActivity(Intent( packageContext: this@InicioAppActivity2, hervabuenaActivity2::class.java))
    "manzanilla" -> startActivity(Intent( packageContext: this@InicioAppActivity2, ManzanillaActivity2::class.java))
    else -> Toast.makeText( context: this@InicioAppActivity2, text: "Planta no reconocida", Toast.LENGTH_SHORT).show()
}
}
} *withContext
} else {
    withContext(Dispatchers.Main) { this: CoroutineScope
        Toast.makeText( context: this@InicioAppActivity2, text: "Error en la respuesta del servidor", Toast.LENGTH_SHORT).show()
    } *withContext
}
}
} catch (e: Exception) {
    withContext(Dispatchers.Main) { this: CoroutineScope
        Toast.makeText( context: this@InicioAppActivity2, text: "Error: ${e.message}", Toast.LENGTH_SHORT).show()
    }
}
}
}

```

Figura 40: Respuesta servidor flask

A continuación, se muestra el archivo XML de la Activity inicio.

Este diseño presentado, configura un ConstraintLayout como el contenedor principal y coloca un ImageView que sirve como fondo con una imagen semitransparente. La imagen de fondo se ajusta para llenar el área disponible, manteniendo su relación de aspecto y mostrando la imagen con cierta transparencia.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 103 de 134

```

<androidx.appcompat.widget.SearchView
    android:id="@+id/buscar"
    android:layout_width="250dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="40dp"
    android:layout_marginBottom="16dp"
    android:queryHint="Buscar..."
    app:layout_constraintBottom_toTopOf="@id/cardBotones"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.496"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@id/cardPresentacion"
    app:layout_constraintVertical_bias="0.0"
    android:background="@drawable/bordes_plantas"/>

<androidx.cardview.widget.CardView
    android:id="@+id/cardBotones"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_margin="16dp"
    android:padding="16dp"
    app:cardCornerRadius="16dp"
    app:cardElevation="8dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent">

<!-- LinearLayout para los botones -->
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:orientation="horizontal"
    android:background="@drawable/bordes_plantas">

    <!-- Botón 1: Galería -->
    <androidx.cardview.widget.CardView
        android:id="@+id/btnGaleria"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_margin="8dp"
        android:layout_weight="1"
        app:cardCornerRadius="16dp"
        app:cardElevation="8dp"
        app:cardBackgroundColor="@color/borderColor">

        <ImageButton
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_margin="25dp"
            android:background="@android:color/transparent"
            android:scaleType="centerInside"
            android:src="@drawable/ic_galeria" />
    </androidx.cardview.widget.CardView>

```

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 104 de 134

```

<!-- Botón 2: Cámara -->
<androidx.cardview.widget.CardView
    android:id="@+id/btncamara"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:layout_weight="1"
    app:cardCornerRadius="16dp"
    app:cardElevation="8dp"
    app:cardBackgroundColor="@color/borderColor">

    <ImageButton
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="25dp"
        android:background="@android:color/transparent"
        android:contentDescription="Camara"
        android:scaleType="centerInside"
        android:src="@drawable/ic_camara" />

</androidx.cardview.widget.CardView>

```

```

<!-- Botón 3: Menú -->
<androidx.cardview.widget.CardView
    android:id="@+id/btnmenu"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:layout_weight="1"
    app:cardCornerRadius="16dp"
    app:cardElevation="8dp"
    app:cardBackgroundColor="@color/borderColor">

    <ImageButton
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="25dp"
        android:background="@android:color/transparent"
        android:scaleType="centerInside"
        android:src="@drawable/ic_menu" />

</androidx.cardview.widget.CardView>

```


	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 105 de 134

```

<!-- Botón 4: Enviar -->
<androidx.cardview.widget.CardView
    android:id="@+id/btnEnviar"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:layout_weight="1"
    android:visibility="gone"
    app:cardCornerRadius="16dp"
    app:cardElevation="8dp"
    app:cardBackgroundColor="@color/buttonBackgroundColor">

    <ImageButton
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="25dp"
        android:background="@android:color/transparent"
        android:scaleType="centerInside"
        android:src="@drawable/ic_enviar" />
    </ImageButton>
</androidx.cardview.widget.CardView>

</LinearLayout>
</androidx.cardview.widget.CardView>

```

Figura 41: Contenedor principal

Así se ve la interfaz principal de la aplicación desarrollada, diseñada para ofrecer una experiencia de usuario intuitiva y visualmente atractiva. Los elementos están dispuestos de manera que facilitan el acceso a las funciones principales de la aplicación, mientras que el diseño estilizado y los efectos visuales buscan tener una presentación profesional y coherente.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 106 de 134



Figura 42: Interfaz principal

De la misma manera se crean todas las interfaces de la aplicación, a continuación, se presentan imágenes del resultado de ello:

Menú:

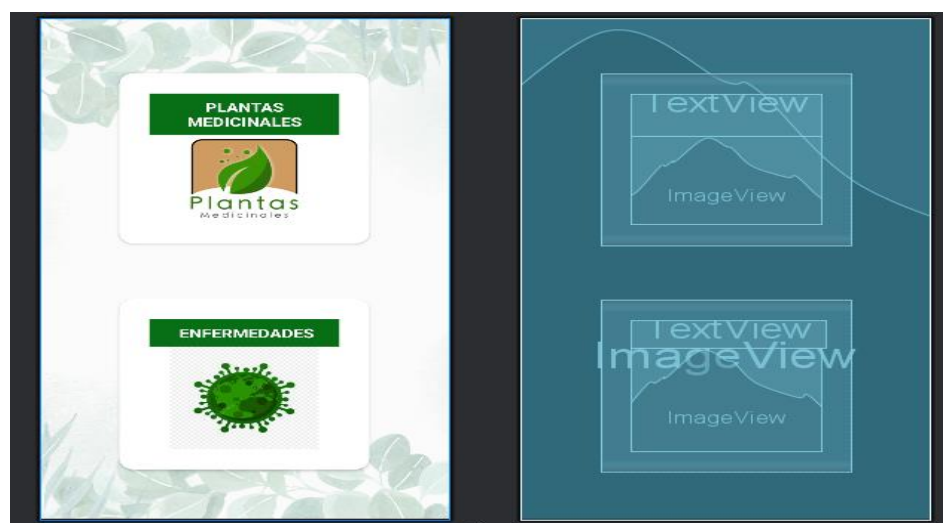


Figura 43:Interfaz Menú

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 107 de 134

Plantas de la aplicación:



Figura 44: Interfaz plantas

Menú enfermedades:

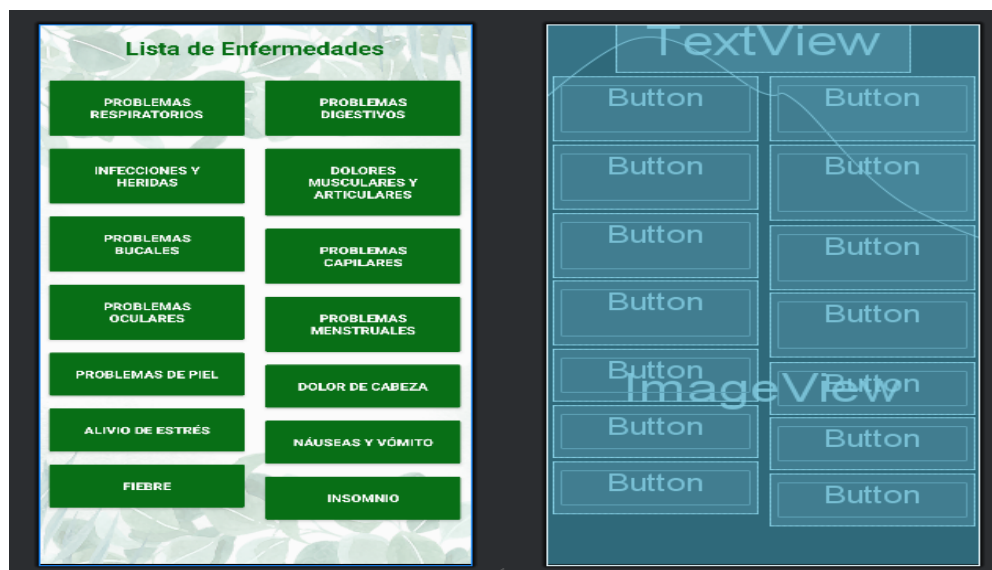


Figura 45: Interfaz lista de enfermedades

Descripción de cada planta:

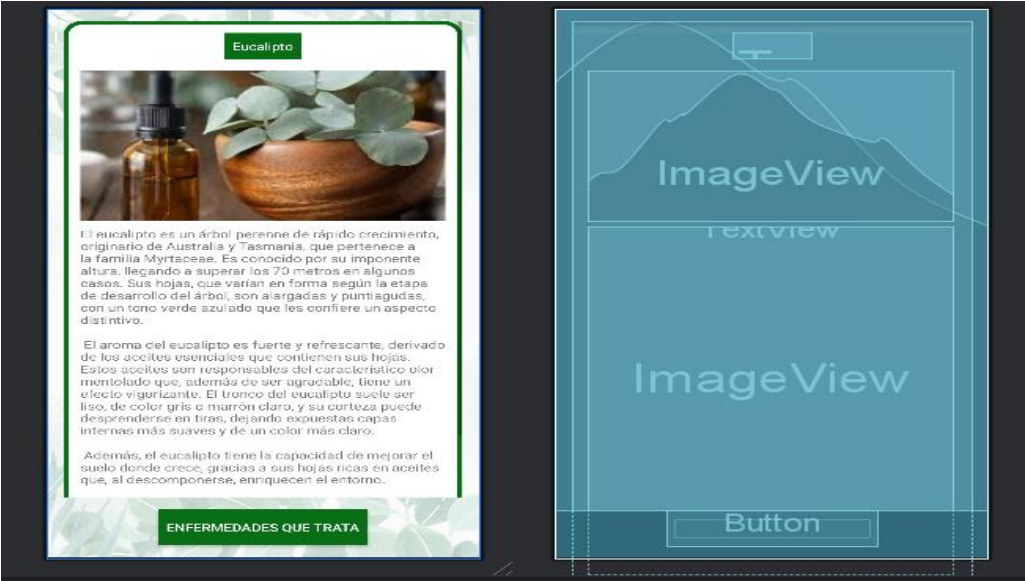


Figura 46: Interfaz descripción por plantas

Enfermedades que trata cada planta:

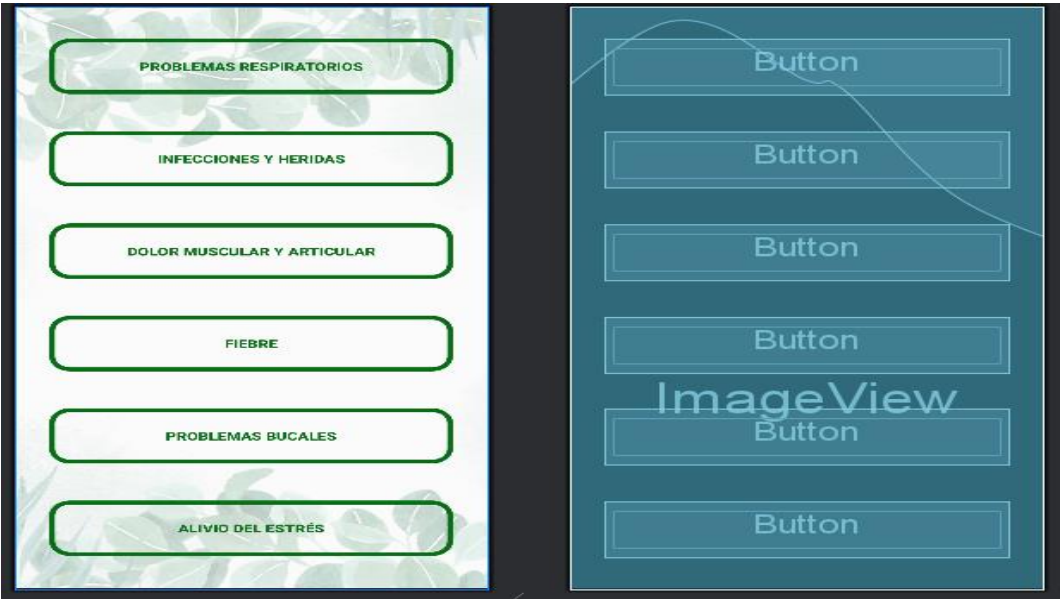


Figura 47: interfaz enfermedades por cada planta

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 109 de 134

Recetas y usos de cada planta:

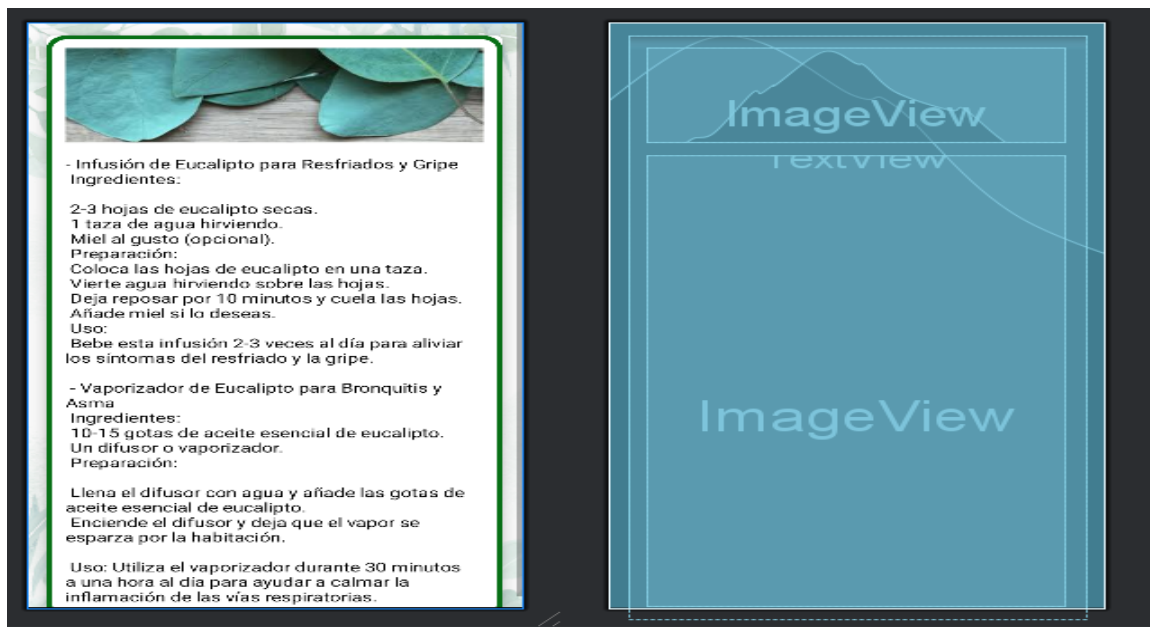


Figura 48: Interfaz recetas y usos de cada planta

Validación y pruebas (Se presentan en el capítulo 5).

Validación de comunicación entre backend y frontend (Se presentan en el capítulo 5).

4.8. Pruebas

En esta parte del ciclo de vida del desarrollo de la aplicación móvil se deben realizar pruebas unitarias, y de integración para ver el funcionamiento de la aplicación, los resultados se pueden observar en el capítulo 5 de este documento.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 110 de 134

V. CAPÍTULO

En este capítulo se describen las diferentes pruebas realizadas para garantizar el correcto funcionamiento de la aplicación, abarcando tanto el backend como el frontend. Cada fase de desarrollo ha sido evaluada a través de diferentes tipos de pruebas, desde unitarias hasta de integración, con el fin de asegurar la calidad y precisión del modelo de clasificación de plantas medicinales y la correcta interacción del usuario con la aplicación en Android.

Las pruebas se organizaron en varias categorías para cubrir los distintos componentes del sistema, desde la hasta la funcionalidad general de la API y la interacción de las interfaces de usuario. A continuación, se presentan los distintos tipos de pruebas realizadas.

5.1. Pruebas unitarias del modelo archivo entrenamiento

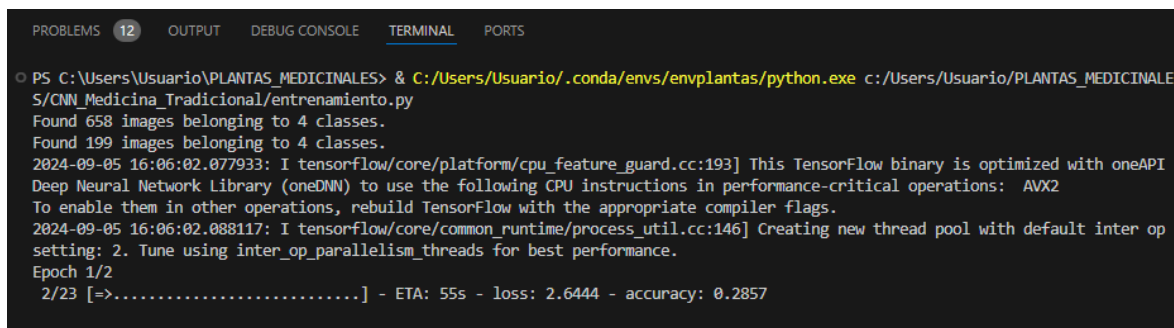
En la imagen se muestra la salida de la ejecución del script de entrenamiento del modelo de red neuronal convolucional (CNN) con TensorFlow y Keras, en la imagen se puede observar que el generador de datos a encontrado 658 imágenes en el directorio de entrenamiento, que están distribuidas en 4 clases diferentes.

Al referirnos a 4 clases estamos indicando las categorías o etiquetas en las que se clasifican los datos. En el archivo de entrenamiento, 658 imágenes están distribuidas en 4 clases, lo que significa que esas imágenes pertenecen a 4 categorías diferentes.

En este caso se está trabajando en un modelo para reconocer plantas medicinales, las clases son las diferentes plantas que el modelo entrena para identificar, como "hierbabuena", "manzanilla", "sábila" y "eucalipto". Cada imagen en el archivo de entrenamiento pertenece a una

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 111 de 134

de esas clases, y el modelo utiliza estas imágenes para evaluar su capacidad de identificar correctamente a qué clase pertenece cada una.



```

PROBLEMS 12 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\Usuario\PLANTAS_MEDICINALES> & C:/Users/Usuario/.conda/envs/envplantas/python.exe c:/Users/Usuario/PLANTAS_MEDICINALES/CNN_Medicina_Tradicional/entrenamiento.py
Found 658 images belonging to 4 classes.
Found 199 images belonging to 4 classes.
2024-09-05 16:06:02.077933: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical operations: AVX2
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
2024-09-05 16:06:02.088117: I tensorflow/core/common_runtime/process_util.cc:146] Creating new thread pool with default inter op setting: 2. Tune using inter_op_parallelism_threads for best performance.
Epoch 1/2
2/23 [=.....] - ETA: 55s - loss: 2.6444 - accuracy: 0.2857

```

Figura 49: Consola entrenamiento inicio

También ha encontrado 199 imágenes en el directorio de validación, que también están distribuidas en 4 clases.

El progreso del Entrenamiento lo muestra de la siguiente manera:

- **Epoch 1/2:** un Epoch es una pasada completa de todo el conjunto de datos de entrenamiento a través del modelo, aquí nos está indicando que está en el primer ciclo de dos que tiene el modelo de entrenamiento.

A continuación, indica el inicio de la primera época:

- 23/23 [=====] - 110s 5s/step - loss: 1.5269 - accuracy: 0.6937 - - -
- **val_loss: 0.9439 - val_accuracy: 0.8929:** Muestra el progreso del entrenamiento de la primera época:
- 23/23: 23 pasos (batches) por época.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 112 de 134

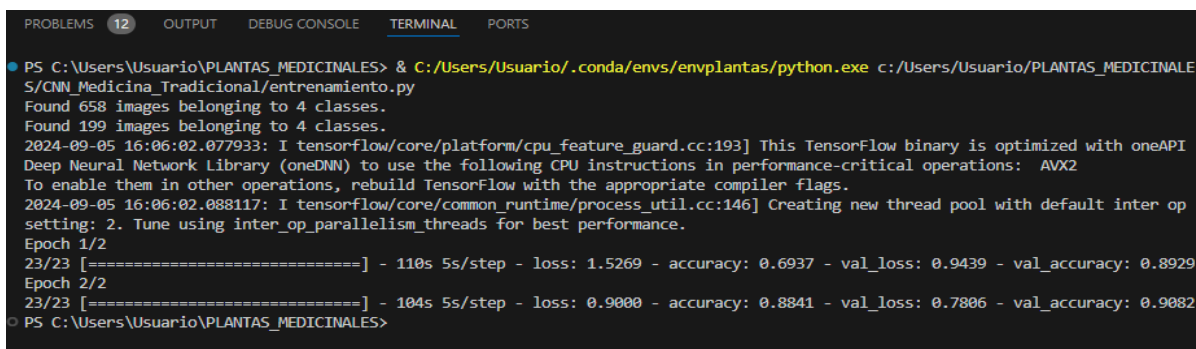
- 110s 5s/step: Tiempo total para completar la época (110 segundos), con un promedio de 5 segundos por paso.
- loss: 1.5269: La pérdida (loss) del modelo en el conjunto de entrenamiento para esta época.
- accuracy: 0.6937: La precisión (accuracy) del modelo en el conjunto de entrenamiento para esta época.
- val_loss: 0.9439: La pérdida del modelo en el conjunto de validación para esta época.
- val_accuracy: 0.8929: La precisión del modelo en el conjunto de validación para esta época.

➤ Epoch 2/2:

- A continuación, indica el inicio de la segunda época del entrenamiento.
- 23/23 [=====] - 104s 5s/step - loss: 0.9000 - accuracy: 0.8841 –
- **val_loss: 0.7806 - val_accuracy: 0.9082:** Muestra el progreso del entrenamiento de la segunda época:
- loss: 0.9000: La pérdida del modelo en el conjunto de entrenamiento para esta época.
- accuracy: 0.8841: La precisión del modelo en el conjunto de entrenamiento para esta época.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 113 de 134

- val_loss: 0.7806: La pérdida del modelo en el conjunto de validación para esta época.
- val_accuracy: 0.9082: La precisión del modelo en el conjunto de validación para esta época.



```

PROBLEMS 12 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\Usuario\PLANTAS_MEDICINALES> & C:/Users/Usuario/.conda/envs/envplantas/python.exe c:/Users/Usuario/PLANTAS_MEDICINALES/CNN_Medicina_Tradicional/entrenamiento.py
Found 658 images belonging to 4 classes.
Found 199 images belonging to 4 classes.
2024-09-05 16:06:02.077933: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical operations: AVX2
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
2024-09-05 16:06:02.088117: I tensorflow/core/common_runtime/process_util.cc:146] Creating new thread pool with default inter op setting: 2. Tune using inter_op_parallelism_threads for best performance.
Epoch 1/2
23/23 [=====] - 110s 5s/step - loss: 1.5269 - accuracy: 0.6937 - val_loss: 0.9439 - val_accuracy: 0.8929
Epoch 2/2
23/23 [=====] - 104s 5s/step - loss: 0.9000 - accuracy: 0.8841 - val_loss: 0.7806 - val_accuracy: 0.9082
PS C:\Users\Usuario\PLANTAS_MEDICINALES>

```

Figura 50: Consola entrenamiento terminado

A continuación, se muestran las gráficas obtenidas de los entrenamientos que se realizaron variando el número de épocas y valor de la tasa de aprendizaje, las gráficas muestran la precisión y la pérdida del modelo durante el entrenamiento.

5.1.1. Interpretación Gráficas de entrenamiento

En la primera grafica tenemos un valor de épocas de 2 y la tasa de aprendizaje de 0.0005

El valor 0.005 indica que el modelo hará cambios muy pequeños a los pesos en cada iteración, lo cual puede ser útil para asegurar una convergencia más estable y precisa, aunque puede requerir más tiempo y épocas para alcanzar un rendimiento óptimo, basándonos en las gráficas donde varían los valores de aprendizaje y épocas observamos que entre más bajo es el

valor de la taza de aprendizaje, la neurona queda mejor entrenada. Se muestra en la Figura 75: Graficas entrenamiento 4.

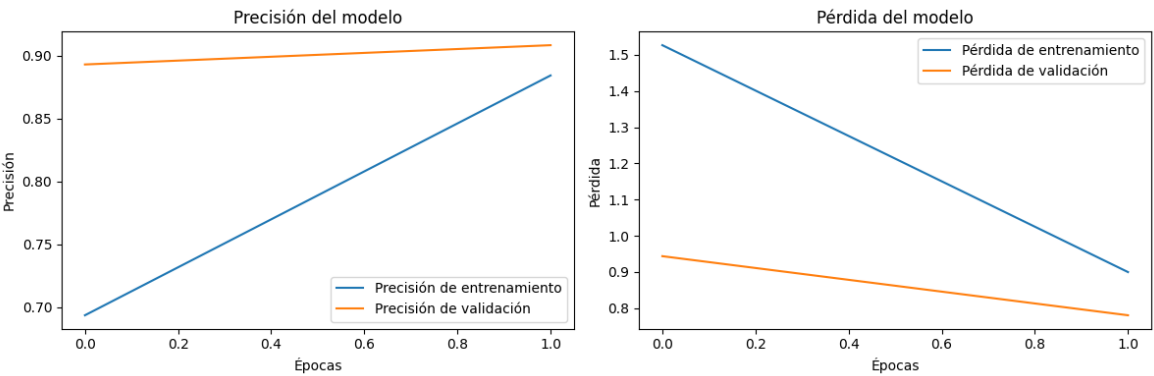


Figura 51: Gráfica entrenamiento 1

Gráfica de precisión:

- Eje Y (Precisión): Indica qué tan bien el modelo está clasificando las imágenes correctamente.
- Eje X (Épocas): Representa el número de ciclos de entrenamiento.
- La línea azul muestra la precisión del modelo en el conjunto de entrenamiento.
- La línea naranja muestra la precisión en el conjunto de validación.
- Se observa un incremento en la precisión (línea azul) del entrenamiento a medida que avanzan las épocas. Esto indica que el modelo está aprendiendo y mejorando su capacidad para clasificar correctamente las imágenes en el conjunto de entrenamiento.
- La precisión (línea naranja) del conjunto de validación es estable y más alta desde el principio. Esto podría significar que el modelo está generalizando bien desde el

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 115 de 134

inicio, pero podría requerir más épocas para cerrar la brecha entre el entrenamiento y la validación.

Gráfica de pérdida:

- Eje Y (Pérdida): Mide el error del modelo. Un valor más bajo indica un mejor rendimiento.
- Eje X (Épocas): Número de ciclos de entrenamiento.
- La línea azul representa la pérdida en el conjunto de entrenamiento.
- La línea naranja representa la pérdida en el conjunto de validación.
- Tanto la pérdida de entrenamiento como la de validación disminuyen a lo largo de las épocas, lo que significa que el modelo está aprendiendo y ajustando sus predicciones con mayor precisión.
- La pérdida de validación es más baja que la de entrenamiento desde el principio, lo cual es un buen indicador de que el modelo no está sobreajustando (overfitting) en las primeras épocas.
- El **overfitting** (sobreajuste) ocurre cuando un modelo de aprendizaje automático se ajusta demasiado a los datos de entrenamiento, aprendiendo no solo los patrones generales sino también el "ruido" o las características específicas de ese conjunto de datos. [49] .

Segunda grafica donde se tiene un numero de épocas de 10 y la taza de aprendizaje es de 0.010.

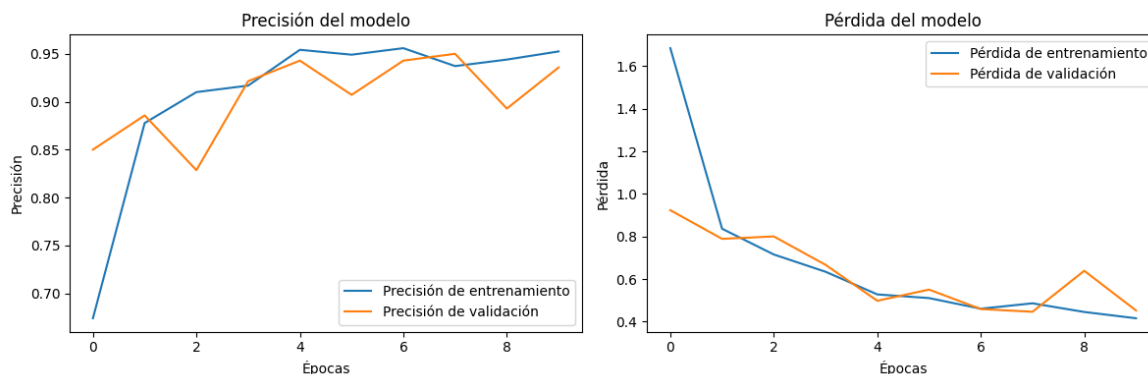


Figura 52: Gráfica entrenamiento 2

Gráfica de precisión:

La precisión en el conjunto de entrenamiento (línea azul) mejora rápidamente en las primeras épocas, alcanzando más del 90% en la época 2 y estabilizándose cerca del 95% a partir de la época 4.

La precisión de validación (línea naranja) también sigue una tendencia positiva, pero muestra más fluctuaciones. A partir de la época 4, parece alcanzar un pico en el 92-93%, aunque baja en algunas épocas intermedias.

La fluctuación en la precisión de validación sugiere que el modelo podría estar experimentando cierto grado de overfitting, donde el modelo está aprendiendo los detalles del conjunto de entrenamiento de manera que no se generaliza tan bien para los datos de validación.

Grafica de perdida:

La pérdida de entrenamiento (línea azul) disminuye de manera constante durante las primeras épocas y alcanza valores bajos (alrededor de 0.4) hacia la época 9.

La pérdida de validación (línea naranja) también disminuye inicialmente, pero muestra fluctuaciones más notables hacia el final. Aunque baja hasta 0.4, hay un pequeño repunte en algunas épocas intermedias.

Estas fluctuaciones en la pérdida de validación son otro indicador de overfitting o que el modelo no está generalizando perfectamente a los datos de validación.

En la tercera grafica se tiene un número de épocas de 10 y la tasa de aprendizaje es de 0.0005.

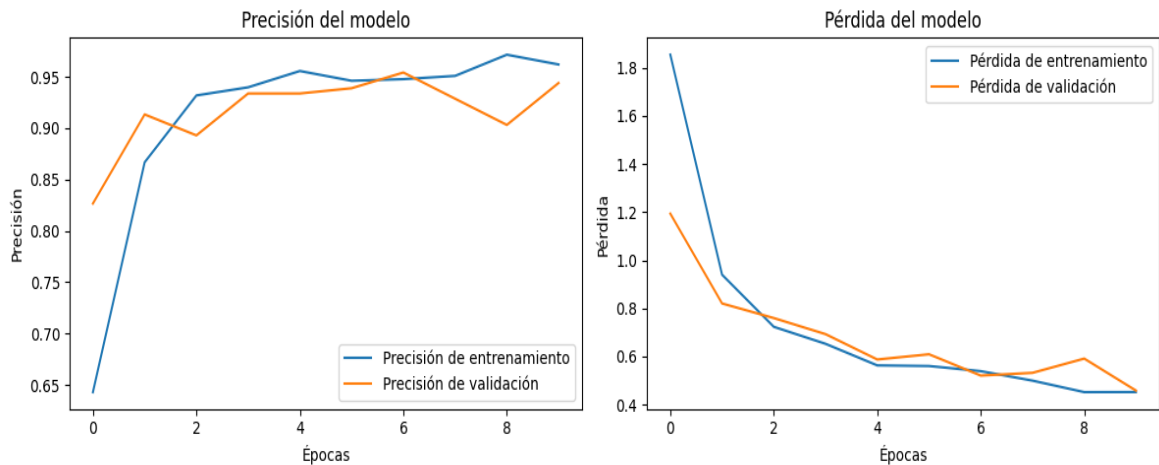


Figura 53: Gráfica entrenamiento 3

Gráfico de Precisión:

La precisión de entrenamiento aumenta de manera progresiva durante las primeras épocas, alcanzando un valor cercano a 0.95. Esto indica que el modelo está aprendiendo los patrones de los datos de entrenamiento.

La precisión de validación sigue un patrón similar al del entrenamiento, lo cual es una buena señal de que el modelo está generalizando bien en los datos de validación.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 118 de 134

Aunque hay una ligera fluctuación entre las épocas 5 y 9 en la precisión de validación, no es lo suficientemente pronunciada como para preocuparnos por un sobreajuste fuerte.

Gráfico de Pérdida:

La pérdida de entrenamiento disminuye de forma significativa en las primeras épocas y continúa descendiendo hasta valores cercanos a 0.4, lo cual indica que el modelo está ajustándose bien a los datos.

La pérdida de validación sigue una tendencia similar, disminuyendo de manera consistente y manteniéndose en niveles bajos, lo que indica que el modelo está generalizando de manera adecuada a los datos que no ha visto antes.

Al final de las 10 épocas, la pérdida de entrenamiento y la pérdida de validación están bastante alineadas, lo que sugiere que el modelo está bien ajustado sin indicios claros de sobreajuste.

La tasa de aprendizaje más baja (0.0005) parece haber permitido un entrenamiento más controlado, evitando fluctuaciones bruscas y permitiendo al modelo generalizar mejor. A diferencia de tasas de aprendizaje más altas, donde los cambios pueden ser más pronunciados y llevar a un sobreajuste, aquí vemos que el modelo mantiene un rendimiento estable tanto en entrenamiento como en validación.

Si bien hay algunas ligeras variaciones en la precisión de validación, estas parecen normales dentro del rango de fluctuación aceptable y no son indicativas de problemas graves de sobreajuste.

En la cuarta grafica se tiene un numero de épocas de 50 y una taza de aprendizaje de 0.0001.

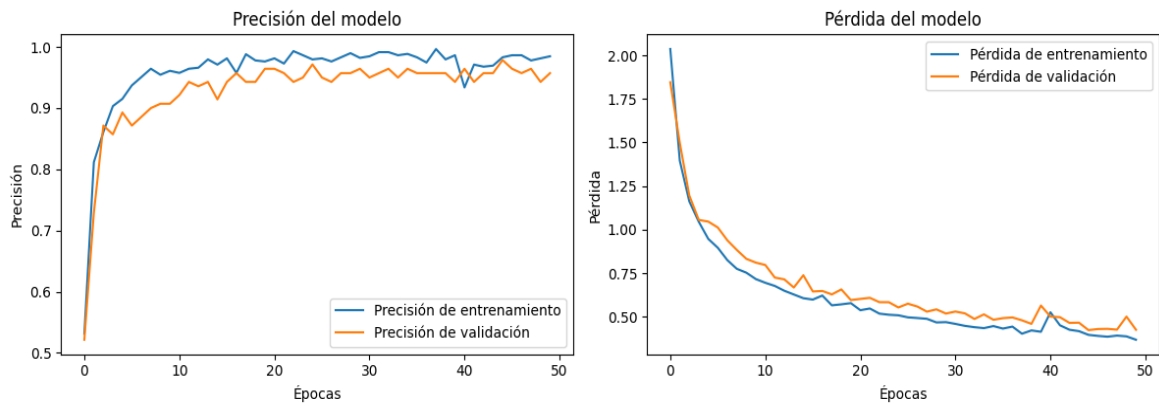


Figura 54: Gráfica entrenamiento 4

Gráfico de Precisión:

La precisión de entrenamiento aumenta rápidamente en las primeras épocas, alcanzando casi un valor perfecto de 1.0 alrededor de la época 20. Sin embargo, después de esto, la precisión se estabiliza, pero con pequeñas fluctuaciones hacia el final del entrenamiento.

La precisión de validación también muestra una tendencia ascendente en las primeras épocas, pero se estabiliza alrededor de 0.9 y presenta pequeñas variaciones a lo largo del tiempo.

La brecha entre la precisión de entrenamiento y la de validación parece ampliarse después de las primeras 20 épocas, lo cual podría ser una señal temprana de sobreajuste, ya que el modelo parece ajustarse más a los datos de entrenamiento que a los de validación.

Gráfico de Pérdida:

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 120 de 134

La pérdida de entrenamiento disminuye de manera consistente, alcanzando un valor de aproximadamente 0.5 al final del proceso. La pérdida sigue disminuyendo, aunque muy lentamente después de las primeras 20 épocas.

La pérdida de validación sigue una tendencia similar, pero es notable que, en algunas épocas, la pérdida de validación es mayor que la de entrenamiento, lo que también podría ser una señal de sobreajuste en algunas etapas.

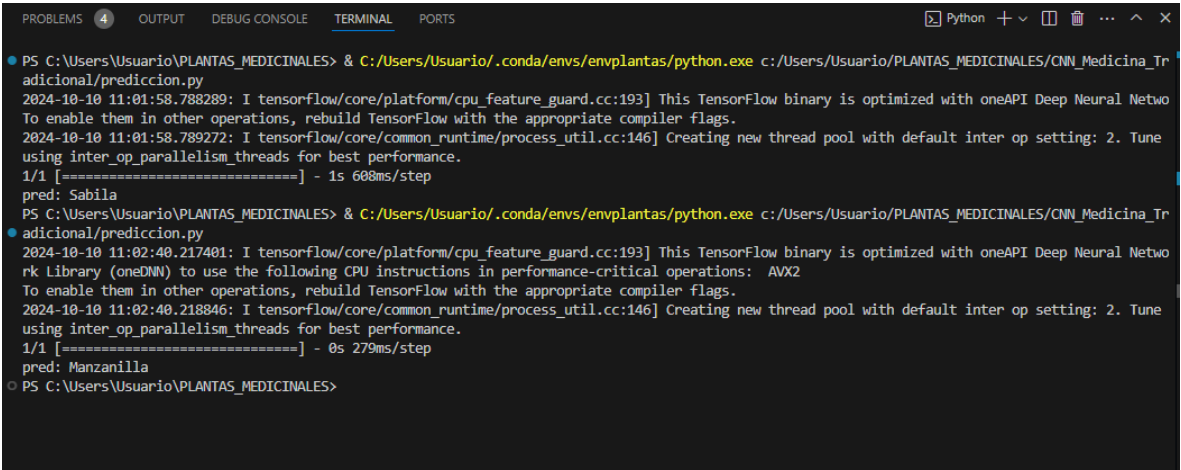
A partir de la época 40, tanto la pérdida de validación como la de entrenamiento parecen estabilizarse, aunque con pequeñas fluctuaciones hacia el final del entrenamiento.

La tasa de aprendizaje baja (0.0001) permitió un entrenamiento más gradual y controlado, lo que parece haber evitado fluctuaciones bruscas en las primeras épocas. Sin embargo, la estabilización de la precisión de validación y las fluctuaciones en la pérdida de validación después de la época 20 sugieren que el modelo podría estar comenzando a sobre ajustarse a los datos de entrenamiento.

Aunque no hay indicios graves de sobreajuste, el hecho de que la precisión de entrenamiento continúe aumentando mientras que la precisión de validación se estabiliza es un comportamiento típico de un modelo que está aprendiendo demasiado los detalles específicos del conjunto de entrenamiento y no está generalizando tan bien a los datos de validación.

5.2. Pruebas unitarias del modelo archivo predicción.

En las siguientes imágenes se muestra la ejecución del script de predicción en Python utilizando TensorFlow para hacer predicción con el modelo de red neuronal convolucional (CNN).



```
PS C:\Users\Usuario\PLANTAS_MEDICINALES> & C:/Users/Usuario/.conda/envs/envplantas/python.exe c:/Users/Usuario/PLANTAS_MEDICINALES/CNN_Medicina_Tr
adicional/prediccion.py
2024-10-10 11:01:58.788289: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimized with oneAPI Deep Neural Netwo
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
2024-10-10 11:01:58.789272: I tensorflow/core/common_runtime/process_util.cc:146] Creating new thread pool with default inter op setting: 2. Tune
using inter_op_parallelism_threads for best performance.
1/1 [=====] - 1s 608ms/step
pred: Sabila
PS C:\Users\Usuario\PLANTAS_MEDICINALES> & C:/Users/Usuario/.conda/envs/envplantas/python.exe c:/Users/Usuario/PLANTAS_MEDICINALES/CNN_Medicina_Tr
adicional/prediccion.py
2024-10-10 11:02:40.217401: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimized with oneAPI Deep Neural Netwo
rk Library (oneDNN) to use the following CPU instructions in performance-critical operations: AVX2
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
2024-10-10 11:02:40.218846: I tensorflow/core/common_runtime/process_util.cc:146] Creating new thread pool with default inter op setting: 2. Tune
using inter_op_parallelism_threads for best performance.
1/1 [=====] - 0s 279ms/step
pred: Manzanilla
PS C:\Users\Usuario\PLANTAS_MEDICINALES>
```

Figura 55: Consola predicción

En la imagen anterior se puede observar la predicción correcta de plantas, sábila y manzanilla, para ello se ingresa en el código del archivo predicción el nombre de alguna planta de las 4 clase que tiene el modelo y al ejecutarse hace el proceso de predicción para tener los resultados que a continuación se muestran:

Progreso de la Predicción planta 1:

1/1 [=====] - 0s 263ms/step: Indica que se ha realizado una única predicción (de un lote de imágenes) y que el tiempo de procesamiento para esta predicción fue de 263 milisegundos.

pred: Sabila: La predicción realizada por el modelo indica que la imagen proporcionada corresponde a la clase "Sábila".

Progreso de la Predicción planta 2

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 122 de 134

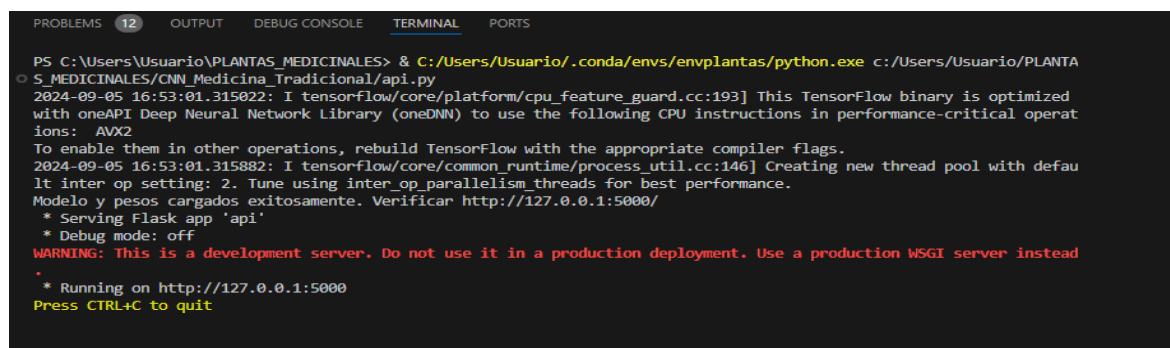
1/1 [=====] - 0s 261ms/step: Indica que se ha realizado una única predicción (de un lote de imágenes) y que el tiempo de procesamiento para esta predicción fue de 261 milisegundos.

pred: Manzanilla: La predicción realizada por el modelo indica que la imagen proporcionada corresponde a la clase "Manzanilla".

En conclusión, el archivo de predicción está realizando la clasificación correctamente, dando a entender que el proceso de entrenamiento del modelo se ha ejecutado correctamente.

5.3. Pruebas unitarias del modelo archivo servidor

La siguiente imagen indica que la aplicación Flask, que está utilizando TensorFlow para hacer predicciones con un modelo de aprendizaje profundo, se está ejecutando correctamente y está lista para recibir solicitudes en la URL proporcionada. Para realizar pruebas o interactuar con la API, se puede hacer visitando <http://127.0.0.1:5000> en el navegador web.



```

PS C:\Users\Usuario\PLANTAS_MEDICINALES> & C:/Users/Usuario/.conda/envs/envplantas/python.exe c:/Users/Usuario/PLANTA
MEDICINALES/CNN_Medicina_Tradicional/api.py
2024-09-05 16:53:01.315022: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimized
with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical operat
ions:  AVX2
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
2024-09-05 16:53:01.315882: I tensorflow/core/common_runtime/process_util.cc:146] Creating new thread pool with defau
lt inter op setting: 2. Tune using inter_op_parallelism_threads for best performance.
Modelo y pesos cargados exitosamente. Verificar http://127.0.0.1:5000/
* Serving Flask app 'api'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead
* Running on http://127.0.0.1:5000
Press CTRL+C to quit

```

Figura 56: Consola servidor

5.3.1. Pruebas de funcionamiento del modelo con su respectivo api (entrenamiento, predicción, servidor).

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 123 de 134

A continuación, se muestra el funcionamiento del modelo que se entrenó para que clasifique la planta que se elige y el funcionamiento del servidor en conjunto con el modelo.

En esta imagen se muestra el servidor corriendo con un diseño de interfase básico en el cual da la opción de seleccionar la planta que se quiere analizar y dar información al respecto.

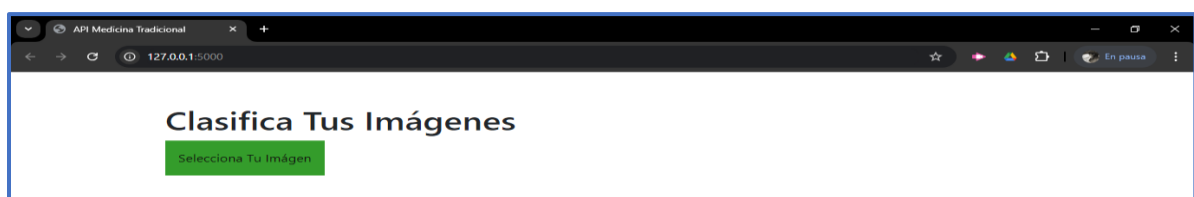


Figura 57: Servidor 1 corriendo

Aquí se puede visualizar el menú de Imágenes para realizar la selección de la planta a predecir.

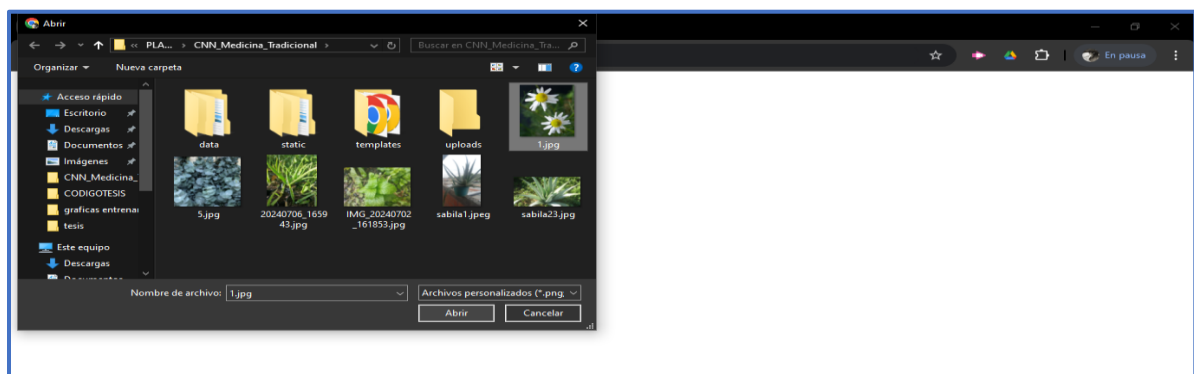


Figura 58: Prueba 1 de predicción en servidor

En esta imagen se observa que se seleccionó la imagen de manzanilla, se da clic en el botón Predicciona tu imagen y nos dará el nombre de la planta seleccionada.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 124 de 134

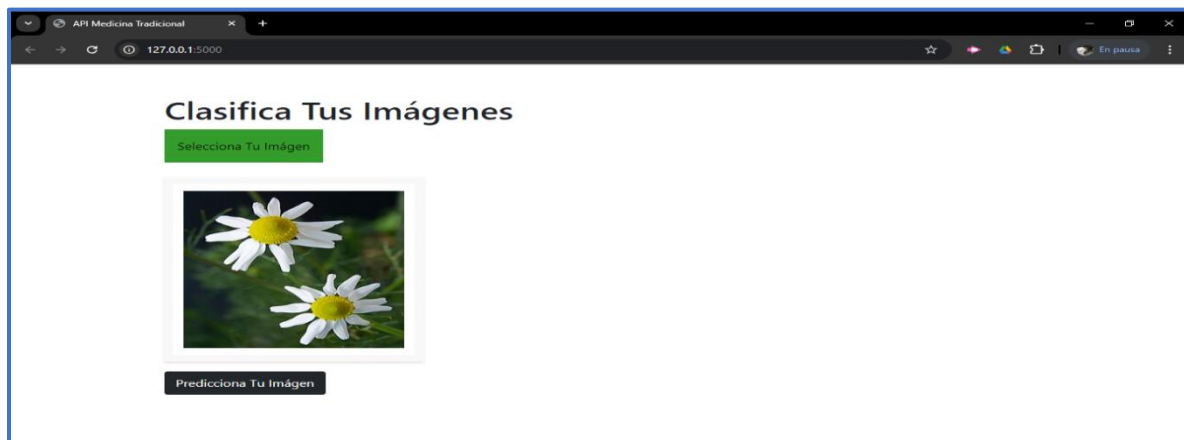


Figura 59: Prueba 2 de predicción en servidor

Aquí observamos el resultado del proceso mostrándonos que efectivamente es una planta de manzanilla, considerando la imagen seleccionada.

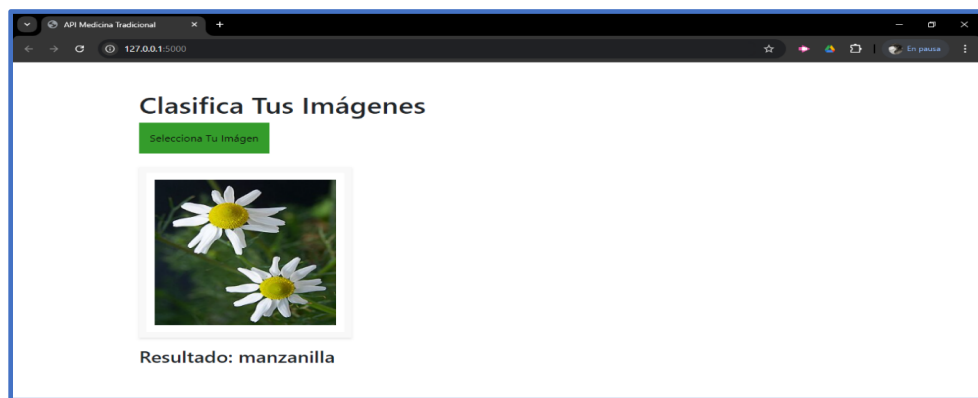


Figura 60: Prueba 3 de predicción en servidor

A continuación, se muestra el mismo proceso, pero realizado para la planta de sábila.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 125 de 134

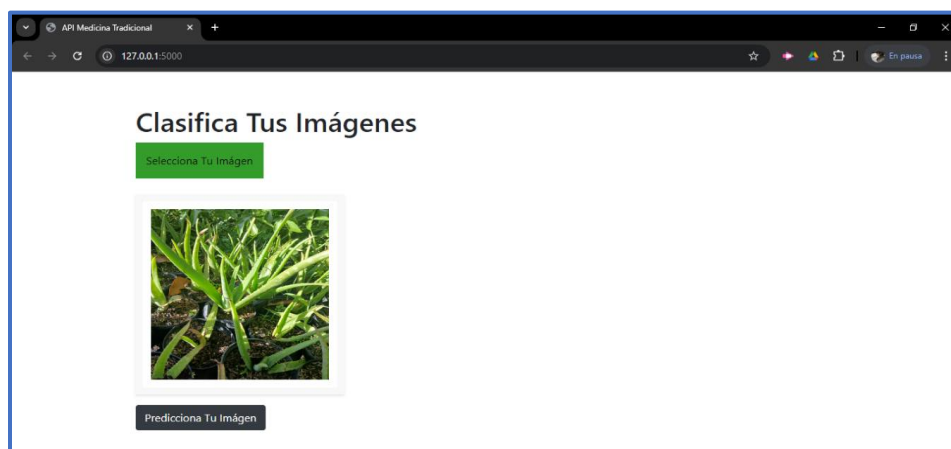


Figura 61: Prueba 4 de predicción en servidor

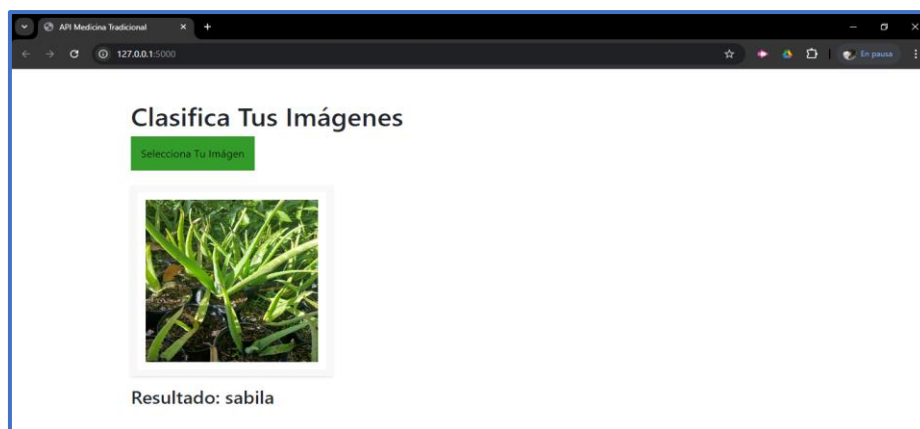
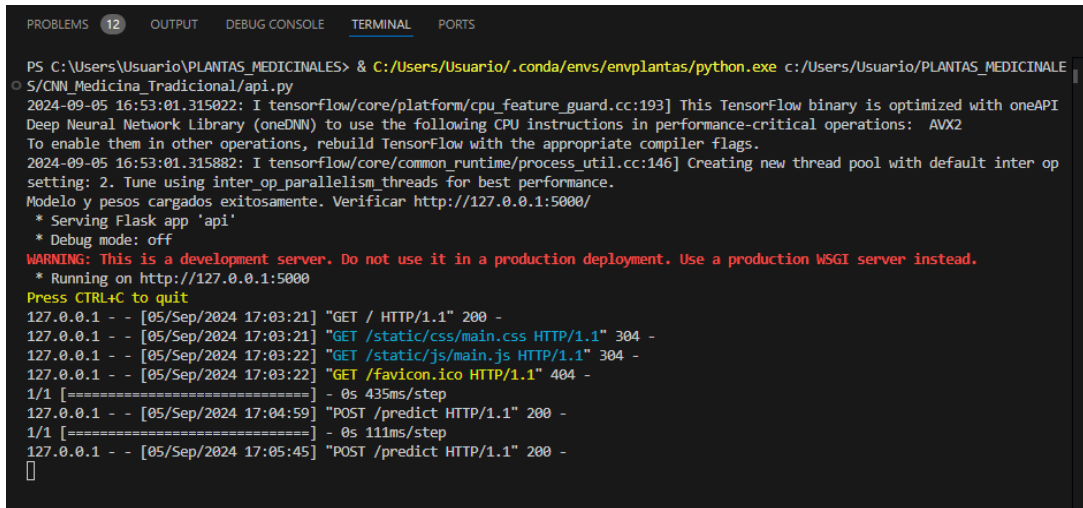


Figura 62: Prueba 5 de predicción en servidor

A continuación, se muestra la salida de la terminal mientras se ejecuta la aplicación Flask en modo de desarrollo que sirve el modelo de predicción de imágenes, se ha lanzado correctamente en el entorno de desarrollo, con la dirección <http://127.0.0.1:5000/>, muestra un mensaje que indica que los pesos del modelo y la arquitectura de la red se han cargado con éxito, también La aplicación está respondiendo a varias solicitudes (request) HTTP en este caso los GET indican que los archivos para dar estilo a la interfaz web se han cargado correctamente

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 126 de 134

y los POST están enviando la imagen al servidor Flask para realizar una predicción y dándonos el código 200 significa que la solicitud fue exitosa.



```

PS C:\Users\Usuario\PLANTAS_MEDICINALES> & C:/Users/Usuario/.conda/envs/envplantas/python.exe c:/Users/Usuario/PLANTAS_MEDICINALES/S/CNN_Medicina_Tradicional/api.py
2024-09-05 16:53:01.315022: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical operations: AVX2
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
2024-09-05 16:53:01.315882: I tensorflow/core/common_runtime/process_util.cc:146] Creating new thread pool with default inter op setting: 2. Tune using inter_op_parallelism_threads for best performance.
Modelo y pesos cargados exitosamente. Verificar http://127.0.0.1:5000/
* Serving Flask app 'api'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
127.0.0.1 - - [05/Sep/2024 17:03:21] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [05/Sep/2024 17:03:21] "GET /static/css/main.css HTTP/1.1" 304 -
127.0.0.1 - - [05/Sep/2024 17:03:22] "GET /static/js/main.js HTTP/1.1" 304 -
127.0.0.1 - - [05/Sep/2024 17:03:22] "GET /favicon.ico HTTP/1.1" 404 -
1/1 [=====] - 0s 435ms/step
127.0.0.1 - - [05/Sep/2024 17:04:59] "POST /predict HTTP/1.1" 200 -
1/1 [=====] - 0s 111ms/step
127.0.0.1 - - [05/Sep/2024 17:05:45] "POST /predict HTTP/1.1" 200 -

```

Figura 63: Prueba 6 de servidor en corriendo

5.4. Pruebas unitarias de las interfaces de la aplicación en Android:

A continuación, se presenta que Gradle, el cual es un asistente encargado de organizar y automatizar todas las tareas necesarias para el correcto funcionamiento de un proyecto de software. En el desarrollo de una aplicación Android, Gradle se asegura de ensamblar el código, descargar y configurar las bibliotecas necesarias, y dejar la aplicación lista para su ejecución o distribución. Gracias a ello, la aplicación se ha compilado correctamente, lo que permite que se ejecute en el emulador Android sin problemas.

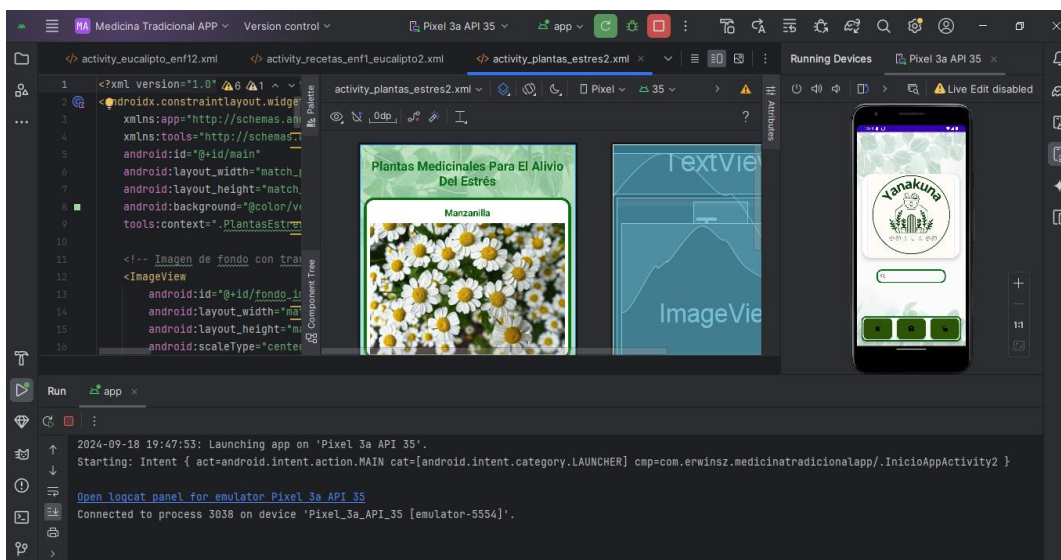


Figura 64: Emulador 1 Android

En la siguiente imagen se presenta la consola Logcat funcionando correctamente, la cual permite identificar y solucionar los problemas con la aplicación, mejora la calidad del código y garantiza una mejor experiencia para los usuarios.

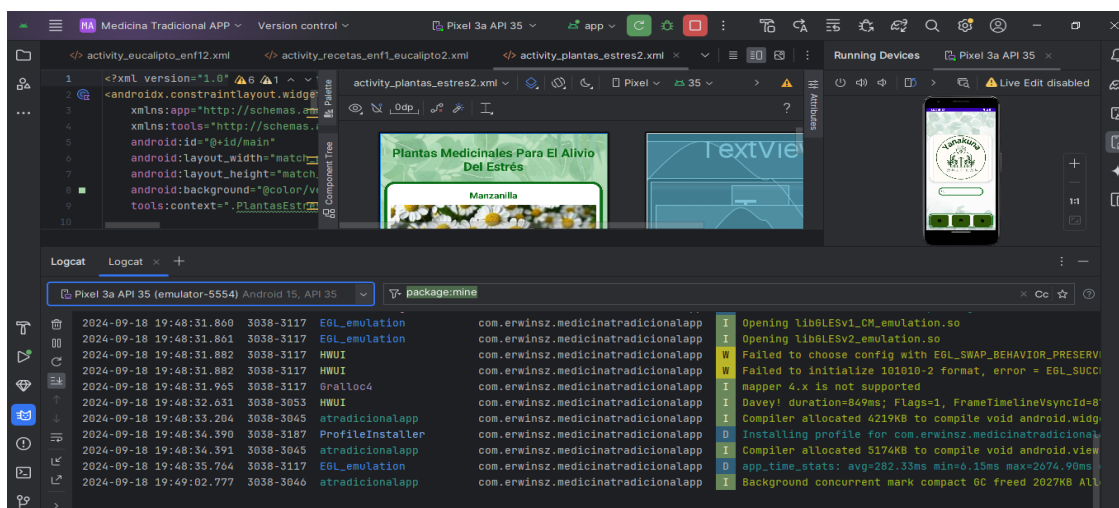


Figura 65: Consola Logcat

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 128 de 134

A continuación, se muestran imágenes de las interfaces ejecutándose en el emulador Android, donde se verifica que la navegación es correcta. Se ha comprobado el funcionamiento adecuado de los botones, así como la presentación de la información sobre las plantas, las imágenes y las recetas. Además, se destaca que la aplicación ofrece respuestas rápidas al pasar de una interfaz a otra, lo que mejora la experiencia del usuario y garantiza un uso fluido y eficiente de la aplicación.

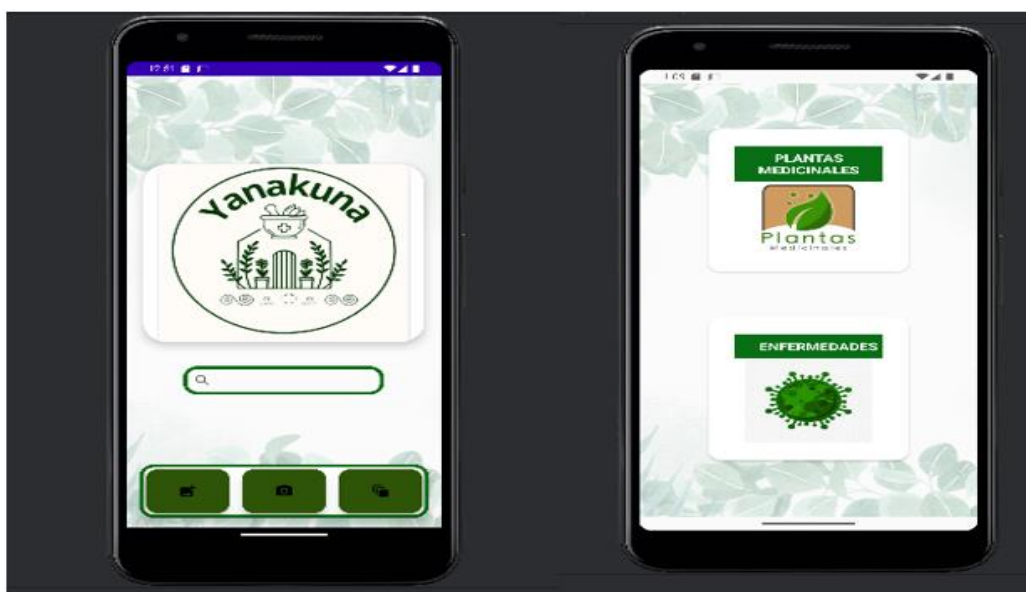


Figura 66: Emulador 2 Android

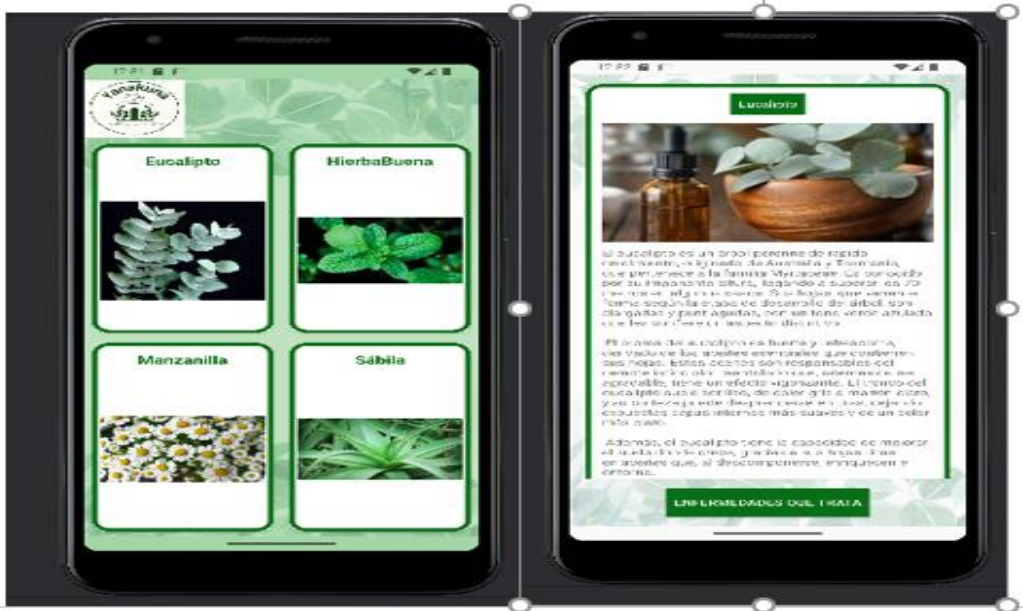


Figura 67: Emulador 3 Android

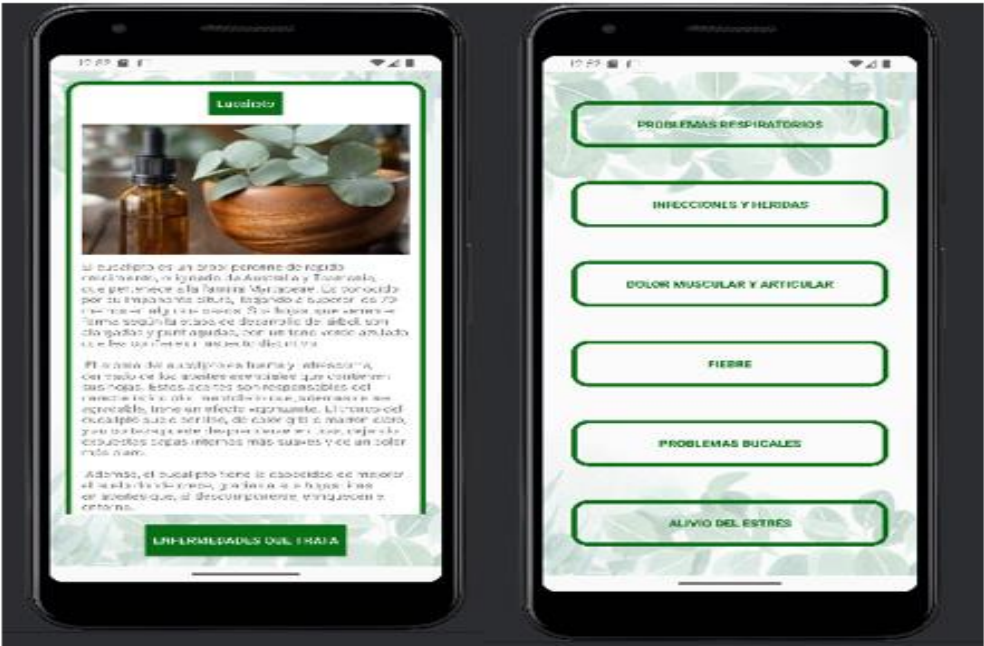


Figura 68: Emulador 4 Android



Figura 69: Emulador 5 Android

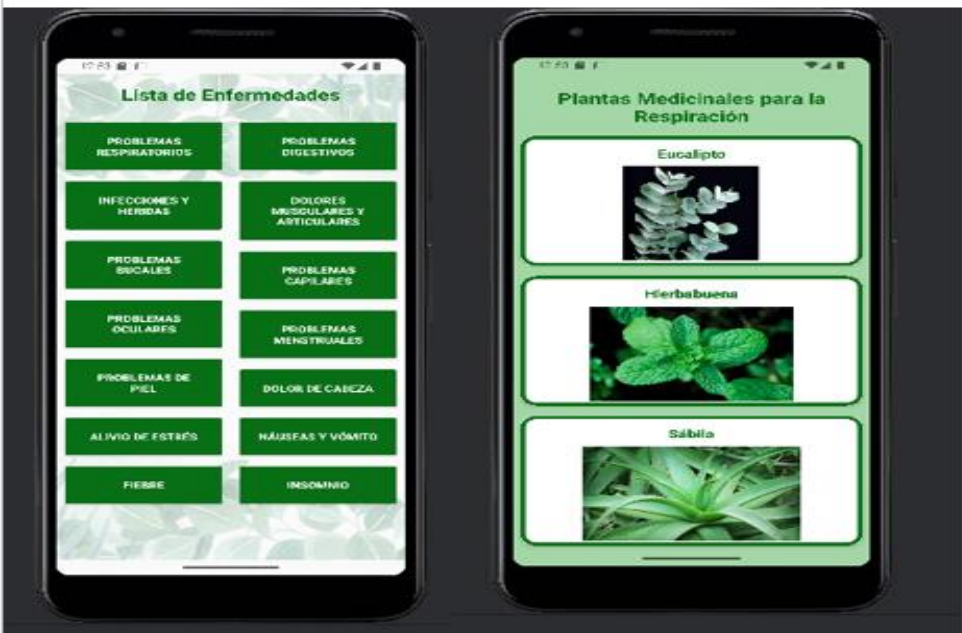


Figura 70: Emulador 6 Android

5.5. Pruebas integrales de funcionamiento backend y frontend

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 131 de 134

A continuación, se muestra las pruebas de funcionamiento de la aplicación integrando el backend con el frontend para ello se siguen los siguientes pasos:

Paso 1: se debe abrir el navegador de conda, elegir el enviromenst que se le ha creado para desarrollar el backend. En este proyecto se creó un entorno virtual llamado envplantas.

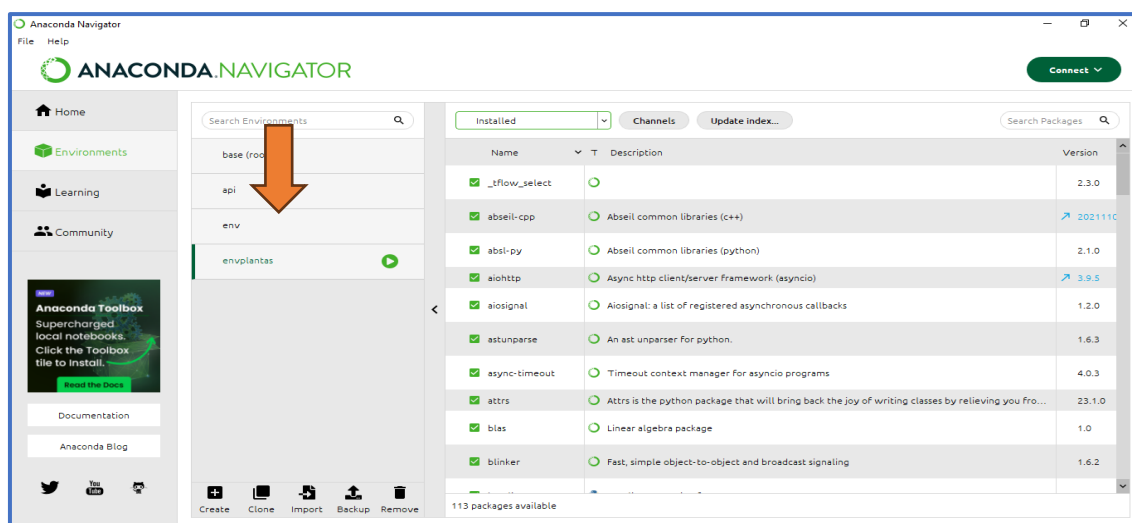


Figura 71: Navegador 1 anaconda

Luego se elige Visual Studio para ver los archivos de entrenamiento. predicción y la api o servidor.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 132 de 134

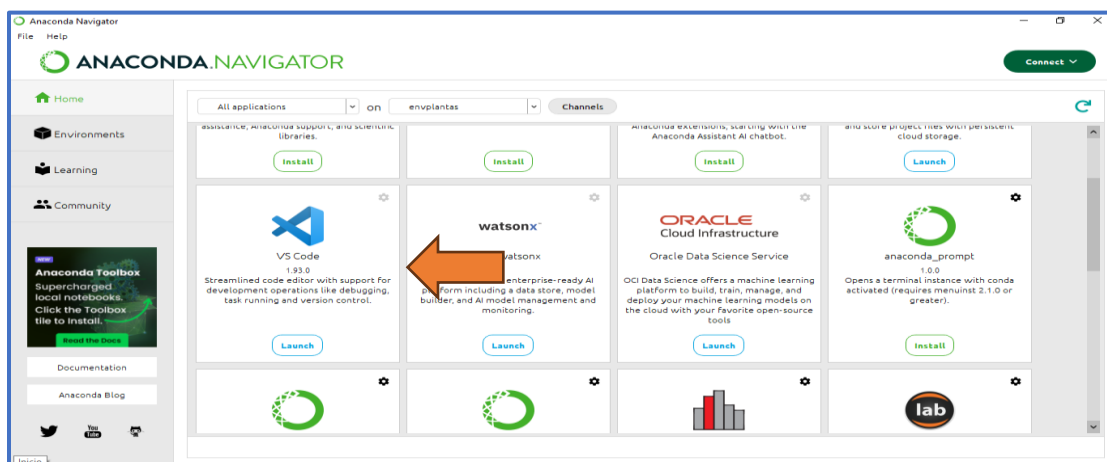


Figura 72: Navegador 2 anaconda

Al tener el entorno de desarrollo de Visual Studio abierto, junto con los archivos de entrenamiento, predicción y el archivo del servidor API, se debe abrir la consola del archivo api.py para ejecutar el siguiente comando: `flask --app api.py run --host=0.0.0.0`

Este comando permitirá establecer la conexión con el frontend y habilitará el consumo de la API desde la aplicación móvil. De esta manera, se garantiza que la aplicación pueda enviar solicitudes a la API y recibir respuestas, lo que es fundamental para el funcionamiento correcto de la app en su totalidad.

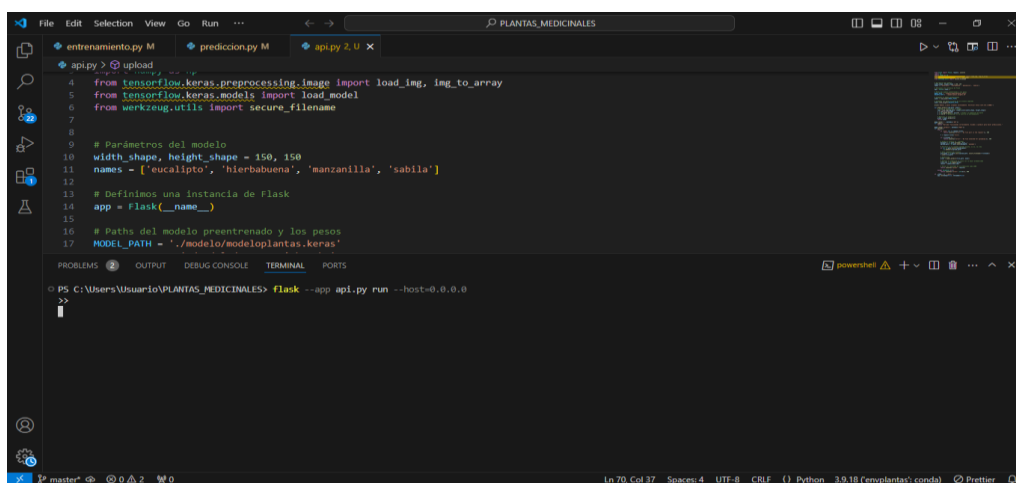


Figura 73: conexión de frontend con backend

En la siguiente imagen se tiene el servidor corriendo y listo para usar con Android.

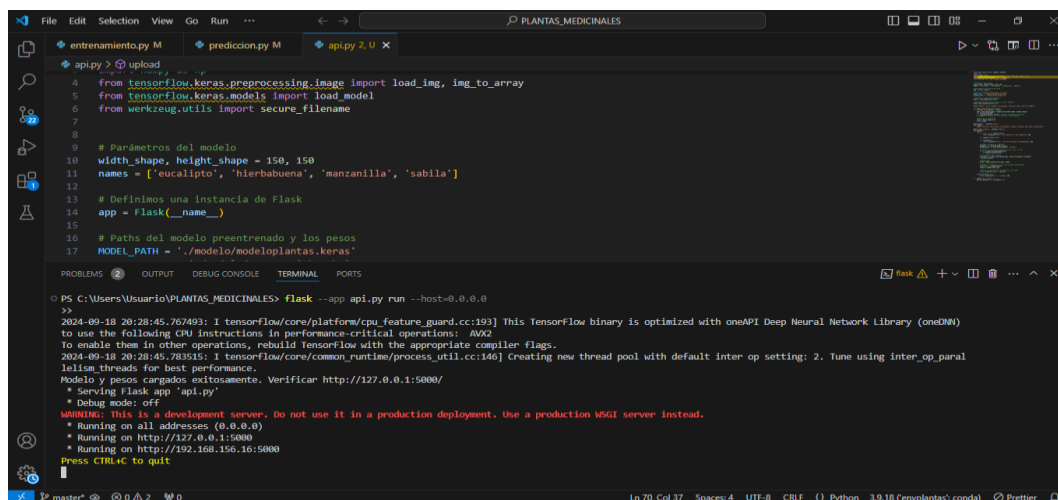


Figura 74: Conexión de servidor establecida

Por la parte de Android se utiliza okhttp3 que es una librería HTTP que facilita las solicitudes de red. Se utiliza para enviar peticiones y recibir respuestas desde un servidor, permitiendo la conexión con el servidor como lo muestra la imagen (figura 95: Conexión de servidor establecida)

Luego se conecta un móvil físico para realizar las pruebas y ver correctamente el funcionamiento y se procede a ejecutar la aplicación:

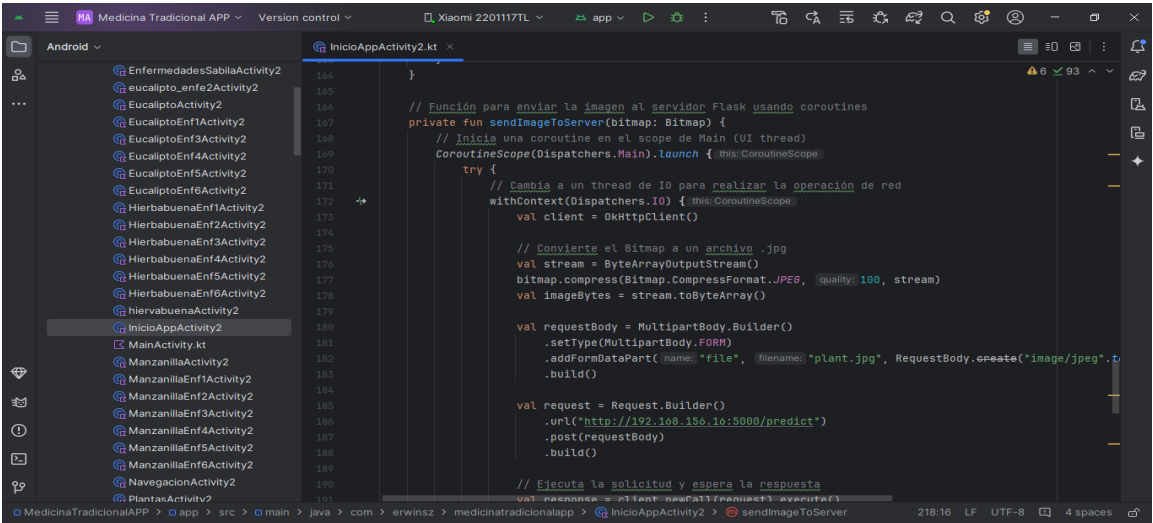


Figura 75: Prueba de Conexión servidor con Android

Se puede visualizar en la siguiente imagen que compilo el gradle el asistente encargado de organizar y automatizar todas las tareas necesarias para el correcto funcionamiento de las dependencias correctamente.

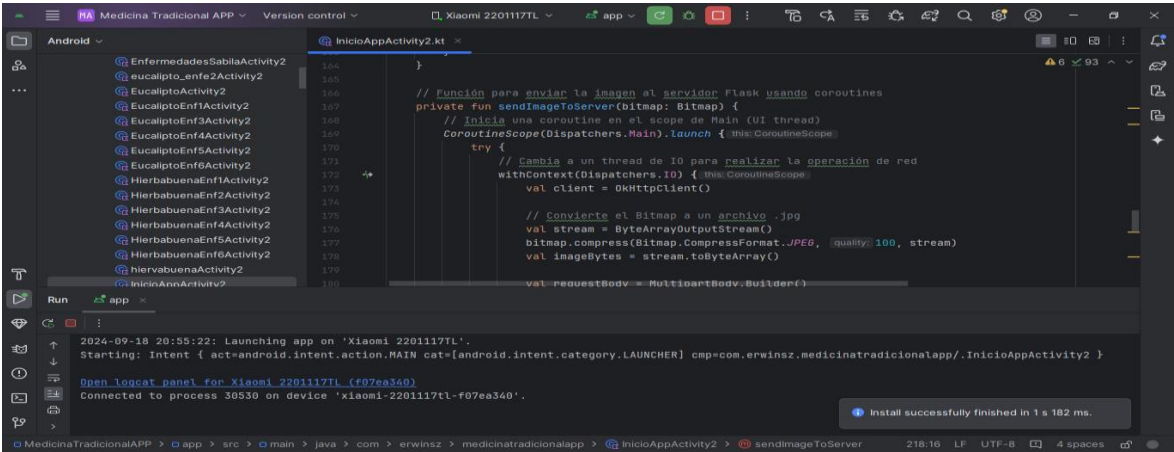


Figura 76: consola 1 gradle

De igual forma se visualiza que la consola Logcat compila correctamente.

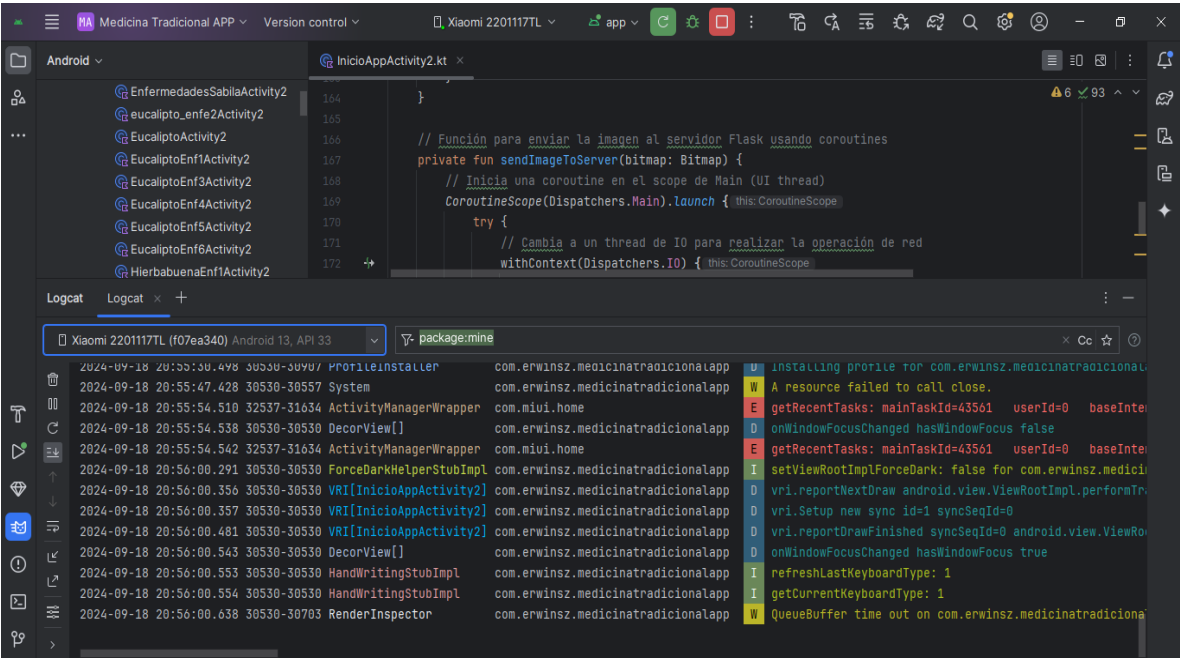


Figura 77: consola 2 gradle

En el móvil funcionando correctamente y se ve la interfaz principal de la aplicación.



Figura 78: prueba ejecución 1 en móvil

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 136 de 134

Con la aplicación se puede ingresar a la galería del móvil y seleccionar la imagen de la hierbabuena.

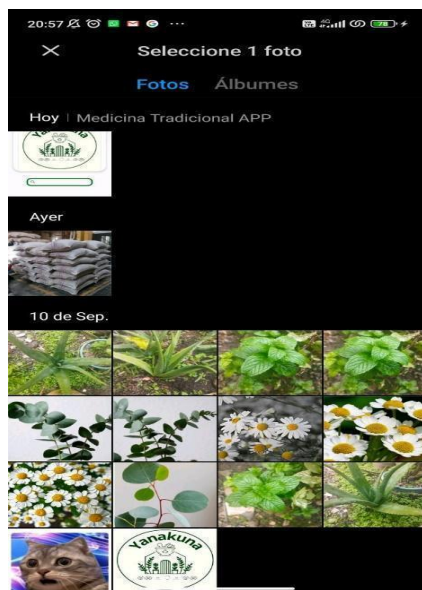


Figura 79: prueba ejecución 2 en móvil

Al seleccionar la imagen de hierbabuena, la pantalla vuelve al inicio, activando el botón "Enviar" mientras que los tres botones iniciales ("Galería", "Cámara" y "Menú") se ocultan. Una vez que la imagen se ha enviado al servidor, los botones ocultos vuelven a hacerse visibles.



Figura 80: prueba ejecución 3 en móvil

Al enviar la imagen, la aplicación redirige al usuario a la sección correspondiente a la planta que clasificó el servidor, mostrando la información relevante. En esta interfaz, se presenta el nombre de la planta junto con una imagen de su preparación. Además, se incluye una descripción detallada que destaca sus características. Finalmente, se proporciona un botón que dirige a los usuarios a la sección relacionada con las enfermedades que puede tratar la planta clasificada, facilitando así el acceso a información adicional y útil.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 138 de 134



Figura 81: prueba ejecución 4 en móvil

Con la planta identificada, se permite seleccionar de un listado de enfermedades posibles que pueden ser tratadas con dicha planta, mostrando la siguiente interfaz.



Figura 82: Figura 105: prueba ejecución 5 en móvil

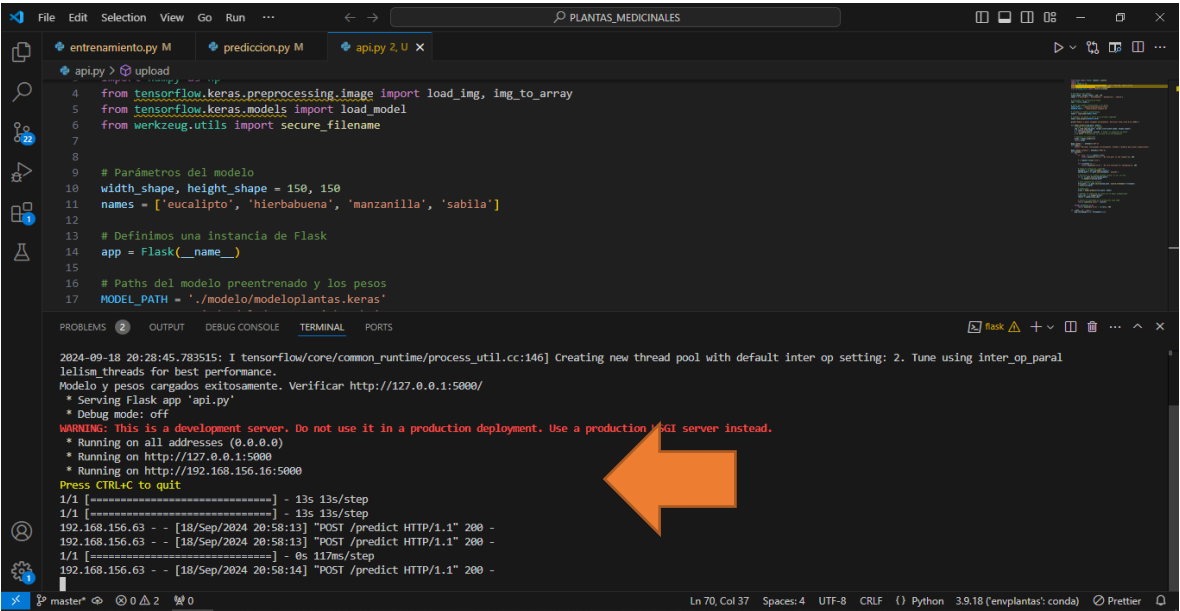
	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 139 de 134

Para esta prueba se eligió problemas respiratorios y la aplicación muestra la receta, ingredientes y usos de dicha planta en la medicina tradicional.



Figura 83: prueba ejecución 6 en móvil

Por el lado del servidor se puede observar que en la consola se va mostrando cada imagen recibida y su procesamiento.



```
File Edit Selection View Go Run ... PLANTAS_MEDICINALES
entrenamiento.py M prediccion.py M api.py 2, U X
api.py > upload
4 from tensorflow.keras.preprocessing.image import load_img, img_to_array
5 from tensorflow.keras.models import load_model
6 from werkzeug.utils import secure_filename
7
8
9 # Parámetros del modelo
10 width_shape, height_shape = 150, 150
11 names = ['eucalipto', 'hierbabuena', 'manzanilla', 'sabila']
12
13 # Definimos una instancia de Flask
14 app = Flask(__name__)
15
16 # Paths del modelo preentrenado y los pesos
17 MODEL_PATH = './modelo/modelo plantas.keras'
18
19
20 2024-09-18 20:28:45.783515: I tensorflow/core/common_runtime/process_util.cc:146] Creating new thread pool with default inter op setting: 2. Tune using inter_op_parallelism_threads for best performance.
21 I tensorflow/core/common_runtime/process_util.cc:146] Creating new thread pool with default inter op setting: 2. Tune using inter_op_parallelism_threads for best performance.
22 Modelos y pesos cargados exitosamente. Verificar http://127.0.0.1:5000/
23 * Serving Flask app 'api.py'
24 * Debug mode: off
25 WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
26 * Running on all addresses (0.0.0.0)
27 * Running on http://127.0.0.1:5000
28 * Running on http://192.168.156.16:5000
29 Press CTRL+C to quit
30 1/1 [=====] - 13s 13s/step
31 1/1 [=====] - 13s 13s/step
32 192.168.156.63 - - [18/Sep/2024 20:58:13] "POST /predict HTTP/1.1" 200 -
33 192.168.156.63 - - [18/Sep/2024 20:58:13] "POST /predict HTTP/1.1" 200 -
34 1/1 [=====] - 8s 117ms/step
35 192.168.156.63 - - [18/Sep/2024 20:58:14] "POST /predict HTTP/1.1" 200 -
36
37 master* 0 0 2 0
38 Ln 70, Col 37 Spaces: 4 UTF-8 CRLF Python 3.9.18 (env: plantas: conda) Prettier
```

Figura 84: consola servidor ejecutando app desde móvil

También se realiza la prueba capturando la imagen con la cámara y la aplicación permite navegar en ella después de la clasificación de la imagen enviada igualmente que seleccionándola de la galería.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 141 de 134

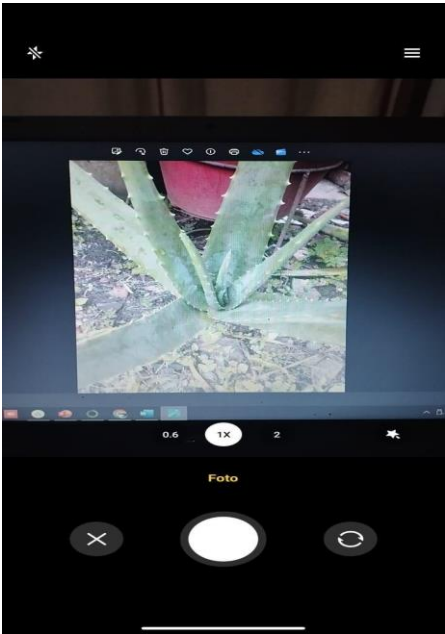


Figura 85: prueba ejecución 7 en móvil



Figura 86: prueba ejecución 8 en móvil

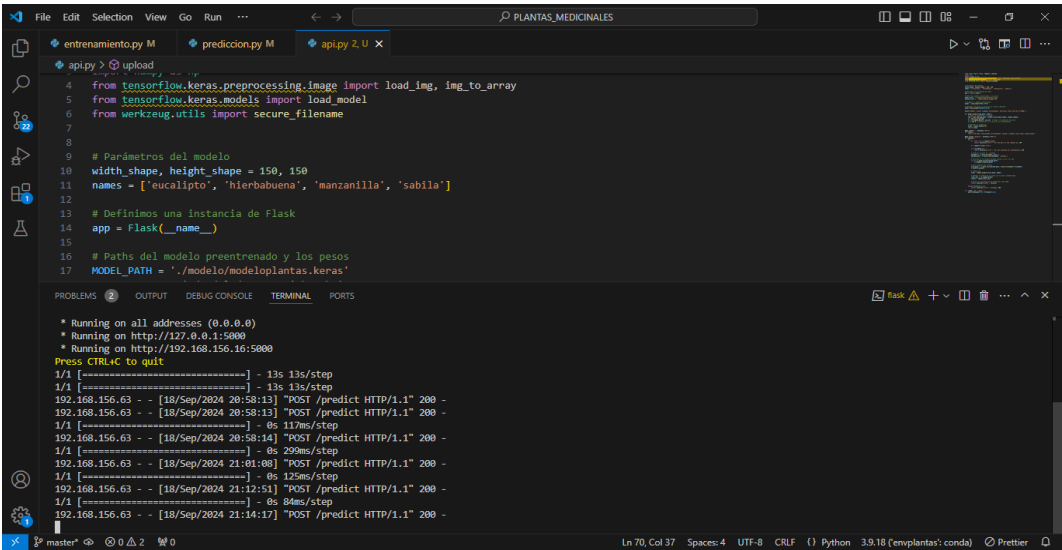


Figura 87: prueba ejecución 9 en móvil



Figura 88: prueba ejecución 10 en móvil

Y de igual manera en consola del servidor se ve el funcionamiento correcto.



```
File Edit Selection View Go Run ... PLANTAS_MEDICINALES
entrenamiento.py M predicción.py M api.py 2.0 x
api.py > upload
4 from tensorflow.keras.preprocessing.image import load_img, img_to_array
5 from tensorflow.keras.models import load_model
6 from werkzeug.utils import secure_filename
7
8
9 # Parámetros del modelo
10 width_shape, height_shape = 150, 150
11 names = ['eucalipto', 'hierbabuena', 'manzanilla', 'sábila']
12
13 # Definimos una instancia de Flask
14 app = Flask(__name__)
15
16 # Paths del modelo preentrenado y los pesos
17 MODEL_PATH = './modelo/modeloplantas.keras'
18
19 PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
20 * Running on all addresses (0.0.0.0)
21 * Running on http://127.0.0.1:5000
22 * Running on http://192.168.156.16:5000
23 Press CTRL+C to quit
24 1/1 [=====] - 13s 13s/step
25 1/1 [=====] - 13s 13s/step
26 192.168.156.63 - - [18/Sep/2024 20:58:13] "POST /predict HTTP/1.1" 200 -
27 192.168.156.63 - - [18/Sep/2024 20:58:13] "POST /predict HTTP/1.1" 200 -
28 1/1 [=====] - 0s 117ms/step
29 192.168.156.63 - - [18/Sep/2024 20:58:14] "POST /predict HTTP/1.1" 200 -
30 1/1 [=====] - 0s 290ms/step
31 192.168.156.63 - - [18/Sep/2024 21:01:08] "POST /predict HTTP/1.1" 200 -
32 1/1 [=====] - 0s 125ms/step
33 192.168.156.63 - - [18/Sep/2024 21:12:51] "POST /predict HTTP/1.1" 200 -
34 1/1 [=====] - 0s 86ms/step
35 192.168.156.63 - - [18/Sep/2024 21:14:17] "POST /predict HTTP/1.1" 200 -
Ln 70, Col 37 Spaces: 4 UTF-8 CRLF (Python 3.9.18 (enviplantas\conda) Prettier
```

Figura 89: prueba consola de servidor ejecución 9 en móvil

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 144 de 134

5.6. Validación

A continuación, se realiza las pruebas de usuario para realizar la validación de usabilidad del funcionamiento de la aplicación con personal de la comunidad Yanaconas y usuarios ajenos al pueblo usando el formato de evaluación heurística.

La evaluación heurística consiste en asignar una puntuación a una serie de preguntas relacionadas con principios heurísticos claves del funcionamiento de la aplicación [50]. Con la evaluación heurística se evalúa la usabilidad de la aplicación clasificando y organizando en tablas según el principio evaluado, algunos principios heurísticos evaluados son: principios generales, lenguaje y redacción, rotulación y organización, estructura y navegación, diseño de la interfaz, elementos multimedia, ayuda, accesibilidad y navegación. Cada ítem dentro de estos principios se puntúa de 1 a 5, donde 1 indica incumplimiento y 5 representa el cumplimiento óptimo.

Se realizó la prueba con seis usuarios, cada instaló la aplicación en su dispositivo móvil mediante una APK y realizó las diferentes actividades que esta permite. Posteriormente, completaron la evaluación heurística proporcionada para valorar su experiencia de uso.

Agregamos imágenes de algunos usuarios realizando las pruebas y un formato realizado, los demás formatos los pueden ver en un documento anexado con dichas evidencias.

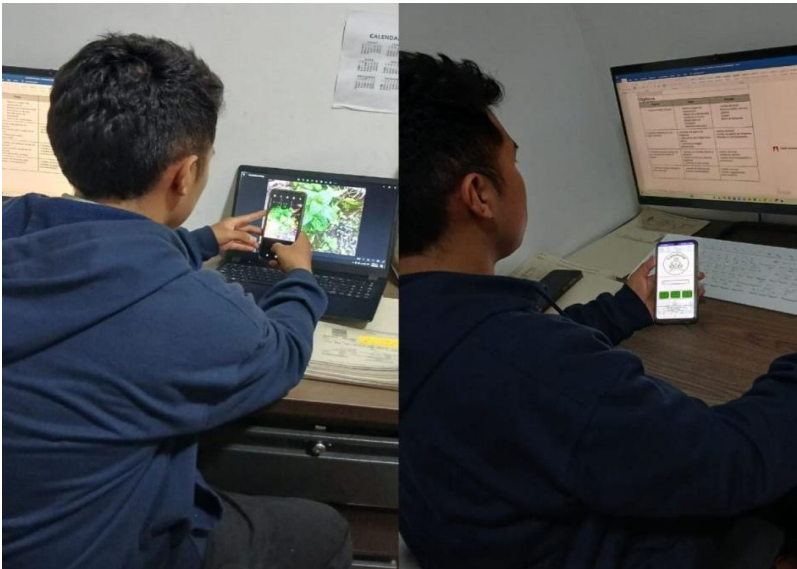


Figura 90: prueba usabilidad usuario 1



Figura 91: prueba usabilidad usuario 2

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 146 de 134

7.2 Análisis heurístico

A continuación, se presenta los resultados del análisis de usabilidad realizado sobre la aplicación móvil Yanakuna usando un formato de principios heurísticos (capítulo 5 documento de anexos).

Datos del análisis

Heurísticos específicos

[Tarea]	Puntos
[Generales]	33
[Identidad e información]	16
[Lenguaje y redacción]	20
[Rotulación y organización]	17
[Estructura y navegación]	41
[Layout de la página]	35
[Elementos multimedia]	20
[Ayuda]	5
[Accesibilidad y visualización]	30
[Control y retroalimentación]	23

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 147 de 134

Conclusiones

A modo de conclusiones se puede ver un mapa de los heurísticos controlados en esta evaluación, agrupados por categorías:

Heurísticos	Resultado	Observaciones
Generales	4.1	Cumple con los criterios de este principio, solamente queda pendiente este criterio que la app móvil se actualice periódicamente e Indique cuándo se actualiza.
Identidad e información	2.6	Pendiente agregar más información de la empresa desarrolladora y los derechos reservados.
Lenguaje y redacción	5	Cumple con los criterios.
Rotulado	4.2	Cumple con los criterios.
Estructura y navegación	4.1	Cumple con los criterios, aunque a la hora de gira la pantalla del dispositivo no se ajusta la app.
Layout de la página	5	Cumple con los criterios.
Búsqueda (en caso de ser necesaria)	5	No cumple con los criterios.
Elementos multimedia	5	Cumple con los criterios.
Ayuda	1.2	No cumple en su mayoría de criterios, pendiente agregar un asistente de ayuda para el uso de app.
Accesibilidad	5	Cumple con los criterios.
Control y retroalimentación	4.6	Cumple con los criterios.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 148 de 134

Propuestas de solución

A continuación, se muestran las propuestas de solución para mejorar la usabilidad de la aplicación considerando los elementos encontrados en la evaluación realizada, siguiendo los resultados obtenidos de la evaluación heurística¹.

#	Propuesta de solución
1	Subir a la tienda playStore de Android para que se puedan aplicar las actualizaciones de la app.
2	Agregar información de los creadores de la app, derechos reservados.
3	Implementar la configuración para que la app se ajuste a la pantalla del móvil cuando esta sea girada.
4	Agregar un asistente como ayuda para el uso de app.

5.7. Resultado de evaluación de utilidad de la aplicación

La evaluación de utilidad se centró en la experiencia de los usuarios al interactuar con la aplicación. Se diseñaron preguntas orientadas a identificar la percepción de utilidad, facilidad de uso, satisfacción general, y recomendaciones para mejorar la herramienta.

A continuación, se presenta un análisis basado en las respuestas obtenidas:

Los usuarios destacaron que la aplicación cumplió con sus expectativas en términos de identificación de plantas medicinales y el acceso a información de usos y recetas de las mismas.

¹ Se marcan con un prefijo **CLAVE** o con un color destacado en la celda aquellas propuestas que se consideran críticas para mejorar sensiblemente la aplicación, siendo recomendado que su implementación sea prioritaria.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 149 de 134

La rapidez del sistema para clasificar las plantas y la claridad de las recetas proporcionadas fueron mencionadas como puntos fuertes.

La mayoría de los usuarios indicaron que la interfaz es intuitiva y fácil de navegar. Los elementos visuales, como imágenes claras de las plantas y menús organizados, facilitaron la experiencia. Algunos usuarios mencionaron dificultades menores, como el manejo a la hora de enviar la imagen al servidor con un breve retardo.

Los usuarios apreciaron que la aplicación integre aspectos culturales al incluir usos y preparaciones tradicionales de las plantas medicinales.

Algunos usuarios sugirieron ampliar la cantidad de plantas disponibles para su reconocimiento e incluir información sobre plantas de difícil acceso en la región, y la posibilidad de guardar un historial de búsquedas o clasificaciones fue una de las funciones más solicitadas.

Conclusiones:

Basándonos en los resultados, se concluye que la aplicación cumple con su propósito de utilidad al facilitar el acceso al conocimiento sobre plantas medicinales de una manera ágil y culturalmente relevante. Sin embargo, se identificaron áreas de mejora, como la ampliación del catálogo de plantas y la incorporación de funciones complementarias que enriquezcan la experiencia del usuario.

Para finalizar algunas recomendaciones como implementar actualizaciones periódicas para incluir más especies de plantas, mejorar la guía para gestionar permisos en dispositivos móviles y explorar funcionalidades que incrementen la personalización, como la opción de crear perfiles o guardar consultas

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 150 de 134

Las encuestas realizadas para evaluar la utilidad de la aplicación móvil indican que este cumple con su objetivo de organizar e identificar plantas utilizadas en la medicina tradicional de la comunidad Yanacona del suroriente del departamento del Cauca. La aplicación proporciona información específica sobre cada planta y sus usos, lo cual fue valorado positivamente por los usuarios que participaron en la encuesta, quienes coincidieron en que la funcionalidad de identificación de plantas medicinales es precisa y efectiva.

Además, esta evaluación de utilidad permite a los usuarios expresar tanto sus acuerdos como desacuerdos respecto a las funcionalidades y la información presentada en la aplicación. También brinda la oportunidad de recibir sugerencias sobre otras funcionalidades e información que les gustaría encontrar al interactuar con la aplicación móvil.

VI. Capítulo

6.1. Conclusiones

A partir de la revisión de la literatura y los conocimientos ancestrales compartidos por los médicos tradicionales del pueblo Yanacona, se ha logrado establecer una visión integral del uso y clasificación de las plantas medicinales en esta comunidad. El análisis permitió identificar plantas esenciales, como la sábila, la manzanilla, el eucalipto y la hierbabuena, entre otras, valoradas por sus numerosas propiedades curativas. A pesar de los desafíos que enfrenta la transmisión del conocimiento entre generaciones, la relevancia cultural y medicinal de estas prácticas sigue siendo significativa. Además, se destaca la importancia de respetar los territorios

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 151 de 134

de donde provienen estas plantas, ya que su crecimiento depende de climas específicos que influyen en su eficacia para distintos tratamientos.

Como resultado del desarrollo de la aplicación móvil, se logró cumplir el objetivo de identificarlas cuatro plantas más comunes en la medicina tradicional de la comunidad del pueblo Yanacona para trabajar la aplicación, proporcionando información detallada sobre cada planta utilizada en la aplicación y sus posibles usos medicinales. La aplicación no solo facilita el reconocimiento de las plantas a través de imágenes, sino que también ofrece a los usuarios una herramienta educativa para preservar y difundir el conocimiento ancestral de la medicina tradicional Yanacona, contribuyendo a la valorización y conservación de estas prácticas.

La validación de usabilidad mediante pruebas heurísticas de la aplicación móvil en la comunidad del pueblo Yanacona ha demostrado que cumple con su objetivo principal: identificar plantas medicinales y proporcionar información sobre sus usos tradicionales. Los usuarios valoraron positivamente su funcionalidad y su potencial como herramienta educativa. Sin embargo, se reconocieron áreas de mejora que requieren adaptaciones adicionales para optimizar la experiencia de usuario y elevar la aplicación a un nivel de excelencia. Estos ajustes permitirán mejorar la interfaz, la usabilidad y la eficiencia, consolidando la aplicación para preservar el conocimiento tradicional.

A partir de los resultados obtenidos en la encuesta de utilidad de la aplicación móvil, realizada por nueve usuarios de la comunidad Yanacona, se identifican varias mejoras claves que se esperan implementar en futuros desarrollos, sin embargo, se determina que es una aplicación útil para la comunidad y que podría servir a preservar el conocimiento en este contexto. Estas mejoras buscan confirmar que la aplicación cumple con el objetivo general de este trabajo

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 152 de 134

de investigación, que es crear un software para organizar e identificar las plantas utilizadas en la medicina tradicional de la comunidad Yanacona del suroriente del departamento del Cauca.

La aplicación tiene como propósito brindar información detallada sobre cada planta, incluyendo sus usos y beneficios específicos, de forma que se respete y preserve el conocimiento propio de la comunidad sobre su flora medicinal. Con estas mejoras, se busca fortalecer la utilidad de la aplicación, asegurando que responda de manera efectiva a las necesidades culturales y prácticas de sus usuarios.

6.2. Trabajos futuros

Considerando las respuestas obtenidas en la encuesta de utilidad realizada a los usuarios de la comunidad Yanacona, se identificaron varias oportunidades de mejora para el futuro. Entre los objetivos principales se encuentra la implementación del dataset más robusto que incluya un mayor número de plantas medicinales, ampliando así el repertorio de conocimientos accesibles. Además, se propone el rediseño de interfaces que reflejan de manera más fiel la cultura del pueblo Yanacona, con el fin de fortalecer la identidad visual y cultural de la aplicación.

Otro aspecto relevante es complementar la información con videos, preferiblemente protagonizados por miembros de la comunidad que posean un conocimiento profundo de la medicina tradicional.

Además, se sugiere agregar más funcionalidades, como una opción de regreso al inicio al llegar a la interfaz de recetas y preparaciones, así como la barra de búsqueda permita realizar búsquedas dentro de la aplicación con exactitud.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 153 de 134

Finalmente, se sugiere implementar una función de historial para registrar las plantas consultadas, lo que permitiría a los usuarios un seguimiento más personalizado de su interacción con la aplicación

VII. Bibliografía

- [1] C.López, «La reconstrucción de la casa yanacona. Etnicidad y transformación del espacio social en el macizo colombiano,» *Virajes, Revista de Antropología y Sociología*, pp. 45-53, 1999..
- [2] C. J, «El ciclo vital del runa yanacona desde su saber ancestral,» 2020.
- [3] ONIC, «Yanacona-Yanakuna,» s.f. . [En línea]. Available: <https://www.onic.org.co/pueblos/1162-yanaconas#:~:text=UBICACI%C3%93N%20GEOGR%C3%81FICA,Valle%20del%20Cauca%20y%20Huila.> [Último acceso: 19 marzo 2023].
- [4] M. O. y. D. V. A. Bermúdez, «La investigación etnobotánica sobre plantas medicinales, una revisión de sus objetivos y enfoques actuales,» *Interciencia*, vol. 30, n° 8, pp. 453-459, 2005.
- [5] Organización Mundial de la Salud 2013, «Estrategia de la OMS sobre la medicina tradicional 2014-2023,» 2013.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 154 de 134

- [6] E. P. C. J. S. S. J. J. Velasco, «Las plantas medicinalesel legado del pueblo Misak,» *agendapropia*.
- [7] f. d. C. y. A. y. I. R. C. d. I. U. S. Medellín, «Desafíos y proyecciones del uso de plantas medicinales en Colombia,» *universidad nacional de colombia*.
- [8] Y. P. I. M. I. C. y. E. G. D. Pascual, «Algunas consideraciones sobre el surgimiento y la evolución de la medicina natural y tradicional,» *Medisan*, vol. 18, nº 10, pp. 1467-1474, 2014.
- [9] Discordies, «Demateria medica,Español,» *En Anuers en casa de Iuan Latio*, 1555..
- [10] S. H. Jaén, «Formalización del concepto de salud a través de la lógica: impacto del lenguaje formal en las ciencias de la salud,» 2016.
- [11] C. V. Zambrano, «Etnicidad y cambio cultural entre los yanaconas del Macizo Colombiano,» 1995. .
- [12] D. P. M. C. A. R. y. L. M. E. Duperet, «Importancia de los reposoitorios para preservar y recuperar la información,» 2015. [En línea]. Available: <http://scielo.sld.cu/pdf/san/v19n10/san141910.pdf>. [Último acceso: 19 marzo 2024].
- [13] L. A. P. H. E. S. H. A. U. V. J. M. Quiñone, «Aplicaciones móviles en salud: usos frecuentes y potenciales,» *repositorio*.
- [14] M. López, «Qué son las Apps y tipos de Apps,» Universidad Tecnológica de Pereira.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 155 de 134

- [15] b. lekhi, 2020. [En línea]. Available:
<https://www.culturalsurvival.org/es/publications/cultural-survival-quarterly/la-perdida-de-conocimientos-tradicionales-se-debe-la-falta>. [Último acceso: 18 octubre 2024].
- [16] W. G. I. Daza, 2022. [En línea]. Available:
<https://www.redalyc.org/journal/5258/525871894009/>. [Último acceso: 18 octubre 2024].
- [17] W. COLOMBIA, «os pueblos indígenas y su conocimiento ancestral son clave para la conservación de la biodiversidad,» Bogotá-Colombia, 2023.
- [18] F. M. Machaca, «Los conocimientos ancestrales son parte de las respuestas ante las crisis actuales,» *El país*, 08 septiembre 2022. .
- [19] R. R. ., S. R. ., J. S. ., S. R. ., K. S. ., T. A. ., G. S. ., C. L. ., C. A. ., R. B. ., D. K. Srikanth N, «Aplicación móvil: uso informado de la medicina tradicional para el mantenimiento de la salud en la India durante la pandemia de COVID-19: estudio de cuestionario transversa,» *JMIR Publications is the leading open access digital health research publisher..*
- [20] G. L. y. R. B. J. Filippi, «las aplicaciones móviles y su impacto en la sociedad,» *universidad y sociedad*, vol. 16, nº 2, pp. 237-243, 2022.
- [21] C. Muñoz, «Diseño y desarrollo de un aplicativo móvil en la plataforma Android Studio para la Empresa Allied Electronics & Service (Servicios Electrónicos Aliados),» perira,colombia.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 156 de 134

- [22] R. Garita, «Tecnología Móvil: desarrollo de sistemas y aplicaciones para las Unidades de Información,» *Revista e- Ciencias de la Información*, vol. 3, nº 2, pp. 1-14, 2013.
- [23] M. d. J. Pérez-Van-Leenden, «La investigación acción en la práctica docente,» *Revista Internacional de Investigación en Educación*, 2019.
- [24] A. K. a. S. P. A. Gulli, «Deep Learning with Tensorflow 2 and Keras,» 2019. .
- [25] Organización Mundial de la Salud, «Estrategia de la OMS sobre medicina tradicional,» 2002-2005. [En línea]. Available:
https://iris.who.int/bitstream/handle/10665/67314/WHO_EDM_TRM_2002.1_spa.pdf?sequence=1&isAllowed=y. [Último acceso: 20 octubre 2024].
- [26] E. J. B. O. C. Ospina Lozano, «Autoatención en salud oral en el pueblo yanacona de los andes del sur de Colombia,» *repositorio institucional UNAL*, 2011.
- [27] J. S. y. M. Cruz, «Repositorios institucionales digitales,» *e-ciencias de la información*, vol. 71, pp. 1-32, 2017. .
- [28] J. Miyahira, «Repositorios Digitales,» PUCP, Lima, Perú, 2012..
- [29] Mathworks, 1994-2023. [En línea]. Available: <https://la.mathworks.com/discovery/image-recognition-matlab.html>.
- [30] Organización Mundial de la Salud, «Estrategia de la OMS sobre medicina tradicional 2014-2023,» *OMS*, 2013.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 157 de 134

- [31] UNESCO, «Patrimonio cultural inmaterial,» 2021. [En línea]. Available:
<https://es.unesco.org/themes/patrimonio-cultural-inmaterial#:~:text=El%20patrimonio%20cultural%20inmaterial%20o,comunidades%20de%20generaci%C3%B3n%20en%20generaci%C3%B3n..> [Último acceso: 23 marzo 2023].
- [32] universidad de los andes, «Patrimonio cultural inmaterial de colombia,» [En línea]. Available: <https://facartes.uniandes.edu.co/patrimonio/inmaterial/manifestaciones/>. [Último acceso: 23 Marzo 2023].
- [33] M. L. M. Benavidez, «Caminando la memoria yanakuna desde la experiencia al retorno al runa shimi,» Popayán-cauca-colombia, 2023.
- [34] D. P. C. M. A. R. y. L. M. E. Duperet, «Importancia de los repositorios para preservar y recuperar la información,» *MEDISAN*, vol. 19, nº 10, pp. 1-8, 2015.
- [35] A. D. P. y. c. H. M. Doria, «Repositorios de información como recurso de información académica de universidades,» 2013..
- [36] M. M. y. A. S. T. G, «Repositorios de acceso abierto en la instituciones de educación superior en México,» *información,cultura y sociedad*, nº 40, pp. 117-130, 2019.
- [37] L. P. D. S. y. J. U. J. Melo, «aplicaciones moviles en salud: uso frecuentes y potenciales riesgos».

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 158 de 134

- [38] G. S. y. E. Hernández, «aplicaciones médicas móviles : definiciones, beneficios y riesgos,» *salud uninorte*, pp. 599-607, 2015.
- [39] A. Fernández, «Aplicaciones móviles relacionadas con la salud,» zaragoza, 2020..
- [40] J. Vásquez, «aceptación tecnológica de aplicaciones móviles para el servicio de la salud a domicilio,» 2019.
- [41] C. H. D. &. R. S. C. Flórez, *procesamiento de imagenes para reconocimiento de daños causados por plagas*, , 2015.
- [42] c. A. V. Fonseca, «software para identificar y localizar diversas variedades de plantas de quinua en el distrito de Querataro,» 2019.
- [43] G. D. C. González, «Detección de enfermedades en plantas de tomate a través del análisis computacional de sus hojas,» . , 2021.
- [44] ica, «plantas aromáticas y medicinales enfermedades y usos terapeuticos,» *repositorio*. .
- [45] G. B. L. V. Pete Deemer, «información básica de scrum,» 2013.
- [46] T. Dimes, «Conceptos Básicos de Scrum: Desarrollo de software Agile y manejo de proyectos Agile.,» 2015.
- [47] R. Shashidhar, Detection of choroidal neovascularization (CNV) in retina OCT images using VGG16 and densenet CNN, *Wireless Personal Communications*, 2022, p. 2569–2583.

	TRABAJO DE GRADO	EDO – 01
		Versión 1
		Vigencia: 10/10/2024
		Página 159 de 134

- [48] A. Developers, «Android Developers,» [En línea]. Available:
<https://developer.android.com/kotlin/coroutines?hl=es-419>. [Último acceso: 20 octubre 2024].
- [49] M. Cárdenas-Montes, «Sobreajuste–overfitting. Ciemat,» Centro de Investigaciones Energéticas, Medioambientales y Tecnológicas,,» 2006. [En línea]. Available:
https://gc.scalahed.com/recursos/files/r161r/w24082w/R5_01.pdf. [Último acceso: 20 octubre 2024].
- [50] M. P. P. A. & L. J. González, « Introducción a la Interacción Persona-Ordenador,» 2001. [En línea].