

**IACULTIVAR: APLICACIÓN MÓVIL CON RECONOCIMIENTO DE IMÁGENES
PARA LA DETECCIÓN Y PREVENCIÓN DE LA MONILLA EN EL CULTIVO DEL
CACAO**

**ANDRES CAMILO RINCON CABALLERO
MARYURY JULIETH RODRIGUEZ CALDERON**

**UNIVERSIDAD AUTÓNOMA DE BUCARAMANGA
FACULTAD INGENIERÍA DE SISTEMAS
TECNOLOGÍA Y SOCIEDAD
BUCARAMANGA**

2020

IACULTIVAR: APLICACIÓN MÓVIL CON RECONOCIMIENTO DE IMÁGENES
PARA LA DETECCIÓN Y PREVENCIÓN DE LA MONILIA EN EL CULTIVO DEL
CACAO

ANDRES CAMILO RINCON CABALLERO

MARYURY JULIETH RODRIGUEZ CALDERON

PROYECTO DE GRADO PRESENTADO PARA OPTAR AL TÍTULO DE
INGENIERO DE SISTEMAS

DIRECTOR

JUAN SEBASTIAN CARDENAS ARENAS

INGENIERO DE SISTEMAS

UNIVERSIDAD AUTÓNOMA DE BUCARAMANGA

FACULTAD INGENIERÍA DE SISTEMAS

TECNOLOGIA Y SOCIEDAD

BUCARAMANGA

2020

AGRADECIMIENTOS

Los autores expresan sus agradecimientos a:

Agradecemos a nuestro tutor de tesis el profesor Juan Sebastián Cárdenas quien nos ha acompañado en el proceso y desarrollo de este proyecto, aportando sus ideas y puntos de vista, apoyando los nuestros, expresamos agradecimientos a el profesor Leonardo Talero y los estudiantes Sebastián Serrano, Camilo Torres quienes muy amablemente nos compartieron el set de imágenes necesario para el reconocimiento de imágenes, agradecemos a la universidad por permitirnos

CONTENIDO	
LISTA DE TABLAS	7
LISTA DE FIGURAS	8
LISTA DE ANEXOS	10
RESUMEN	11
1. INTRODUCCIÓN	12
1.1 PLANTEAMIENTO DE LA PROBLEMATICA	12
2. OBJETIVO GENERAL	14
2.1. OBJETIVOS ESPECÍFICOS	14
3. ESTADO DEL ARTE	15
4. MARCO CONCEPTUAL	23
5. METODOLOGÍA	38
6. ANALISIS DE REQUERIMIENTOS	41
6.1 REDACCIÓN Y DEFINICIÓN DE HERRAMIENTAS, METODOLOGÍAS Y PREGUNTAS PARA LAS ENCUESTAS Y ENTREVISTAS	42
6.2 ENCUESTAS Y ENTREVISTAS A AGRICULTORES Y ENTES AGRICULTORES	43
6.3 SEGMENTACIÓN DE DESEOS Y NECESIDADES	43
6.4 PLANTEAMIENTO DE REQUERIMIENTOS FUNCIONALES Y NO FUNCIONALES	44
7. DISEÑO DIAGRAMAS, WIREFRAMES E INTERFACES	47
7.1 DISEÑO DE DIAGRAMAS DE CASOS DE USO Y DIAGRAMAS DE CLASES	48
7.2 DISEÑO DIAGRAMA DE BASE DE DATOS	52
7.3 DISEÑO DEL DIAGRAMA DE ARQUITECTURA	53
7.4 CREACIÓN DE WIREFRAMES E INTERFACES	55
8. DESARROLLO DE ADMIN WEB, API REST, ENTRENAMIENTO DE LA IA Y APPLICACIÓN MÓVIL	58
8.1 DESARROLLO DE ADMIN WEB Y API REST	58
8.2 ENTRENAMIENTO DEL MÓDULO DE RECONOCIMIENTO DE IMÁGENES	66
8.3 DESARROLLO DE LA APLICACIÓN MÓVIL	71
8.4 INTEGRACIÓN DEL SISTEMA	81
9. FASE DE PRUEBAS Y COMPROBACIÓN DEL SISTEMA	83
9.1 DISEÑO DE ESCENARIOS DE PRUEBAS	83

9.2 EJECUCIÓN DE LOS ESCENARIOS DE PRUEBAS	86
9.3 ANÁLISIS DE LOS RESULTADOS DE LOS ESCENARIOS DE PRUEBAS	88
CONCLUSIONES	92
RECOMENDACIONES O TRABAJOS FUTUROS	93
REFERENCIAS BILIOGRÁFICAS	94
ANEXOS	98

LISTA DE TABLAS

Tabla 1 Articulo 1	15
Tabla 2 Articulo 2	15
Tabla 3 Articulo 3	16
Tabla 4 Articulo 4	16
Tabla 5 Articulo 5	17
Tabla 6 Articulo 6	17
Tabla 7 Articulo 7	18
Tabla 8 Articulo 8	18
Tabla 9 Articulo 9	19
Tabla 10 Articulo 10	19
Tabla 11 Articulo 11	19
Tabla 12 Articulo 12	20
Tabla 13 Articulo 13	21
Tabla 14 Articulo 14	21
Tabla 15 Articulo 15	22
Tabla 16 Teachable machine VS IBM Watson	68
Tabla 17 Tabla precisión tasa de aprendizaje 0,001	69
Tabla 18 Tabla precisión tasa de aprendizaje 0,0001	70
Tabla 19 Ejemplo prueba	83
Tabla 20 Resumen de las pruebas a realizar	85
Tabla 21 Resultados pruebas	86

LISTA DE FIGURAS

Figura 1 Fruto de cacao tipo criollo	24
Figura 2 Fruto de cacao tipo forastero.....	25
Figura 3 Clon de cacao tipo híbrido "clon san Vicente 155".....	26
Figura 4 Ciclo de vida de la monilia	27
Figura 5 Ciclo de vida de la enfermedad escoba de bruja	28
Figura 6 Modelo entidad relación	31
Figura 7 Modelo relacional relación	32
Figura 8 API Rest.....	33
Figura 9 Red neuronal.....	35
Figura 10 Sprint	38
Figura 11 Casos de uso admin web.....	48
Figura 12 Casos de uso aplicación móvil.....	49
Figura 13 Diagrama de clases IAcultivar.....	50
Figura 14 Explicación clases principales.....	51
Figura 15 Diagrama de bases de datos.....	52
Figura 16 Explicación relaciones muchos a muchos.....	53
Figura 17 Vista general de la arquitectura	54
Figura 18 Diagrama de despliegue	55
Figura 19 Interfaces móviles	56
Figura 20 Interfaces móviles	57
Figura 21 Interfaces móviles	57
Figura 22 Explicación estructura proyecto Django	59
Figura 23 Modelos Django.....	60
Figura 24 Menú principal web admin	61
Figura 25 Listado preguntas web admin	62
Figura 26 formulario cultivo web admin.....	62
Figura 27 Serializers Django	63
Figura 28 ViewSets Django	63
Figura 29 Routers Django.....	64
Figura 30 API Root.....	65
Figura 31 JSON cultivo	65
Figura 32 Teachable machine entrenamiento	66
Figura 33 IBM Watson entrenamiento	67
Figura 34 Directorio de imágenes para entrenamiento.....	69
Figura 35 Matriz de confusiones mejor modelo	71
Figura 36 Comandos flutter	72
Figura 37 Extensiones VScode	73
Figura 38 Explicación directorio aplicación móvil.....	73
Figura 39 Código e interfaz aplicación.....	74
Figura 40 función transformación JSON	75
Figura 41 Función de recolección de información API.....	76

Figura 42 Widget FutureBuilder para traer datos.....	77
Figura 43 Función POST para mandar datos al API	78
Figura 44 ListView Builder para crear objetos de manera dinámica.....	78
Figura 45 Función para comprobar internet.....	79
Figura 46 Función para abrir la cámara.....	80
Figura 47 función POST para crear sugerencias	80
Figura 48 bloqueo al acceso publico.....	81
Figura 49 sitio registrado.....	82
Figura 50 Cultivo en sitio registrado	82
Figura 51 Análisis de pruebas.....	88

LISTA DE ANEXOS

Anexo A Encuestas y Entrevistas a Personal con cercano con el manejo del cultivo de cacao.....	98
Anexo B Especificación de requerimientos de software	110
Anexo C Documento de casos de uso.....	116
Anexo D Documento de arquitectura.....	168
Anexo E Interfaces.....	176
Anexo F Pruebas.....	181

RESUMEN

El objetivo de este documento es mostrar la solución planteada por el equipo de trabajo al problema de moniliasis en cultivos de cacao del departamento de Santander desde un enfoque ingenieril desarrollando una aplicación móvil que hace uso del reconocimiento de imágenes para detectar esta enfermedad, para esto se identificó el problema a partir de documentos en entidades agricultoras como Fedecacao y Agrosavia, las cuales hacen hincapié en que la mayor parte de sus soluciones a este problema están alejadas del entorno tecnológico, esto hizo que el equipo de trabajo se enfocara en desarrollar una solución tecnológica la cual mezcla diferentes sistemas en uno para formar una arquitectura completa que supla las necesidades de este problema, se partió primero por realizar encuestas y entrevistas a habitantes del municipio San Vicente de Chucuri en el departamento de Santander una vez recogidas estas encuestas y entrevistas se dispuso a hacer un análisis de cuáles son las necesidades de este poblado, una vez identificadas se trabajó en la redacción de requerimientos que puedan dar solución a esta problemática de forma que este poblado a su vez pueda usar la solución de manera fácil y sencilla, los requerimientos redactados dieron paso a diseñar diagramas que pudieran clarificar el objetivo de solución, diagramas como los de casos de uso y bases de datos fueron vitales para esto, aquellos diagramas fueron transformados a su vez en interfaces amigables programadas por los desarrolladores. El sistema compuesto de un admin web para la administración de datos y el API el cual se encarga de distribuirlos es solo una de las partes importantes, el reconocimiento de imágenes hecho en teachable machine dio excelentes resultados para detectar la monilia en cultivos de cacao y una vez integrado con la aplicación se dio a la tarea de probar todo el sistema, el cual cumplió con las pruebas establecidas, a modo de conclusión el sistema realiza todas las funciones establecidas por el equipo de trabajo y tiene un gran potencial el sector agricultor de Colombia ya que al ser un sistema que puede incluir rápidamente nuevos cultivos soluciona un problema latente de enfermedades y plagas a lo largo del territorio nacional

Palabras Clave: *Agricultura, Reconocimiento de imágenes, Aplicaciones móviles, Sistema compuesto.*

Línea de Investigación: Tecnología y Sociedad.

1. INTRODUCCIÓN

1.1 PLANTEAMIENTO DE LA PROBLEMATICA

La producción agrícola de Santander es una de las más sólidas del país; además, su relevancia radica en que esa actividad es adelantada, en su gran mayoría, por pequeños agricultores a lo largo y ancho del departamento, durante los últimos años el ministerio de agricultura ha realizado estudios estadísticos que conllevan a destacar cultivos como la palma de aceite, café, caña panelera, plátano y cacao los cuales arrojan una participación del 60.5% de la producción departamental (Fedecacao, 2015). El cultivo del cacao es uno de los cultivos tropicales de mayor importancia, tanto en el ámbito nacional como internacional. En Colombia, intervienen de forma directa más de 25 mil familias, de las cuales el 90% desarrolla su proceso productivo en condiciones de economía campesina. Datos históricos afirman que en el año 2008, Santander participó en la producción nacional de cacao con el 50%, equivalente a 19 mil toneladas de grano(Fedecacao, 2015).

El factor que más limita la producción de cacao en el país es la presencia de enfermedades, entre las cuales se destacan la monilia (*Moniliophthora roreri*), la escoba de bruja (*Phytophthora sp.*), entre otras. Alrededor de 40% de la producción del cacao se ve afectada por la monilia(Javier et al., 2014), la cual se prolifera por ambientes de alta humedad, mala fertilización e inadecuado manejo del cultivo por parte de los agricultores, estos factores harán que el cultivo esté débil y expuesto a la incidencia y severidad de la enfermedad causando pérdidas de hasta un 100% de la producción del fruto (Javier et al., 2014); estas y muchas otras razones hacen que la enfermedad sea considerada como la más frecuentes y severas en el cultivo de cacao Colombiano. Existen organizaciones a nivel nacional como el ICA, ministerio de agricultura y fedecacao, que se encargan de proveer la mayor información de prevención y control en contra de estas enfermedades por medio de artículos, páginas web, panfletos, carteles de información, entre otros; estas estrategias se encargan de combatir la proliferación de la enfermedad pero aún no se ha evidenciado un cambio significativo en las pérdidas de cacao con la implementación de ellas, a partir de la búsqueda realizada por el equipo de trabajo acerca de las diferentes herramientas usadas por los entes nacionales, se evidenció el poco uso de tecnología de última generación como las aplicaciones móviles y el reconocimiento de imágenes, hacer uso de estas herramientas puede potenciar la detección de la monilia en una etapa temprana y de esta forma se pueda actuar rápidamente para no sufrir pérdidas en el cultivo.

Cabe mencionar que el uso de los dispositivos móviles en territorio colombiano a aumentado considerablemente, según Deloitte el 72% de las personas encuestadas

en su estudio de “consumo móvil en Colombia” han adquirido sus móviles en los últimos 2 años (Deloitte, 2019), aparte de que en este mismo estudio se puede observar la cantidad de tiempo que las personas usan su dispositivo móvil y las aplicaciones que suelen usar, basado en estos datos se propone el desarrollo de una aplicación móvil que haga uso de la detección de imágenes para proporcionar un diagnóstico de la enfermedad, dicho diagnóstico se presentará a modo de porcentaje que indicará la presencia de la enfermedad y a partir de este dato se aconsejara al agricultor sobre los tratamientos adecuados para prevenir y evitar la propagación de la moniliasis, de esta manera se contribuirá a disminuir el tiempo que conlleva la detección de la enfermedad y así el agricultor pueda tomar acciones en el menor tiempo posible.

La aplicación móvil extraerá los datos directamente de un API REST que se encargará de gestionar los datos que deberá recibir la aplicación, este API se diseñará con una estructura abstracta para que en un futuro dado se puedan almacenar no solamente datos sobre el cultivo de cacao si no sobre otros cultivos, y a partir de su estructura abstracta se facilitará la actualización de información y se promoverá la escalabilidad del proyecto lo cual puede dar camino a otros desarrollos basado en la estructura de la API.

2. OBJETIVO GENERAL

Desarrollar el prototipo de una aplicación móvil para el sistema operativo Android haciendo uso de algoritmos de reconocimiento de imágenes para la detección de la monilia en cultivos cacao del departamento de Santander.

2.1. OBJETIVOS ESPECÍFICOS

- Realizar un análisis de requerimientos para la detección de la monilia en los cultivos de cacao que se siembran en el departamento de Santander por medio de consultas a entes agricultores con el fin de definir las características, elementos o funcionalidades del sistema a desarrollar
- Diseñar la estructura que tendrá el sistema a desarrollar haciendo uso de diagramas UML y wireframes para definir la arquitectura, componentes e interfaces que este tendrá
- Desarrollar el sistema planteado mediante el uso de bases de datos, servicios de reconocimiento de imágenes y frameworks de desarrollo de aplicaciones móviles multiplataforma, con el fin de cumplir con los requerimientos definidos previamente.
- Comprobar que el sistema realice correctamente las tareas asignadas en la especificación de requerimientos a través de la creación de escenarios de pruebas las cuales determinarán su funcionamiento integral.

3. ESTADO DEL ARTE

A continuación, se presentan los artículos, publicaciones, revistas, aplicaciones y sitios web que se tomaron en cuenta para el desarrollo de este proyecto.

Tabla 1 Articulo 1

Articulo 1	
Nombre del proyecto	Principales Cultivos por Área Sembrada en 2017 Santander
Autores	Ministerio de agricultura, Evaluaciones agropecuarias municipales
Resumen	En este informe se destaca la producción a nivel general agrícola y la predominancia de ciertos cultivos en el departamento de Santander, al mismo tiempo muestra en que municipios del departamento se han sembrado los diferentes cultivos.
Aporte al proyecto	Esta información ayuda a tener una idea de los principales cultivos de la región y cuál de ellos se pueden usar como material de trabajo, dado su ubicación y su popularidad.
Fuente: Autor	

Tabla 2 Articulo 2

Articulo 2	
Nombre del proyecto	Biodiversidad de ecotipos de Moniliophthora roreri en cacao (<i>Theobroma cacao</i>) clon ccn-51 y la actividad antagonista de PGPR
Autores	Romero Meza Ricardo Fernando
Resumen	El objetivo principal de esta tesis de grado es observar el comportamiento de la Moniliasis utilizando "Rizobacterias Promotoras del Crecimiento Vegetal" con la implementación de estas rizobacterias se busca un trato amigable con el medio ambiente, ya que los métodos de control utilizados normalmente inhiben el crecimiento de otros seres vivos dentro del ambiente donde crece el cacao.
Aporte al proyecto	Dentro de este artículo se habla sobre la variedad de moniliasis que pueden presentarse en el medio y como la ubicación y condiciones climáticas pueden ayudar a la proliferación de la misma, estos datos ayudan a proveer a los usuarios información más detallada acerca de la enfermedad que está atacando su cultivo.
Fuente: Autor	

Tabla 3 Articulo 3

Articulo 3	
Nombre del proyecto	Estado de la moniliasis del cacao causada por Moniliophthora roreri en Colombia
Autores	Javier Correa Álvarez, Sergio Castro Martínez y Jairo Coy
Resumen	Este artículo provee información general acerca de la Moniliasis cubre su origen, países donde se encuentra actualmente, distribución en el territorio colombiano, su ciclo de vida desde lo genético hasta lo percible al ojo humano y los diferentes controles que se le han dado a la enfermedad.
Aporte al proyecto	Da un enfoque general del estado de la Moniliasis en el país y diferentes características a tomar en cuenta como su ciclo de vida ya que de esta forma se puede identificar el sector al que se debe enfocar el proyecto.
Fuente: Autor	

Tabla 4 Articulo 4

Articulo 4	
Nombre del proyecto	Factores que afectan la producción de cacao (theobroma cacao l.) en el ejido francisco i. madero del plan Chontalpa, tabasco, México
Autores	Miguel Sánchez Hernández, Tomás Martínez Saldaña, Engelberto Sandoval Castro, Víctor Córdova Avalos, Néstor Estrella Chulim, Carlos F Ortiz García y Alfonso Macías Layalle
Resumen	Se evalúan las diferentes condiciones que pueden afectar la producción de cacao, teniendo en cuenta los diferentes procesos que se usan para control, cuidado, siembra, podas y cosecha, estos procesos se analizaron con cuidado y llegaron a la conclusión de que diferentes factores tales como la edad avanzada de las plantaciones y control de plagas eran de vital importancia para la alta productividad del cacao.
Aporte al proyecto	Es importante conocer los diferentes factores que pueden afectar a la plantación, sobre todo los factores externos ya que, aunque no se vea con notoriedad estos factores pueden influir en el diagnóstico que se va a proveer

Fuente: Autor

Tabla 5 Artículo 5

Artículo 5	
Nombre del proyecto	Estrategias de control de Moniliophthora roreri y Moniliophthora perniciosa en Theobroma cacao L.: revisión sistemática
Autores	Paola Andrea Tirado Gallego, Andrea Lopera Álvarez and Leonardo Alberto Ríos Osorio
Resumen	En este artículo se trabajan dos tipos de moniliasis con un enfoque investigativo haciendo uso primero de filtros en bases de datos como Scopus y Springerlink para hacer un reconocimiento de la literatura actual acerca de las estrategias de control que existen para la enfermedad, a partir de los resultados de estos filtros los autores realizan una segmentación de resultados a partir de criterios establecidos por los mismos, una vez aplicados los criterios quedan con un banco de literatura de la cual extraen información clara acerca de las estrategias de control que se usa en el Theobroma cacao, dentro de qué categoría están y cómo se ejecuta esta estrategia de control.
Aporte al proyecto	Su enfoque investigativo aporta una forma de realizar filtros a la investigación en curso, provee información acerca de las diferentes estrategias que se pueden recomendar al usuario para el control de su cultivo.
Fuente: Autor	

Tabla 6 Artículo 6

Artículo 6	
Nombre del proyecto	Caracterización fisicoquímica de materiales regionales de cacao Colombiano
Autores	Janeth Aide Perea, Olga Lucia Ramírez and Arley Rene Villamizar
Resumen	A partir del uso de una variación de granos bastante aceptable los investigadores proponen el análisis físico y químico de estos para verificar su funcionamiento desde su nivel de pH hasta los niveles de zinc que estos contienen, el resultado de estos análisis destacó dos granos con características sobresalientes a comparación de las demás
Aporte al proyecto	Los diferentes tipos de granos pueden ser tomados en cuenta a la hora de dar un diagnóstico, por lo tanto, esta información puede ser de gran ayuda.

Fuente: Autor

Tabla 7 Articulo 7

Articulo 7	
Nombre del proyecto	Guía Técnica del cultivo de cacao
Autores	Fedecacao
Resumen	posicionamiento del cacao colombiano en el mercado en mercados internacionales presentando una reseña histórica, botánica y sus diferentes fases, manejo de la plantación
Aporte al proyecto	Estadísticas de la producción y posicionamiento del cacao colombiano a nivel nacional e internacional, contextualización de los bajos niveles de tecnologías aplicados al cultivo y cómo esto afecta a su producción, proceso de crecimiento del cultivo con sus diferentes etapas, afecciones y manejo de estas mismas
Fuente: Autor	

Tabla 8 Articulo 8

Articulo 8	
Nombre del proyecto	<u>Pl@ntNet Identify</u>
Autores	The plantNet
Resumen	pl@ntNet es una aplicación para la recopilación, anotación y recuperación de imágenes que ayudan al reconocimiento de plantas, incluye una base de datos de imágenes botánicas de las diferentes plantas la aplicación arroja como resultado nombre botánico de la planta y una imagen de la base de datos.
Aporte al proyecto	Reconocimiento de imágenes, diagnóstico de y clasificación, la base de datos evoluciona a medida que el usuario contribuye con contenido
Fuente: Autor	

Tabla 9 Articulo 9

Articulo 9	
Nombre del proyecto	Plantix
Autores	Plantix
Resumen	Plantix es una aplicación móvil que usa reconocimiento de imágenes para detectar si una planta está enferma, una vez detectado da recomendaciones acerca de los tratamientos que se le deben hacer a la planta.
Aporte al proyecto	Aporta gran contenido a la parte de interfaces de la aplicación y la organización de la información.
Fuente: Autor	

Tabla 10 Articulo 10

Articulo 10	
Nombre del proyecto	Iterative development of Vegethon: theory-based mobile app intervention to increase vegetable consumption
Autores	Sarah A. Mumah, Abby C. King, Christopher D. Gardner y Stephen Sutton
Resumen	Esta aplicación destaca por ser informativa en el campo del consumo de vegetales por población estadounidense que sufre de problemas de obesidad, donde estos organizan la información de todos los vegetales a consumir en una dieta semanal y a medida que pasa el tiempo se van haciendo recomendaciones de diferentes vegetales para una dieta balanceada.
Aporte al proyecto	El modo en que categorizan la información de los vegetales y sus interfaces gráficas pueden ser usados como referencia para el proyecto.
Fuente: Autor	

Tabla 11 Articulo 11

Articulo 11	
Nombre del proyecto	Effect of a mobile app intervention on vegetable consumption in overweight adults: a randomized controlled trial
Autores	Sarah Mumah, Thomas N. Robinson, Maya Mathur, Sarah Farzinkhou, Stephen Sutton y Christopher D. Gardner
Resumen	Este artículo evalúa la intervención de una app para el control de consumo de vegetales en poblaciones con sobrepeso, donde se toma en cuenta la cantidad de vegetales que consume el usuario a lo largo de la semana y con esto evaluar la funcionalidad de la aplicación y proponer posibles proyectos a futuro.
Aporte al proyecto	Da un enfoque estadístico del impacto que podría dar una aplicación en una población y provee una segmentación de población que puede ser usada en el proyecto
Fuente Autor	

Tabla 12 Articulo 12

Articulo 12	
Nombre del proyecto	Diseño de una aplicación Android para la enseñanza de tropismos y adaptaciones vegetales
Autores	Jesús Santos Cuesta
Resumen	A lo largo de este trabajo, se presenta una aplicación para sistemas Android que incluye una serie de actividades, cuya temática tratada son las adaptaciones de los vegetales a distintos ambientes y sus tropismos en distintos cursos académicos. Esta aplicación promueve el uso de la tecnología con fines educativos, a la vez que pretende despertar el interés de los estudiantes por el estudio de las plantas y su importancia en la vida cotidiana
Aporte al proyecto	El aporte que esta tesis puede dar al proyecto es la estructuración de la aplicación y organización del trabajo.
Fuente: Autor	

Tabla 13 Articulo 13

Articulo 13	
Nombre del proyecto	Mobile smart device-based vegetable disease and insect pest recognition method
Autores	Kaiyi Wang, Shuifa Zhang, Zhibin Wang, Zhongqiang Liu and Feng Yang
Resumen	A lo largo de este artículo se habla de los diferentes algoritmos que se pueden usar para la clasificación y detección de enfermedades e insectos que pueden atacar a las hojas en cualquier tipo de cultivo, dentro de sus procedimientos se destaca la toma de imágenes y la identificación de posibles segmentos de la imagen donde se encuentre la enfermedad o el insecto.
Aporte al proyecto	Este artículo da información acerca de los algoritmos de clasificación imágenes y su funcionalidad, la identificación de píxeles que se puede hacer a las imágenes para diferenciar ciertas tonalidades que pueden ser decisivas a la hora de clasificar si el vegetal está enfermo o no.
Fuente: Autor	

Tabla 14 Articulo 14

Articulo 14	
Nombre del proyecto	Identification and Classification of Diseases in Carrot Vegetable Using Discrete Wavelet Transform.
Autores	Gouri C Khadabadi, Arun Kumar and Vijay S Rajpurohit
Resumen	En este artículo se destaca el uso del procesamiento de imágenes para la detección y clasificación de enfermedades en las zanahorias, dividen el trabajo en varias etapas donde se realiza primeramente la toma de las imágenes para posteriormente cortarlas usando EDGE y tratar con el área específica de la imagen y por último usar la transformada de wavelet para la identificación de los segmentos enfermos de la zanahoria.
Aporte al proyecto	El algoritmo de recorte de imágenes EDGE aporta la importancia de hacer un análisis solo al objeto a analizar, ya que dentro de los algoritmos de reconocimiento de imágenes se toman en cuenta todos los píxeles de una imagen, la transformada de wavelet hablada en el artículo aporta otro algoritmo a tener en cuenta para la identificación de los segmentos afectados.
Fuente: Autor	

Tabla 15 Articulo 15

Articulo 15	
Nombre del proyecto	Automatic fruit and vegetable classification from images
Autores	Anderson Rocha, Daniel C Hauagge, Jacques Wainer and Siome Goldenstein
Resumen	Este artículo destaca por el uso de diferentes algoritmos para recorte de fondos en imágenes a procesar, donde abordan el problema de identificar frutas y vegetales de un supermercado haciendo uso de 5 algoritmos los cuales recortan y analizan los píxeles de la imagen para después procesarlos e identificar qué fruta o vegetal es.
Aporte al proyecto	Las gráficas que contiene el artículo comparando la precisión de los 5 algoritmos de recorte aportan un factor de decisión a la hora de escoger algoritmos de esta clase.
Fuente: Autor	

4. MARCO CONCEPTUAL

A continuación, se presentará una serie de conceptos necesarios para la comprensión del cómo se desarrolló el proyecto.

Cacao: El cacao es un árbol leñoso, fuerte, de porte relativamente bajo, originario del sotobosque. Pertenece al orden malvales, familia Malvaceae, género Theobroma, especie (*Theobroma cacao L*), cuya denominación científica fue dada por el eminente botánico Linneo. Las flores del cacao se encuentran distribuidas a lo largo del tronco y de las ramas, agrupadas en sitios llamados cojines florales. La flor del cacao es caulinar, es decir, se produce en el tronco, ramas y tallos leñosos. La flor del cacao es hermafrodita, posee ambos sexos: masculino y femenino. (Fedecacao, 2015)

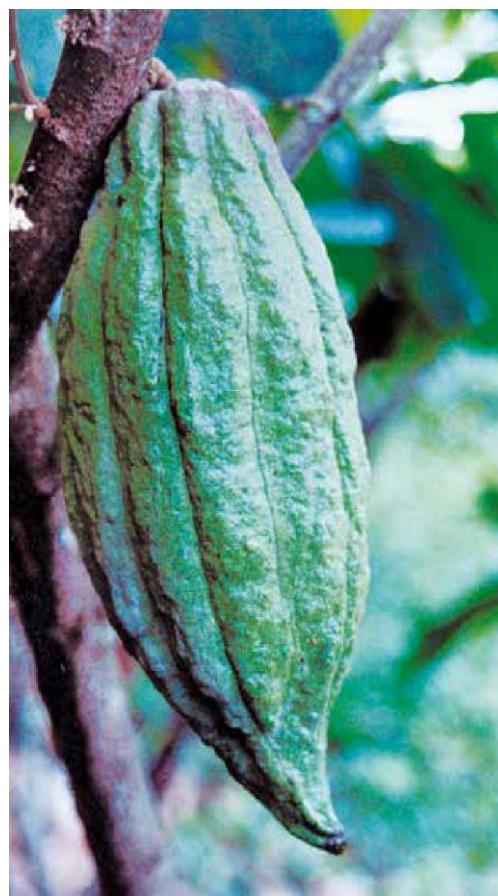
Tipos de cacao cultivados: Para el agricultor es conveniente conocer los tipos de cacao existentes porque, de acuerdo con ellos, se define la calidad del grano; aspecto de máxima importancia en los tiempos modernos cuando el mercado exige determinadas características favorables para la industria transformadora y para atender el gusto del consumidor de chocolate. De otro lado, los tipos de cacao también presentan diferencias en el comportamiento agronómico de las plantaciones, sobre su precocidad, su tolerancia frente a las plagas y enfermedades y en gran medida, en relación con los rendimientos. Tales asuntos deben ser tenidos en cuenta a la hora de la decisión sobre el tipo de cacao a cultivar pues ello atañe de manera definitiva a la economía del cultivador ya que una elección equivocada puede limitar los rendimientos o significar dificultades para la comercialización. El tipo de cacao a utilizar, debe corresponder a las condiciones propias de la finca y del agricultor y a las exigencias concretas del mercado. Las posibilidades de éxito de un determinado tipo de cacao, proporcionan las bases para la toma de la decisión más acertada. Desde el punto de vista de la botánica, existen tres tipos de cacao, que a la vez determinan tres clases de grano, en cuanto a la calidad. Ellos son los tipos criollos, los forasteros y los híbridos. (Fedecacao, 2015)

Tipo Criollo: Son los más finos, caracterizados por su agradable sabor y exquisito aroma. El tipo criollo fue el único cultivado en Colombia hasta 1885 cuando se introdujo el llamado cacao pajarito de origen amazónico, cuya mazorca es de tipo calabacillo. Se caracteriza por presentar tronco erecto, con poca ramificación lateral, con tendencia al crecimiento vertical. El fruto es muy rugoso, con diez surcos profundos, su cáscara es delgada fácil de quebrar. Las semillas son rollizas, casi redondeadas, cuyos cotiledones frescos son de color blanco o rosado. El tamaño y forma de la mazorca varía según los tipos regionales, pero es común en Colombia que el fruto sea alargado, un poco más ancho cerca del pedúnculo y delgado o punta aguda en el extremo apical. En épocas pasadas, era común mencionar el

cacao criollo como “Hartón” en Antioquía o Nariño, “Criollo Real” en Santander y en otras regiones. En términos generales, los criollos puros presentan condiciones desfavorables para su manejo, pues resultan muy delicados, con alta susceptibilidad a las plagas y enfermedades. En particular, son retardados en el inicio de la producción y de poca productividad por árbol. A pesar de las condiciones de alta calidad del grano, apetecido en los mercados más exigentes del mundo, resultan poco atractivos a la mayoría de los agricultores, dadas estas características. Dentro de este tipo de cacao, se clasifican los originarios de México y Centroamérica y los venezolanos que fueron llevados a otras partes del

mundo como Trinidad, algunas islas de las Antillas e incluso a África. En Colombia existen varias poblaciones de tipo criollo, infortunadamente en decadencia y franca disminución, lo que ameritaría un trabajo de rescate de ese germoplasma que podría ser útil en el empeño de mejorar la calidad del grano. (Fedecacao, 2015)

Figura 1 Fruto de cacao tipo criollo



Fuente: Guía técnica del cultivo de cacao

Los Tipos Forasteros: Son los de menor calidad relacionada con el sabor y el aroma que confieren sus granos al chocolate elaborado con ellos. Sin embargo,

presentan otras condiciones de calidad interesantes para la industria como lo puede ser su rendimiento en contenido de grasa y otras características deseables. Es también denominado amazónico por relacionar su origen con la región de ese nombre. El árbol del tipo forastero suele ser vigoroso, con tendencia a ramificar lateralmente y en algunos casos su follaje decumbente. (Fedecacao, 2015)

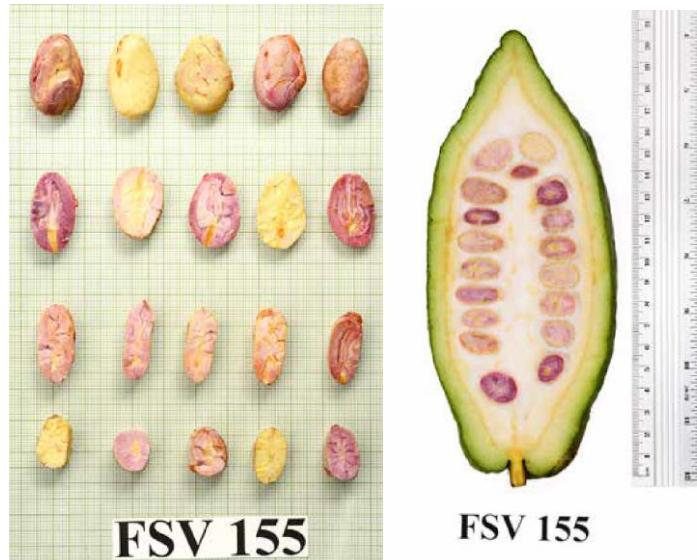
Figura 2 Fruto de cacao tipo forastero



Fuente: “Guía técnica del cultivo del cacao”

Los principales tipos híbridos: Resultan del cruzamiento sexual de dos árboles, usualmente dirigido por el hombre luego de un proceso de selección, tratando de generar determinadas características deseables. El cruzamiento en términos generales se hace entre clones, con condiciones opuestas a fin de mejorar aspectos de interés como la calidad, productividad, precocidad, respuesta a plagas y enfermedades, etcétera. La hibridación se dio de manera espontánea, en la isla antillana de Trinidad donde el criollo que fue llevado en un principio se cruzó con el forastero, dando origen a un tipo intermedio que, si bien fue catalogado como forastero, su calidad resultó superior a la de éste. Posteriormente en Trinidad, de manera consciente, se inició la producción de los híbridos que en Colombia se propagaron de manera generalizada desde mediados del siglo XX, hasta los inicios del siglo XXI. Muchos de los clones conocidos en la actualidad como universales, fueron seleccionados en Trinidad a partir de poblaciones híbridas, es decir, Clon Fedecacao San Vicente 155. Clon Fedecacao San Vicente 155. (Fedecacao, 2015)

Figura 3 Clon de cacao tipo híbrido "clon san Vicente 155"



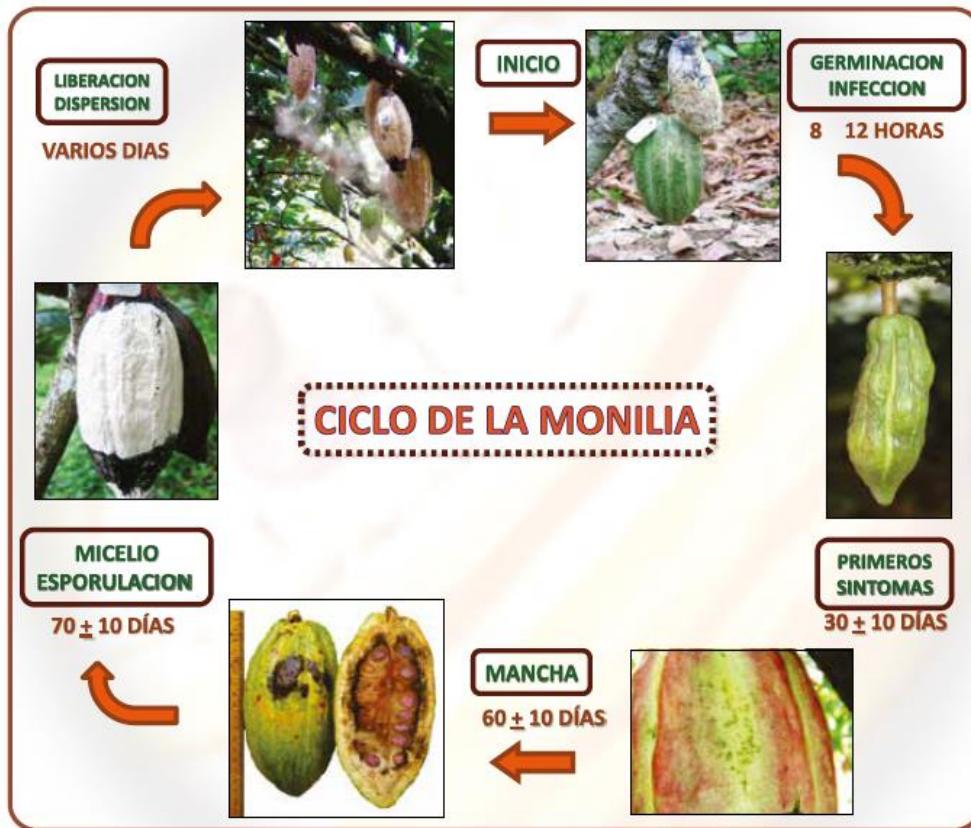
Fuente: “Guía técnica del cultivo de cacao”

Clonación del cacao corresponden a individuos sobresalientes de la descendencia de aquellos cruzamientos. Los clones más conocidos, generados de esa manera en Trinidad, son el: ICS-I, ICS-39, ICS 40, ICS-60, ICS-95, TSH-565, entre otros. Entre dichos clones no se encuentran materiales puramente criollos, ni puramente forasteros, tratándose de los que proceden de cruzamientos entre individuos con genes de unos y otros. (Fedecacao, 2015)

Enfermedades del cacao: El cacao, como todos los seres vivos, plantas y animales, está sometido a la acción de patógenos que causan efectos negativos en términos de la productividad, deteriorando los órganos de la planta, ocasionando la muerte de las mismas. El control de tales enfermedades constituye una de las prácticas esenciales en el cultivo. Algunas atacan los órganos aéreos de la planta y otras las raíces. Las de mayor ocurrencia son las causadas por hongos. Las enfermedades son los fenómenos que más pérdidas económicas producen al cacaocultor, siempre y cuando no se les proporcione el manejo adecuado. (Fedecacao, 2015)

Monilia (*Moniliphthora roreri*): La monilia es una de las enfermedades de mayor frecuencia en el cultivo de cacao ya que es causada por hongos específicamente el hongo denominado (*Moniliphthora roreri*). Ataca solamente los frutos en todos los estados de desarrollo desde los recién formados, hasta los que se encuentran a punto de maduración. En frutos jóvenes o pepinos, de menos de tres meses se produce deformaciones, abultamientos seguido por la aparición de manchas de color negro que terminan cubriendo todo el fruto. (Fedecacao, 2015)

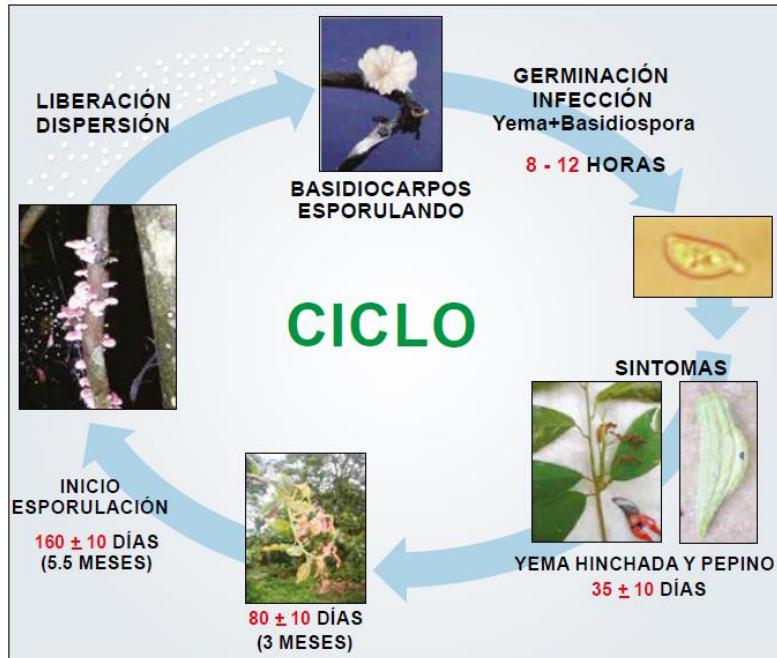
Figura 4 Ciclo de vida de la monilia



Fuente: "Guía técnica del cultivo de cacao"

Escoba de bruja (*Moniliophthora perniciosa*): Esta enfermedad es causada por un hongo, cuyo nombre científico es (*Moniliophthora perniciosa*), afecta todas las estructuras aéreas de la planta, ubicándose en los tejidos meristemáticos que son los que están en proceso de multiplicación dentro del crecimiento de los órganos de la planta, excepto en los tejidos de la raíz. Así que ataca las yemas terminales de las puntas de las ramas, las yemas axilares, los cojines florales y los frutos. El síntoma característico es la proliferación de brotes que dan origen a estructuras, abultadas exageradamente, produciendo el crecimiento de varias ramas hipertrofiadas al mismo tiempo a manera de abanico, como si fuera una escoba, lo cual dio origen a su denominación popular. (Fedecacao, 2015)

Figura 5 Ciclo de vida de la enfermedad escoba de bruja



Fuente: “Guía técnica del cultivo de cacao”

Sistema agroforestal: Un buen sistema agroforestal nos permite definir en donde estarán ubicados en la parcela, el número de plantas de cacao y árboles acompañantes, y cómo éstos van a ir disminuyendo con el tiempo. Un buen diseño agroforestal debe garantizar que haya un mejor uso del espacio, el agua y los nutrientes. El diseño del sistema debe ayudar a regular la temperatura en la plantación, moderar la entrada de luz, disminuir el efecto del viento, proteger y mejorar el suelo. Todo esto ayuda a generar una alta productividad y fomentar la biodiversidad en las parcelas del cacao. (Fedecacao, 2015)

Características agroecológicas: La agroecología es un enfoque teórico que busca aumentar la sustentabilidad agraria desde las perspectivas ecológica, social y económica. (Fedecacao, 2015)

Conidias: Las conidias son esporas asexuales no móviles, que se forman (exógenamente) en el ápice o en el lado de una célula esporógena. A veces una célula hifal preexistente puede convertirse en una conidia. Las conidias no se forman obligadamente dentro de una estructura como ocurre con las esporangiosporas, sino que se desprenden de las células terminales o laterales de hifas especializadas denominadas conidióforos. Germinan siempre por la formación

de tubos germinativos. Las conidias son producidas exógenamente sobre hifas especiales llamadas conidióforos, normalmente en gran número y velocidad. (Fedecacao, 2015)

Lenguaje de programación: Un lenguaje de programación es un lenguaje de computadoras que usan los programadores para desarrollar programas de software, scripts e instrucciones que son ejecutadas en el ordenador, en la actualidad existen una gran cantidad de lenguajes de programación y cada uno de ellos tiene su sintaxis, una vez un programador aprende las reglas y estructura del lenguaje, escriben su código en un editor de texto, luego de esto se compila todo en código de máquina para que pueda ser entendido por la computadora y de esta manera se ejecute lo que el usuario o programador desea. (Computer Hope, 2019)

Programación orientada a objetos: Este paradigma de programación usa objetos en sus interacciones para diseñar aplicaciones y programas informáticos, se basa en varias técnicas en las cuales está la herencia, el polimorfismo, encapsulamiento entre otros, a este paradigma se lo comenzó a dar uso en la década de 1990 y en la actualidad su uso es frecuente tanto así que la mayoría de lenguajes de programación lo soportan, los principios que se destacan de la programación orientada a objetos son los siguientes:

Abstracción: Denota las características fundamentales de un objeto, este objeto es abstracto y su trabajo es informar y cambiar su estado.

Encapsulamiento: Reúne a todos los elementos de un mismo objeto desde sus atributos, hasta sus métodos donde podemos definir si estos pueden compartir información con unas clases y otras.

Herencia: Las clases se relacionan entre sí, llevando una jerarquía definida por el usuario, diferentes objetos heredan propiedades de todas las clases a las que pertenecen.

Polimorfismo: Se describe como los métodos que contiene un objeto, estos métodos pueden llamarse de la misma manera y tener comportamientos diferentes. (MDN contributors, 2020)

Python: Python es un lenguaje de programación multiparadigma que soporta la programación orientada a objetos, programación imperativa y la programación multifuncional, este lenguaje es interpretado, dinámico y multiplataforma, hoy en día Python es reconocido mundialmente como uno de los lenguajes de programación más potentes y robustos, su avance le ha hecho un camino en la inteligencia artificial en la cual expertos no dudan en recomendarlo si se está empezando en esta ciencia, la comunidad dedicada a la creación de librerías para inteligencia artificial han enriquecido en gran manera este lenguaje llevándolo a los extremos donde nunca se había llegado como el reconocimiento y tratado de imágenes y videos a tiempo real. (Knowlton & Jim, 2008)

Dart: Originalmente llamado DASH es un lenguaje de programación optimizado al cliente para aplicaciones multiplataforma, este lenguaje desarrollado por Google es usado para desarrollar aplicaciones de escritorio, web y móviles, como la gran mayoría Dart es orientado a objetos lo que lo hace tan especial es que este puede compilar en su código nativo o en JavaScript, soportando interfaces, clases abstractas entre otras características que lo hacen un lenguaje potente a la hora de programar. (Google, 2020b)

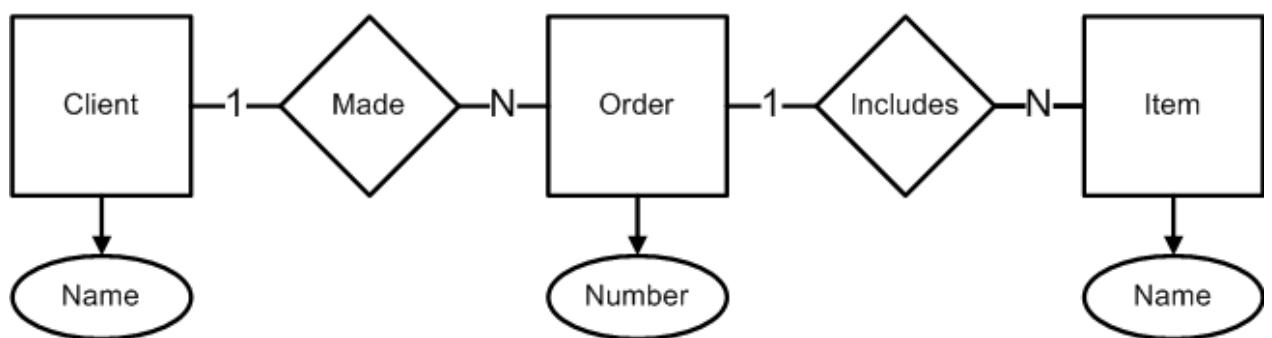
Framework: Un framework o marco de trabajo, es un entorno con una estructura definida, usualmente contienen artefactos o módulos que lo ayudan a tener una organización dentro del desarrollo de software, incluye programas, librerías y otras herramientas que se necesitan a la hora de desarrollar un proyecto. estos marcos de trabajo tienen como objetivo principal ofrecer una funcionalidad definida siendo construidos en base a patrones de diseño. (Riehle & Dirk, 2000)

Django: Django es un framework de programación web de alto nivel que destaca por su rápido y limpio desarrollo, usando un diseño pragmático haciendo uso del patrón de diseño MVC, Django fue desarrollado para ayudar a los desarrolladores a tomar conceptos de aplicaciones, modificarlos a su manera lo más rápido posible, dentro de sus características principales resalta la escalabilidad que provee a sus aplicaciones siendo este fundamental en cualquier proyecto a desarrollar, La meta fundamental de Django es facilitar la creación de sitios web complejos. este hace énfasis en la reutilización, la conectividad, el desarrollo rápido y el principio de no repetir. (Django Software Foundation, 2020)

Base de datos: Es un conjunto de datos pertenecientes a un mismo contexto y almacenados de forma sistemática estas que permiten almacenar y posteriormente acceder a los datos de forma rápida y estructurada, estos datos pueden ser usados en un sin fin de medios, las bases de datos son usadas ampliamente en el entorno del desarrollo de software, aunque este no es su único campo de acción. Estas bases de datos proporcionan métodos para mantener la integridad de los datos, para administrar el acceso de usuarios a los datos y para recuperar la información si el sistema llegase a corromperse. (Oracle, 2020)

Modelo Entidad Relación: El modelo o diagrama entidad-relación, es también conocido como ER, este es un tipo de diagrama de flujo que permite ilustrar las entidades como personas u objetos o conceptos que se relacionan entre sí dentro de un sistema. Este tipo de diagramas se usan comúnmente para diseñar o depurar bases de datos de tipo relacional en los campos de ingeniería de software, sistemas de información empresarial, y en campos de educación e investigación. También conocidos como los ERD o modelos ER, emplean un conjunto definido de símbolos, tales como rectángulos, diamantes, óvalos y líneas de conexión para representar la interconexión de entidades, relaciones y sus atributos. Estos permiten reflejar la estructura gramatical, y emplear a las entidades como sustantivos y a las relaciones como verbos. (LucidChart, 2019)

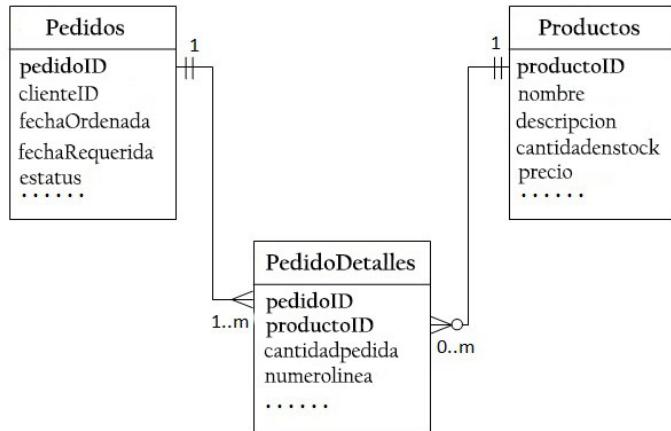
Figura 6 Modelo entidad relación



Fuente: Explain extended

Modelo relacional: El modelo relacional está basado en el concepto matemático de relación, que se puede representar gráficamente mediante una tabla, Es decir, se representa una relación como una tabla en donde las columnas y filas permiten relacionar entre sí los elementos. (Universidad Nacional de Córdoba, 2019)

Figura 7 Modelo relacional relación



Fuente: Lifeder

Sistema de Gestión de Base de Datos (SGBD): Se definen como un tipo de software muy específico, que se dedica a servir como una interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Estos sistemas de gestión permiten describir los elementos de datos con su respectiva estructura, sus interrelaciones y sus validaciones. El propósito más esencial de los sistemas de gestión de bases de datos es consiste en manejar de manera clara, sencilla y ordenada un conjunto de datos para que posteriormente éstos puedan convertirse en información relevante que necesita una organización. (EcuRed, 2019)

PostgreSQL: Es un sistema de gestión de bases de datos relacional orientado a objetos y de código abierto, dirigido por una comunidad de desarrolladores conocida en el mundo de la programación. Una de las características más importantes de PostgreSQL es el control de multiversiones, el cual agrega una imagen del estado de la base de datos a cada transacción lo cual nos da la comodidad de tener transacciones estables y con gran rendimiento lo cual aporta gran flexibilidad a los proyectos, otra característica interesante es la definición de funciones por medio de varios lenguajes de programación lo cual la hace una herramienta perfecta para el desarrollo multiplataforma. (The PostgreSQL Global Development Group, 1996)

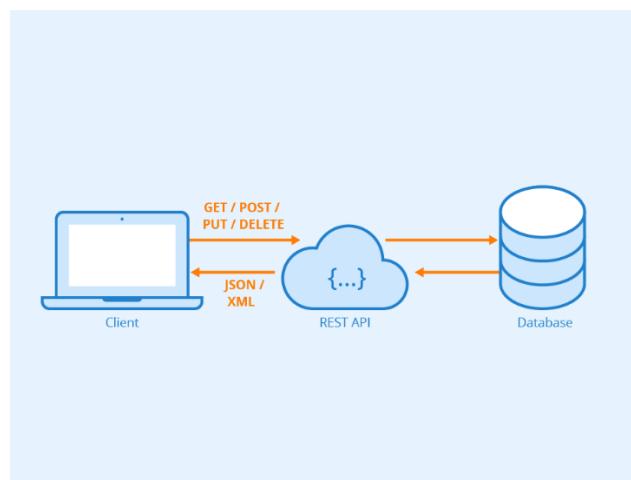
Aplicación móvil: Una aplicación móvil es un programa diseñado para funcionar en celulares, tabletas y otros dispositivos móviles, que concede a las personas realizar actividades de diferente tipo, como por ejemplo de ocio, profesionales, acceder a servicios, informarse, comunicarse, entre muchas otras posibilidades de uso. (SoftCorp, 2019)

Android: Android es uno de los sistemas operativos móviles más usados del mundo, este sistema fue creado por Google, tomando como principal base Linux y otros códigos de software libre. Fue creado para funcionar en dispositivos móviles, como teléfonos inteligentes, tabletas, relojes inteligentes, automóviles y televisores. En un principio fue desarrollado por Android Inc, empresa respaldada por Google económicamente y que fue posteriormente adquirida en 2005. Android fue presentado en 2007 para adelantar los estándares abiertos de los dispositivos móviles. El código fuente principal de Android se distingue como Android Open Source Project. (Android, 2019)

Flutter: Flutter es una plataforma para el desarrollo de aplicaciones móviles, es un SDK código fuente libre creado por Google. Comúnmente es usado para el desarrollo y diseño aplicaciones con los sistemas operativos de Android y iOS. La principal ventaja es que se pueden manejar ambos sistemas operativos en un solo desarrollo, es decir, se tiene el código base de cada sistema operativo. Además de la rapidez de desarrollo y la calidad de aplicación nativa que maneja esta plataforma. (Google, 2019)

API Rest: Un API es el conjunto de comandos y protocolos que se usan para el desarrollo y la integración el software de una aplicación. REST es un conjunto de restricciones con las que se puede realizar un diseño de arquitectura de software. Las operaciones más importantes que permiten manipular los recursos son: GET para consultar y leer, POST para crear, PUT para editar y DELETE para eliminar. (Ordóñez, 2019)

Figura 8 API Rest



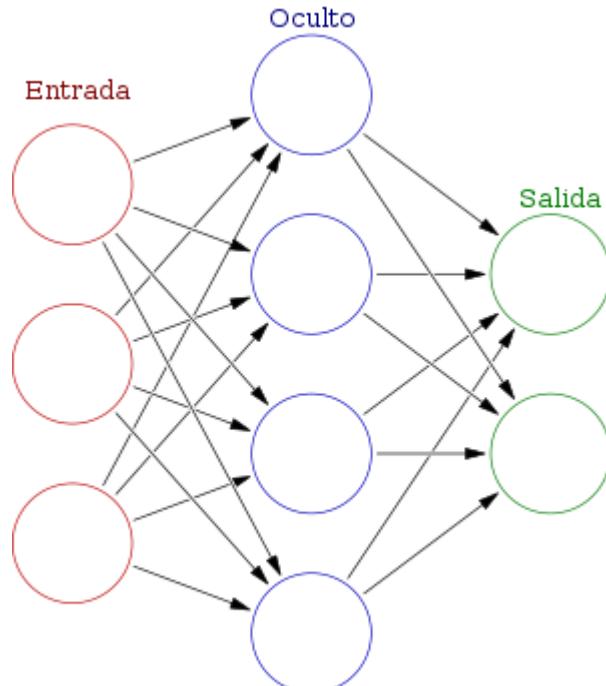
Fuente: Seobility

Modelos de color: Es una manera de describir los colores en un sistema de color casi todos los modelos tienen 3 dimensiones siendo el más conocido el RGB, esto con el hecho de que los sistemas de color son representados en 3 ejes ósea en 3D, la mayoría de los modelos salen del RGB aunque también hay otro a tener en cuenta que es el CMY(cian, magenta y amarillo) estos modelos de color se subdividen en modelos de subcolores y de aquí parten submodelos como el HSL(Hue, Saturation, Lightness que en español sería Matiz, Saturación, Luminosidad) y HSV(Hue, Saturation, Value que en español sería Matiz, Saturación, Valor) ya llevándolo a las aplicaciones del procesamiento de imágenes se deben tener en cuenta estos modelos de color para así dar con diagnósticos y predicciones más precisas (Animations, 2018)

Procesamiento de imágenes: El procesamiento de imágenes hace uso de un computador para procesar imágenes a través de varias técnicas que tienen como foco central mejorar la calidad de la imagen o la búsqueda de información dentro de la misma, este procesamiento consta de un proceso principal llamado "filtrado" en esta etapa se usan una variedad de técnicas tales como la suavización de la imagen, realzamiento de bordes, eliminación de ruido, entre otros aplicando estas técnicas como resultado obtendríamos una imagen más clara si de información se trata ya que se quitan partes en la imagen que no son necesarias para el objetivo deseado. ("Procesamiento de imágenes", n.d.)

Redes neuronales: Modelo que toma inspiración del cerebro humano, se conforma por nodos a los cuales se les denomina neuronas artificiales, estas neuronas se conectan entre sí y transmiten señales a todas las neuronas del nodo, dichas señales se transmiten por todas las neuronas hasta generar una salida, el objetivo principal de las redes neuronales es aprender de forma autónoma de modo que simule el pensamiento humano clasificando de manera lógica cierta información hasta dar con el resultado que se espera de la información que se le ha brindado a la misma, dentro del campo de las redes neuronales se habla de los algoritmos usados para el entrenamiento de la red uno de estos llamado el descenso de gradiente el cual utiliza propagación hacia atrás para hacer cálculos y que estos arrojen un valor de salida que luego será interpretado ya sea por la persona que lo programó o por la misma red neuronal, agregando a los algoritmos está la rama de los evolutivos y por ultimo los recursivos aunque estos son usados con menor frecuencia. (Innovation, 2020)

Figura 9 Red neuronal



Fuente: Wikipedia

Redes neuronales convolucionales: Al igual que las redes neuronales estas toman inspiración del cerebro humano pero con una pequeña variación ya que se basa en lo visual exactamente la corteza visual del cerebro, estas redes tienen la misma constitución que las redes neuronales normales al ser un conjunto de neuronas artificiales pero a diferencia de las otras estas están organizadas en capas de manera que por cada capa que tenga la red neuronal mejor va a ser el resultado cabe destacar que a medida que se va avanzando a través de las capas se reduce el número de neuronas aunque estas a su vez heredan toda la información de sus predecesoras haciendo que estas puedan abarcar más información, para exemplificar se tomara la detección de perros y gatos a través de imágenes cuando se empieza el proceso la primera capa de neuronas detectara líneas rectas y algunas curvas al pasar a la segunda capa se adaptara y detectara líneas rectas y curvas junto con algunas formas poco convencionales así hasta llegar a la última capa donde podrá detectar lo que se le ha dado de entrada en este caso perros y gatos. (Aprende machine learning, 2018)

Custom Visión: Es un servicio cognitivo que deja construir, desarrollar y mejorar clasificadores de imágenes, un clasificador de imágenes es un servicio de inteligencia artificial que aplica etiquetas de acuerdo a sus características visuales, este usa un algoritmo de aprendizaje de máquina para poner estas etiquetas. El uso es simple el usuario o desarrollador sube las imágenes qué desea etiquetar, este mismo etiqueta las imágenes y el algoritmo se auto entrena con estos datos y calcula su nivel de precisión, esto lo hace por medio de test así mismo por último el algoritmo ya entrenado puede ser puesto a prueba por el usuario subiendo varias imágenes que no había colocado en el set de entrenamiento y revisar que su comportamiento sea el adecuado. (Microsoft, 2020)

Teachable machines: Herramienta web que hace que el entrenamiento de modelos basados en machine learning sea rápido, fácil y accesible para todo el mundo, Teachable machine usa 3 pasos claves para el entrenamiento de modelos entre ellos la recolección de datos donde se hace la división de categorías de las imágenes que le pasas al sistema, el entrenamiento donde se entrena el modelo y se puede probar si se están clasificando las imágenes correctamente y por último al exportar el modelo para poder usarlo en webs, aplicaciones y más. (Google, 2020c)

IBM Watson: IBM Watson es la plataforma de Inteligencia Artificial para negocios. No es un elemento aislado, sino una colección de servicios y habilidades que incluyen Machine Learning, razonamiento y decisiones tecnológicas; así como lenguaje, habla y visión tecnológica. Estas capacidades están diseñadas para aprender a escala, razón con propósito, e interactuar con humanos de forma natural para resolver una gran cantidad de problemas prácticos, impulsar la productividad y fomentar descubrimientos a través de diversas industrias.

<https://inndot.com/blog/que-es-ibm-watson-y-los-sistemas-de-ia/>

Visual studio Code: Visual Studio Code. Es un editor de código fuente desarrollado por Microsoft para Windows, Linux y macOS. Incluye soporte para depuración, control de Git integrado, resaltado de sintaxis, finalización de código inteligente, fragmentos de código y refactorización de código. También es personalizable, de modo que los usuarios pueden cambiar el tema del editor, los métodos abreviados de teclado y las preferencias. Es gratuito y de código abierto

https://www.ecured.cu/Visual_Studio_Code

5. METODOLOGÍA

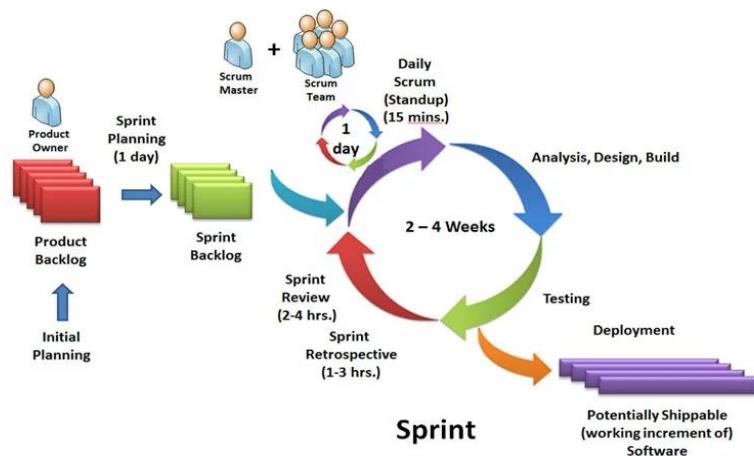
SCRUM es un framework centrado en el desarrollo de software y se puede definir como un marco de trabajo de procesos (Ken Schwaber y Jeff Sutherland, 2017), que posibilita paso a paso el establecer de manera exitosa y organizada el proyecto. Su agilidad es solo una de las muchas características que se pueden destacar de este framework, con el uso de este se pueden lograr excelentes resultados.

Una de las características más representativas de SCRUM es la habilidad de manejar de manera organizada las tareas y contar con la planificación necesaria para lograrlas, estas características son factores clave a la hora de ahorrar tiempo y dinero. En SCRUM se introduce el término de “daily scrum” el cual consiste en reuniones diarias de 15 minutos que aseguran que el proceso se encuentre en la etapa correcta, según se estableció al inicio de cada sprint.

La posibilidad de apreciar el progreso del trabajo ofrece una garantía de que el proceso será transparente y acorde con lo establecido, también ofrece la posibilidad de adaptarse a las necesidades de cualquier proyecto. De esta manera, no solo se concibe la idea de una producción organizada de las tareas, sino también la integración de todas las partes involucradas en el proyecto. La participación y administración de SCRUM es sencilla y de fácil manejo para todas las etapas. De igual manera, se cuenta con un registro de labores realizadas y se le da seguimiento para lograrlo de forma eficiente.

Procedimiento:

Figura 10 Sprint



Copyright © 2011, William B. Heys

Fuente: William B. Heys

El corazón de Scrum es el Sprint, es un bloque de tiempo de un mes o menos durante el cual se crea un incremento de producto terminado utilizable y potencialmente desplegable (Ken Schwaber & Jeff Sutherland, 2017) el sprint es un conjunto de actividades las cuales se desarrollan a lo largo de la duración del mismo y estas son de vital importancia para el funcionamiento del sprint, dentro de estas actividades se destaca el *sprint planning* o planificación del sprint el cual se realiza cada vez que se empezará con un sprint, dentro de este se define el trabajo a realizar usualmente el sprint planning tiene una duración máxima de 8 horas para un sprint de un mes.

Los daily scrums o scrums diarios es una reunión con un bloque de tiempo de 15 minutos para el equipo de desarrollo. El scrum diario se lleva a cabo cada día del sprint. En él, el equipo planea el trabajo para las 24 horas siguientes. Esto ayuda optimiza la colaboración y el desempeño del equipo dando inspección al trabajo avanzado desde el último scrum Diario y realizando una proyección del trabajo del sprint a realizar a continuación

Sprint Review o Revisión del Sprint consta de inspeccionar el trabajo que se hizo a lo largo del sprint, durante esta revisión el equipo de trabajo y los interesados revisan si se debe hacer algún cambio para optimizar el valor del trabajo entregado, esta reunión suele ser de 4 horas para sprints de un mes.

Sprint Retrospective o Retrospectiva del Sprint es un análisis que debe hacerse el equipo de trabajo a sí mismos y revisar en falencias en la manera de trabajar para posteriormente crear un plan que mejore estas falencias para el siguiente sprint, esta retrospectiva suele darse después de hacer la revisión del sprint.

Actividades por sprint

En esta sección se establecieron las actividades a realizar por sprint a lo largo del proyecto.

Sprint 1

- Redactar y definir herramientas, metodologías y preguntas para las encuestas y entrevistas
- Encuestar y entrevistar a entes agricultores para tener varios puntos de vista acerca de las necesidades del sector
- Segmentar los deseos y necesidades identificadas de los entes agricultores para conocer las características a las que se debe enfocar el proyecto
- Plantear los requerimientos funcionales y no funcionales del sistema a desarrollar con el fin de aclarar las funcionalidades a desarrollar

Entregable

- Documento de especificación de requerimientos

Sprint 2

- Diseñar los diagramas de casos de Uso y diagramas de clases para conocer cómo el usuario interactuara con la aplicación
- Diseñar el diagrama de Base de datos con el fin de saber cómo se correlacionan los datos
- Diseñar el diagrama de arquitectura para conocer cómo los componentes interactúan entre sí
- Crear wireframes con el fin de estipular interfaces claras y concisas para la aplicación

Entregable

- En este Sprint se entregarán los diagramas UML, diagrama de bases de datos, wireframes o mockups

Sprint 3

- Desarrollar el API REST y el sistema de administración web con el fin de alimentar este mismo con los datos necesarios que serán consumidos por la aplicación
- Entrenar el sistema de reconocimiento de imágenes para tener listo el ejecutable que se integrará a la aplicación

Entregable:

- API REST, Evidencia entrenamiento de la IA

Sprint 4

- Desarrollar la app móvil para poder integrar todos los componentes previamente desarrollados
- Integrar el sistema de entrenamiento con la app móvil para verificar que el diagrama de arquitectura se está cumpliendo a cabalidad

Entregable:

- Prototipo de aplicación móvil con uso de reconocimiento de imágenes

Sprint 5

- Diseñar escenarios de pruebas para evaluar el desempeño de la aplicación
- Ejecutar los escenarios de pruebas para extraer los datos de desempeño de la aplicación
- Analizar los resultados de los escenarios de pruebas para conocer si el desempeño de la aplicación es el esperado

Entregable:

- Escenarios de prueba y resultados de los mismos

6. ANALISIS DE REQUERIMIENTOS

Con el fin de definir requerimientos idóneos para la solución planteada, el equipo de trabajo decidió dividir esta sección en diferentes actividades, las cuales dieron como resultado un documento de análisis de encuestas y entrevistas realizadas a personas que tienen relación con el cultivo en el departamento de Santander, y un documento de requerimientos tomando como base las entrevistas y las encuestas como también requerimientos generales de un sistema integrado como lo es el desarrollado, ya que comprende el desarrollo de un admin web, un API rest y una aplicación móvil.

6.1 REDACCIÓN Y DEFINICIÓN DE HERRAMIENTAS, METODOLOGÍAS Y PREGUNTAS PARA LAS ENCUESTAS Y ENTREVISTAS

La herramienta que usó para diligenciar las encuestas fue Google forms herramienta enfocada en la creación de formularios de forma rápida y sencilla, aparte de su funcionalidad principal esta también provee de un análisis detallado de todas las respuestas digitadas a la población a la que se haya distribuido la encuesta, así mismo se puede generar un archivo formato Excel para tener estás respuestas a la mano.

Para las entrevistas se optó por grabar las respuestas de las entrevistas y hacer una transcripción del texto en Google docs. Esto con el fin de hacer un análisis de las necesidades y deseos planteados por los entrevistados.

La redacción de las preguntas para la encuesta y las entrevistas se hizo de manera conjunta haciendo uso igualmente de Google docs. Ya que permite trabajar desde 2 estaciones diferentes en el mismo documento.

La encuesta se realizó en dos fases en una se preguntó por todos los datos demográficos con el fin de identificar mejor a la población objetivo, y en la siguiente se preguntó sobre enfermedades que más afectan al cultivo, tratamientos que conocen y manejan, se les preguntó que si en algún momento han manejado una aplicación con el mismo fin o uno similar.

En las entrevistas se tomó en cuenta que las personas a entrevistar tuvieran una relación directa con el cultivo y que tuviesen experiencia con el manejo de enfermedades. Por lo tanto las preguntas se enfocaron en problemas específicos del cacao en relación con las enfermedades, como se ve afectado por las mismas, que parte de la planta ataca, cuáles son las más vulnerables, se tuvo en cuenta que al conocer más del cacao podrían dar una opinión acerca del contenido que debería tener la aplicación con base en esto se diseñaron preguntas como:

- ¿Cuándo el cacao se ve afectado por la monilia cuales con los primeros síntomas que presenta?
- ¿Qué procesos o controles son efectivos para combatir a la monilia?

Para más información de la encuesta y entrevistas revisar *Anexo A Encuestas y Entrevistas a Personal con cercano con el manejo del cultivo de cacao*

6.2 ENCUESTAS Y ENTREVISTAS A AGRICULTORES Y ENTES AGRICULTORES

6.3 SEGMENTACIÓN DE DESEOS Y NECESIDADES

A partir de las encuestas y las entrevistas realizadas se evidencian deseos y necesidades del sector cacaotero tanto desde el punto de vista agricultor como de expertos del tema, cabe recalcar que se encuentran varios puntos los cuales se explayaran a continuación:

- Una necesidad intrínseca del sector cacaotero es la información de controles básicos del cultivo de cacao, como lo explicaban los expertos muchos agricultores no hacen sus labores básicas en el cultivo esto perjudica enormemente el cultivo por lo tanto es información a tener en cuenta para el desarrollo de la solución.
- El apartado de controles también se nota como una necesidad latente del sector, por esto mismo se hará hincapié en varios tipos de controles para el cacao.
- Descripciones generales del cultivo de cacao son importantes para contextualizar no solo a los agricultores si no al público en general porque, aunque el foco de la solución es el sector cacaotero, no se puede dejar de lado de que publico general también pueda hacer uso de la solución.

- Cabe destacar que a lo largo de las entrevistas los entrevistados nos manifestaron su preocupación por la interacción entre agricultores del sector cacaotero, porque, aunque no se presenta en la zona objetivo estos mismo solo se comunican entre esta zona y desearían poder tener una interacción mas a profundidad con agricultores de otras zonas.

6.4 PLANTEAMIENTO DE REQUERIMIENTOS FUNCIONALES Y NO FUNCIONALES

Para la redacción y planteamiento de los requerimientos funcionales se tuvieron en cuenta las encuestas y entrevistas realizadas por el equipo de trabajo, así como los deseos y necesidades observados en los mismos y también tomando ideas de algunos de los artículos adjuntados en el estado del arte tales como “*Iterative development of Vegethon: a theory-based mobile app intervention to increase vegetable consumption*” (S. A. Mumma et al., 2016) y su complemento “*Effect of a mobile app intervention on vegetable consumption in overweight adults: a randomized controlled trial*” (S. Mumma et al., 2017) que habla tanto del desarrollo de la aplicación móvil como de su uso posterior y observaciones que se pueden usar para el planteamiento de requerimientos.

A partir de esto se generó un acta de requerimientos que la cual comprende una corta descripción del proyecto, metas del documento y a su vez el alcance del mismo.

A continuación, se mostrará algunas de las metas planteadas por el equipo de trabajo:

- Determinar todos los requisitos y funcionalidades a desarrollar de la arquitectura de software propuesta:
 - Determinar requerimientos funcionales y no funcionales del web admin
 - Determinar requerimientos funcionales y no funcionales de la aplicación móvil
 - Determinar requerimientos funcionales y no funcionales del API

En base a las metas principales del documento se recolectaron los siguientes requerimientos:

18 requerimientos no funcionales todos enfocados al buen funcionamiento de la arquitectura comprende tanto disponibilidad de las herramientas como fluidez de la misma, 4 de los 18 requerimientos no funcionales se comparten para los 3 sistemas de la arquitectura, estos son:

- El sistema debe tener una disponibilidad del 99,99% de las veces en que un usuario intente acceder.
- El tiempo para iniciar o reiniciar el sistema no podrá ser mayor a 1 minutos.
- La tasa de tiempos de falla del sistema no podrá ser mayor al 0,5% del tiempo de operación total.
- Incluir datos de contacto directo con un equipo de desarrollo.

De los 14 restantes 4 hacen parte del sistema de administración web y 10 de la aplicación móvil, a continuación, una muestra de algunos de ellos:

Admin web:

- El sistema debe soportar la gestión y almacenamiento de la información de varios usuarios admin
- La aplicación deberá consumir menos de 500 Mb de memoria RAM
- Los tiempos de espera para actualizar información no debe ser mayor 3 segundos
- Los formularios para agregar información una vez finalizados no deben tomar más de 5 segundos para agregar la información a la base de datos

Aplicación móvil:

- La aplicación móvil tendrá una interfaz fácil de usar para los usuarios, y coherente visualmente con la plataforma.
- La aplicación móvil será escalable para poder añadir más cultivos o enfermedades agregando nuevas casillas y funcionalidades.
- Tendrá acceso desde cualquier dispositivo móvil con sistema Android que tenga acceso a internet utilizando con datos móviles o por medio de redes wifi entre otras.
- El sistema será desarrollado únicamente para dispositivos con sistema operativo Android.

En materia de los requerimientos funcionales se identificaron 5 para el admin web, 5 para el API rest y 9 para la aplicación móvil para un total de 19 requerimientos funcionales, algunos ejemplos de ellos son:

Admin web:

- El sitio web debe tener un login donde solamente los administradores del sitio podrán ingresar para poder hacer una inspección a la información que se ha agregado o se quisiese añadir.
- El sitio web debe permitir agregar, editar o eliminar varios cultivos, pero se iniciará con el cacao, este debe contar con un esquema definido, el cual estará compuesto por datos básicos como: nombre del cultivo, nombre científico del cultivo, descripción del cultivo, ambiente de crecimiento y sus variedades, esto con el objetivo de tener una arquitectura flexible a la hora de querer agregar más cultivos.
- El sitio web debe permitir agregar enfermedades esta contará con una relación directa a los cultivos, si no existe relación con algún cultivo no se podrá agregar la enfermedad (Ejemplo: Monilia afecta al cacao), las enfermedades agregadas deben tener unos datos básicos como: Nombre de la enfermedad, nombre científico, cultivo al que ataca, descripción de la enfermedad, daños hacia el cultivo (Se debe ser específico en el hecho de categorizar en donde se presenta la enfermedad, tallos, flores, hojas, entre otros) (cabe aclarar que solo se podrá relacionar una enfermedad a un cultivo si hay una enfermedad igual que afecta a otro cultivo se tendría que registrar como la misma enfermedad pero afectando a otro cultivo)

API rest:

- El api debe tener múltiples conexiones tanto a la aplicación, como el admin web y la base de datos está misma debe asegurarse de que la conexión sea segura por ambas partes
- El api debe tener funciones que conecten a la aplicación con el admin web haciendo que este admin web provea información a la aplicación estas funciones no pueden actualizar la información de la base de datos solo solicitan datos de una a la otra
- Las funciones que tenga este api deben ser explícitas en cuanto a su nombramiento por ejemplo (Listado cultivos, listado de enfermedades, listado de controles, entre otros) debería traer a la aplicación todos los cultivos que están registrados en la página web, si los cultivos o en este caso la

información ya está almacenada localmente en la aplicación no se bajará ningún dato.

Aplicación móvil:

- La aplicación en su inicio debe mostrar un listado de los cultivos disponibles en la aplicación siendo el principal el cacao, este listado debe tener un ítem tipo card para cada cultivo, cada ítem debe mostrar el nombre del cultivo acompañado por una pequeña miniatura y una pequeña descripción plegable.
- Cuando se acceda al cultivo seleccionado la pantalla tendrá que mostrar el nombre del cultivo, el nombre científico, descripción, imágenes ilustrativas, variedades de cultivo y aparte de esto se mostrarán las enfermedades que afectan a este cultivo.
- La aplicación móvil debe tener un apartado de sugerencias para que los usuarios puedan escribir qué mejoras se le pueden hacer a la aplicación, este apartado de sugerencias sólo pedirá el nombre de la persona y la sugerencia correspondiente, esta sugerencia será almacenada en la base de datos del proyecto y posteriormente será mostrada en la página web para qué los administradores la tengan en cuenta en dado caso de que haya que realizar alguna mejora.

Se podrá encontrar más información y detalles en los anexos como *Anexo B Especificación de requerimientos de software*

7. DISEÑO DIAGRAMAS, WIREFRAMES E INTERFACES

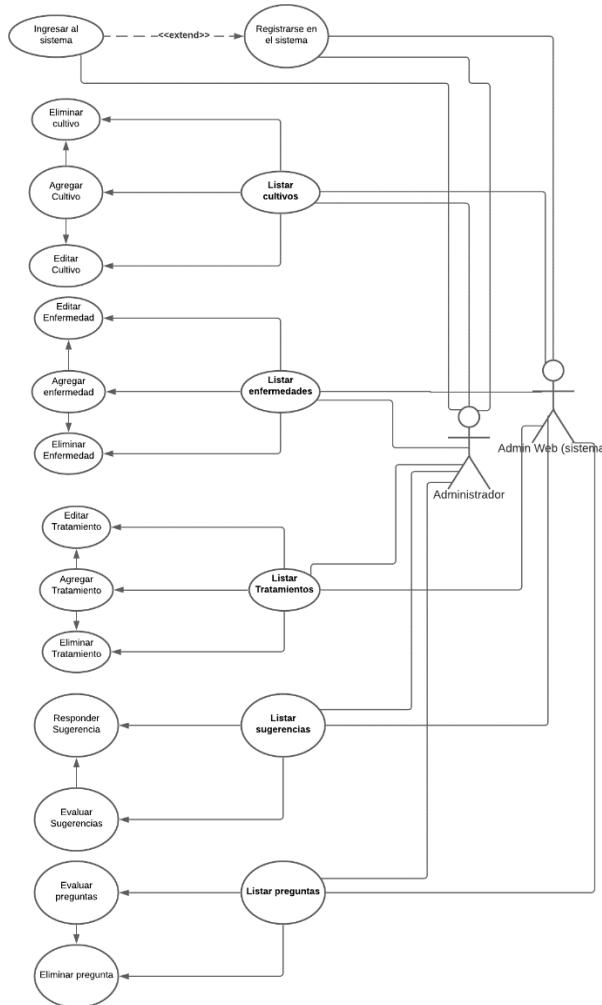
Para el diseño de diagramas de casos de uso, clases, bases de datos, arquitectura y wireframes se usaron herramientas como Lucidchart alojada en la web especializada en gran cantidad de diseños, las plantillas predeterminadas alojadas en este sitio web fueron idóneas para los diagramas desarrollados, otra de estas herramientas fue Figma su foco principal es la creación de wireframes, mockups e interfaces para cualquier plataforma, uno de sus fuertes y por la cual se escogió esta herramienta es porque permite el trabajo colaborativo y el prototipado de la aplicación móvil de este modo se puede tener un acercamiento al producto final directamente desde la maquetación.

7.1 DISEÑO DE DIAGRAMAS DE CASOS DE USO Y DIAGRAMAS DE CLASES

Para el diseño de diagramas de casos de uso y diagramas de clase se tomaron en cuenta los requerimientos planteados en la sección anterior y de acuerdo a estos se formularon una serie de procesos que debe poseer el sistema para su debido funcionamiento.

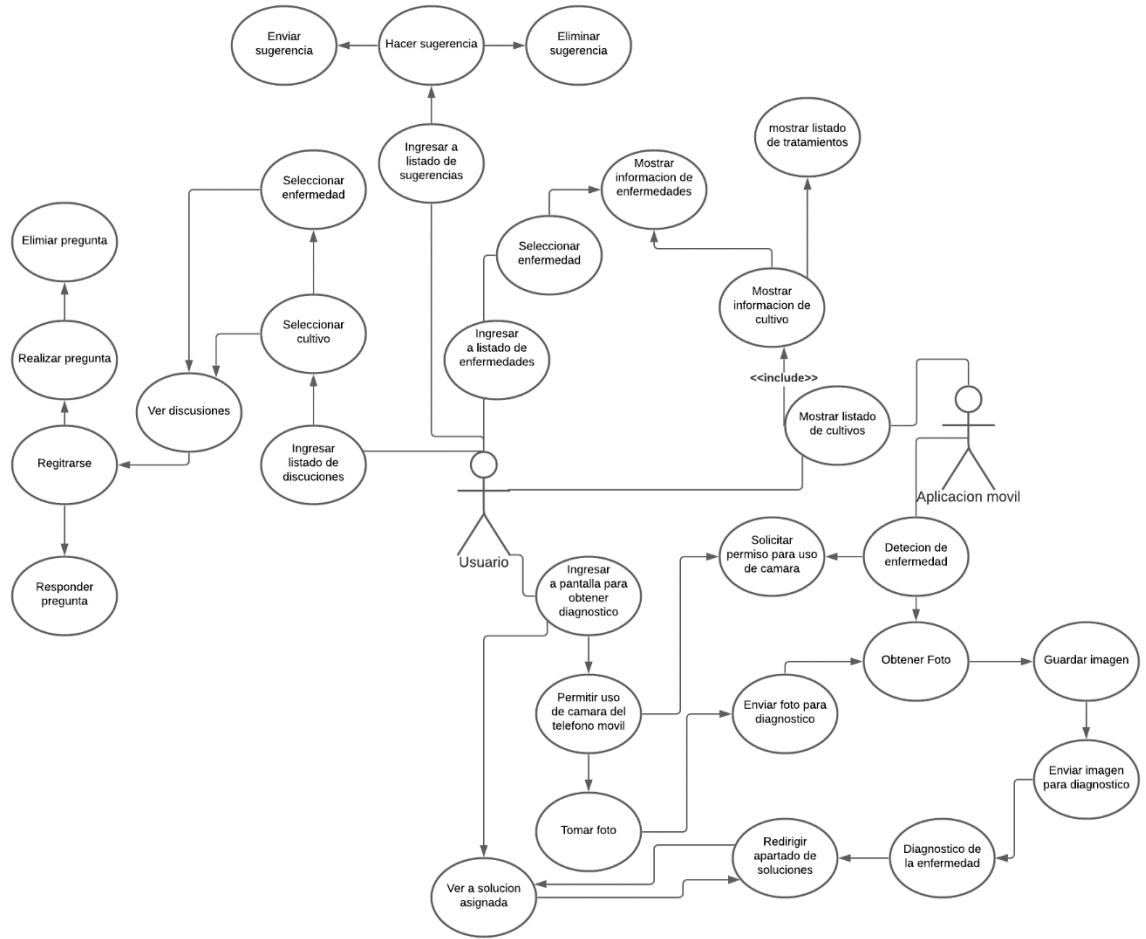
Para los diagramas de casos de uso se identificaron 2 actores principales como lo son el usuario y el administrador que hacen parte integral del sistema y 2 actores del sistema como el admin web y la aplicación móvil, por esto mismo se presentan los siguientes diagramas:

Figura 11 Casos de uso admin web



Fuente: Autor

Figura 12 Casos de uso aplicación móvil

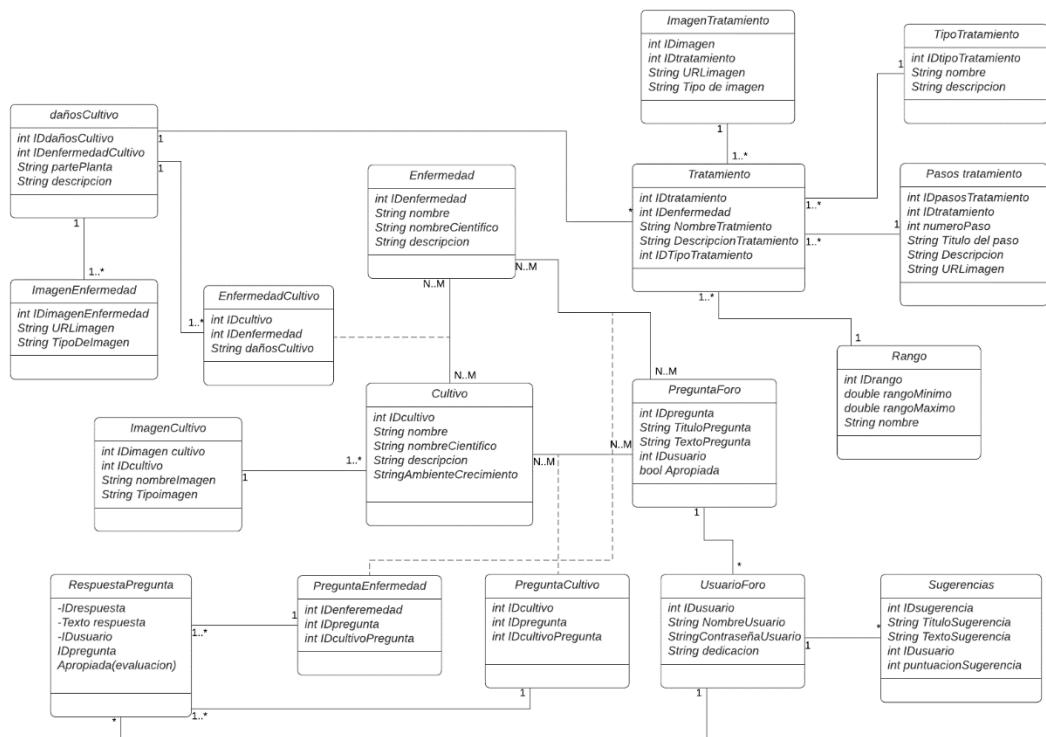


Fuente: Autor

A partir de estos diagramas de casos de uso se generó un documento especificando quienes actúan sobre ellos y como se desempeñan cada uno de los mismos, este documento lo podrás encontrar en el apartado de anexos como *Anexo C Documento de casos de uso*

En cuanto al diagrama de clase se diseñaron 17 diferentes clases las cuales ayudaran al buen funcionamiento de la aplicación, estas clases toman como base las necesidades identificadas por el grupo de trabajo y los requerimientos recogidos por los mismos, a su vez se pensó de una forma la cual agilice la recolección de información dentro de la aplicación para que esta responda de manera rápida a cualquier solicitud de datos por parte de los usuarios finales.

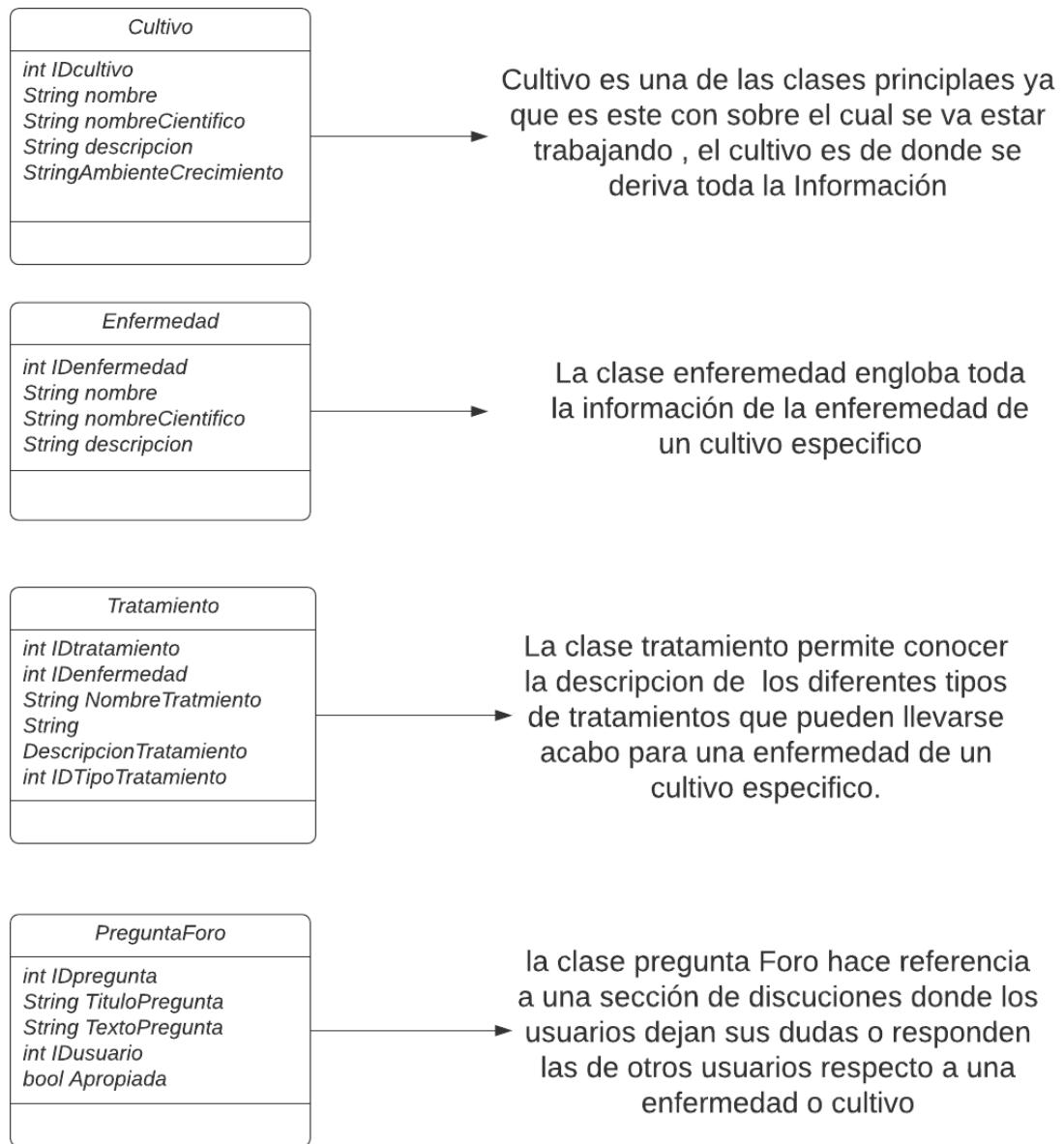
Figura 13 Diagrama de clases IAcultivar



Fuente: Autor

Dentro de estas 17 clases se pueden seleccionar algunas que son básicas o principales las cuales destacan porque es de estas de donde se deriva las demás clases y nos brindan la información requerida para el correcto funcionamiento de la aplicación.

Figura 14 Explicación clases principales

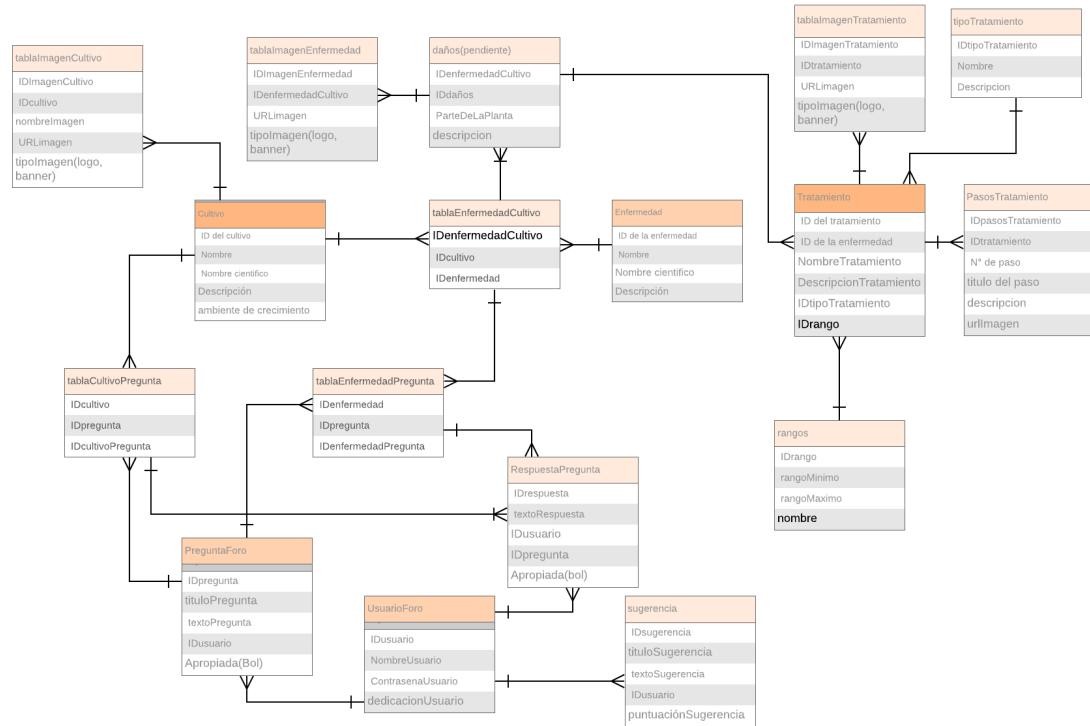


Fuente: Autor

7.2 DISEÑO DIAGRAMA DE BASE DE DATOS

Siguiendo el diagrama de clases desarrollado en el apartado anterior, se diseñó la base de datos tomando en cuenta las relaciones establecidas en el diagrama de clases así mismo los tipos de datos y la generación de tablas extra debido a relaciones muchos a muchos.

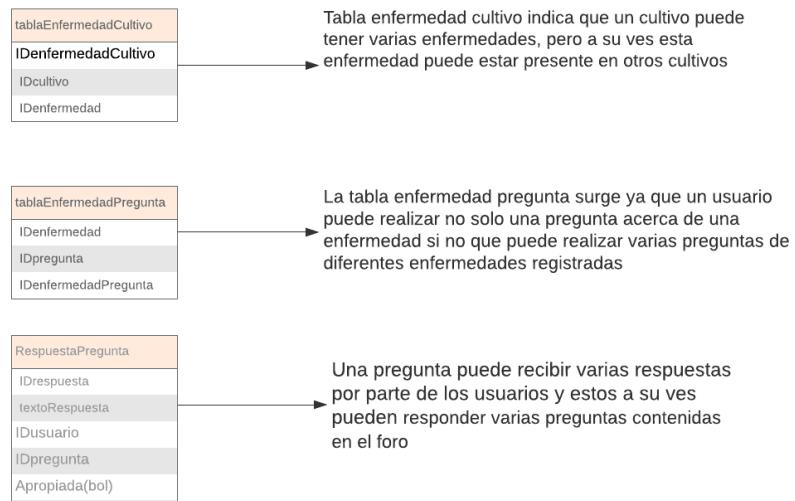
Figura 15 Diagrama de bases de datos



Fuente: Autor

Una relación de muchos a muchos se produce cuando varios registros de una tabla se asocian a varios registros de otra tabla, y esto genera una nueva tabla que recopila la información de ambas tablas a continuación, se muestran las tablas que surgen de estas relaciones.

Figura 16 Explicación relaciones muchos a muchos



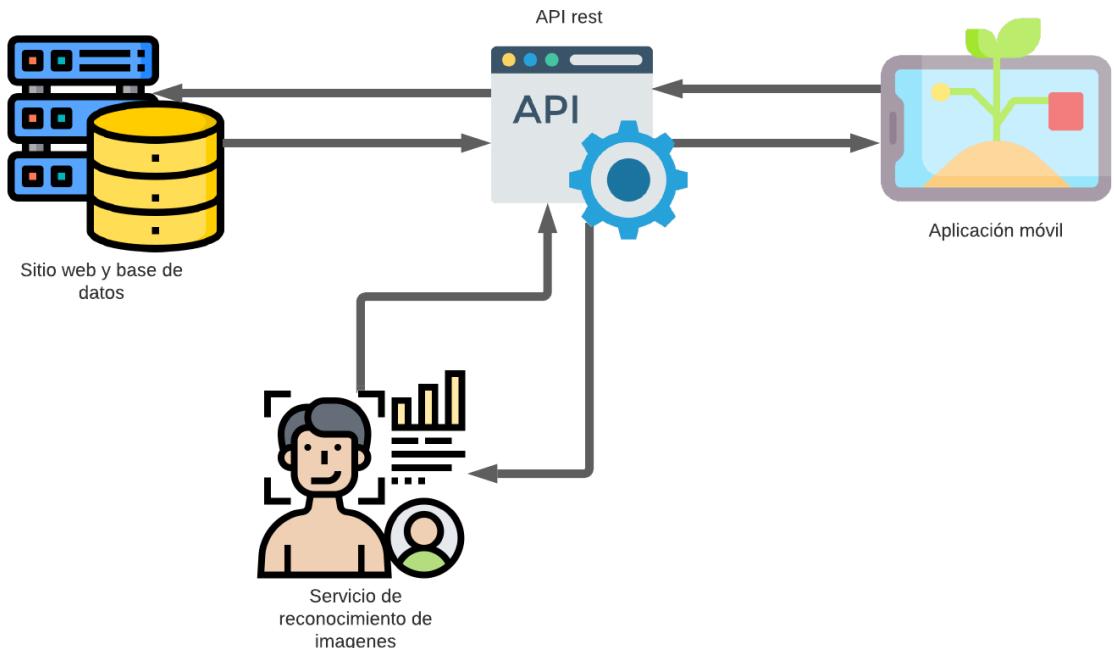
Fuente: Autor

7.3 DISEÑO DEL DIAGRAMA DE ARQUITECTURA

Para el diseño del diagrama de arquitectura se tomó en cuenta los 3 diferentes sistemas que hacen parte de la arquitectura desarrollada, a lo largo tanto del diseño de arquitectura como el documento que se genera de este mismo se establecen una serie de objetivos a cumplir para dar por terminado este punto, no sin olvidar el alcance del mismo documento y algunas nociones básicas para entenderlo como una pequeña descripción de cada uno de los componentes a desarrollar y las plataformas que se usarán para ello.

- Evidenciar el software qué se usará para la creación del sistema
- Documentar los procesos de desarrollo del proyecto.
- Dar una perspectiva del sistema en general

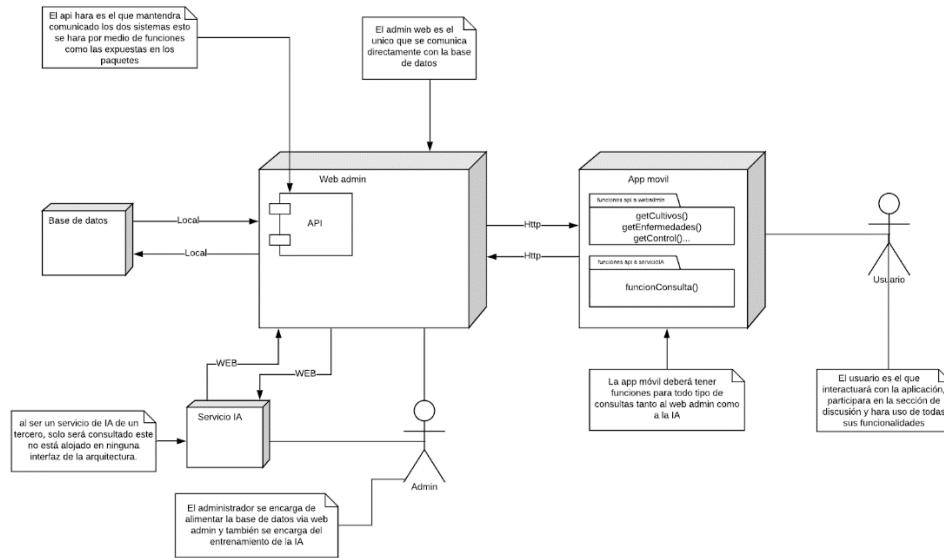
Figura 17 Vista general de la arquitectura



Fuente: Autor

Así mismo se diseñó un diagrama de despliegue donde se detalla a profundidad como se comunican los sistemas, el web admin será el sistema principal donde se provee datos al API y se hacen consultas a la base de datos, el API será la capa que protegerá el sistema admin ya que todos los sistemas desarrollados tendrán que solicitarle datos a este, la base de datos es de los elementos aislados de los demás sistemas ya que este únicamente se comunicara con el web admin, a su vez esta administrado por los desarrolladores o también denominados administradores, por último la aplicación móvil se comunicara exclusivamente por consultas http para extraer la información del API o solicitarle a este que se comunique con el servicio de reconocimiento de imágenes, este diagrama también muestra la interacción que tendrán los actores como el administrador y el usuario en la arquitectura a desarrollada, para más información puede consultar el *Anexo D Documento de arquitectura*

Figura 18 Diagrama de despliegue



Fuente: Autor

7.4 CREACIÓN DE WIREFRAMES E INTERFACES

Para la creación de wireframes e interfaces se usó la herramienta *Figma*, esta herramienta fue seleccionada por dos razones, la primera es el trabajo colaborativo ya que permite asignar integrantes a un proyecto en específico y que estos dos puedan trabajar al mismo tiempo y la segunda su poderosa herramienta de prototipado que permite enlazar cualquier parte de la interfaz a otras pantallas de esa manera se puede simular el resultado esperado de la aplicación.

Tomando en cuenta que la población objetivo es tiene un nivel socioeconómico medio-bajo y se encuentran en zonas de poca cultura móvil, el grupo de trabajo decidió en la sección de requerimientos crear interfaces sencillas, por esto mismo se tomaron en cuenta los principios y fundamentos del “UI design” o Diseño de interfaces de usuario del artículo “*Interfaz y experiencia de usuario: parámetros importantes para un diseño efectivo*” (Acosta, 2017) el cual aborda el diseño centrado en el usuario y la importancia que este concepto tiene para la construcción de interfaces claras que cubren las necesidad latentes del usuario final, así mismo se habla de usar solo lo necesario dentro del diseño de las interfaces, por eso mismo se decidió crear interfaces claras y agradables al usuario final, haciendo uso de fondos claros y colores que no confundan al usuario, las fuentes usadas son grandes para que de esta manera se pueda leer claramente el contenido de la

aplicación, interfaces que interactúan con el usuario como los botones usan colores más intensos para que este pueda diferenciarlas del contenido general.

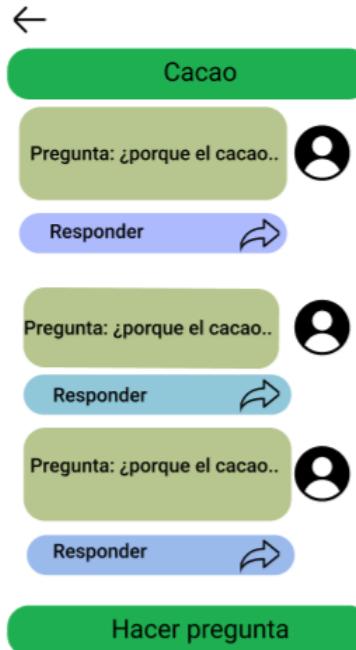
Se diseñaron un total de 23 pantallas para la aplicación móvil las cuales satisfacen todos los contenidos planteados en los requerimientos de la aplicación. Las cuales puede observar en el *Anexo E Interfaces*

Figura 19 Interfaces móviles



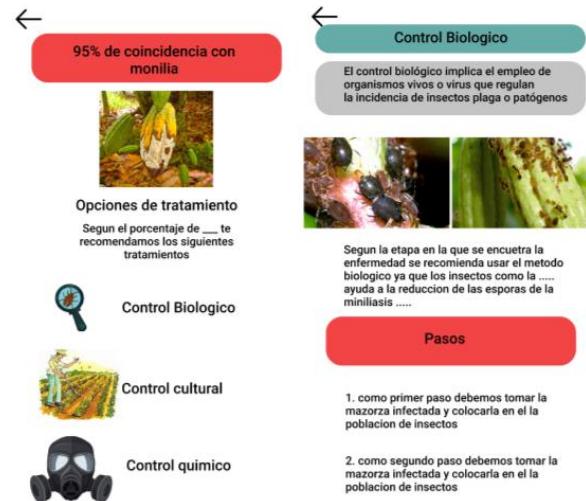
Fuente: Autor

Figura 20 Interfaces móviles



Fuente: Autor

Figura 21 Interfaces móviles



Fuente: Autor

8. DESARROLLO DE ADMIN WEB, API REST, ENTRENAMIENTO DE LA IA Y APLICACIÓN MÓVIL

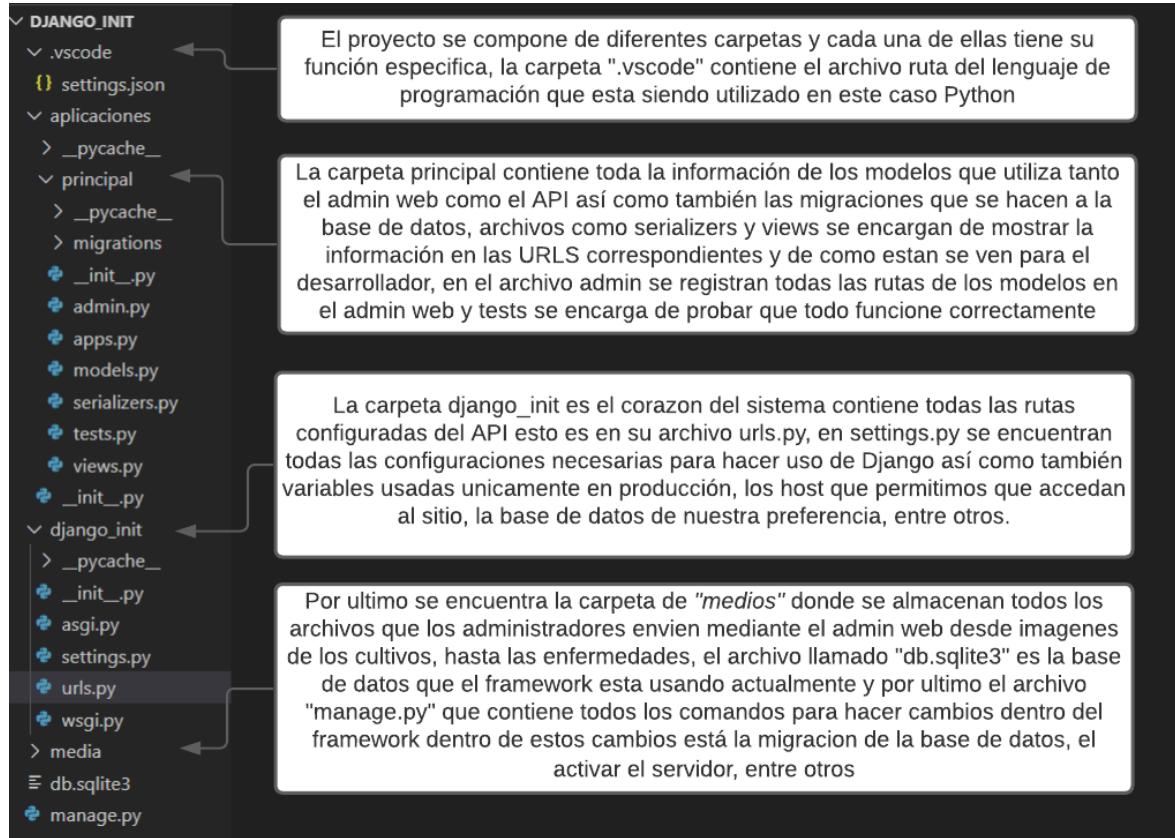
El desarrollo de los diferentes sistemas se llevó a cabo con diferentes frameworks, plataformas y lenguajes de programación.

8.1 DESARROLLO DE ADMIN WEB Y API REST

Para el desarrollo del admin web se usó el framework Django que usa como base el lenguaje de programación Python, se escogió este framework por su fácil acceso e interfaces integradas que ayudan a una programación ágil del admin web, uno de los puntos clave por lo que también se escogió Django por encima de otros frameworks es su gran cantidad de librerías para el API rest ya que es uno de los sistemas mas importantes de la arquitectura a desarrollar sin dejar de lado la gran documentación y comunidad que este framework posee a lo largo del mundo, lo cual permite que cualquier error o duda que se tenga fácilmente sea consultada y resuelta en un tramo de tiempo corto.

Para el admin web se codificaron un total de 20 modelos, estos modelos trabajados en Django tienen la ventaja de directamente tener una interfaz relacionada al mismo por lo tanto se ahorra el tiempo de desarrollar una interfaz para cada modelo, así mismo estos contienen un formulario automático y a su vez un CRUD (Create, Read, Update, Delete), lo cual es clave para el manejo de esta información. El proyecto dispone de un directorio extenso donde se pueden encontrar los diferentes archivos y carpetas que hacen parte integral del mismo.

Figura 22 Explicación estructura proyecto Django



Fuente: Autor

Los modelos de mas importancia en el proyecto son los relacionados directamente con el propósito de la aplicación como lo es el modelo de *cultivo* el cual comprende los diferentes campos necesarios para dar una descripción del mismo, también está el modelo *enfermedad* que cumple con la misma función del cultivo, cuando se unen estos dos modelos en una sola tabla da como resultado el modelo *enfermedadCultivo* y este representa la enfermedad que puede afectar al cultivo, teniendo esta relación se ahorran pasos innecesarios para acceder a la información, lo cual hace que tanto el sitio web como la aplicación usen menos recursos.

Figura 23 Modelos Django

```
class TimeStampedModel(models.Model):
    created = models.DateTimeField(auto_now_add=True)
    modified = models.DateTimeField(auto_now=True)

    class Meta:
        abstract = True

class Cultivo(TimeStampedModel):
    nombre = models.CharField(max_length=200)
    nombreCientifico = models.CharField(max_length=200)
    descripcion = models.CharField(max_length=10000)
    ambienteCrecimiento = models.CharField(max_length=200)

    def __str__(self):
        return self.nombre

class Enfermedad(TimeStampedModel):
    nombre = models.CharField(max_length=200)
    nombreCientifico = models.CharField(max_length=200)
    descripcion = models.CharField(max_length=10000)

    def __str__(self):
        return self.nombre

class EnfermedadCultivo(TimeStampedModel):
    cultivo = models.ForeignKey(Cultivo, on_delete=models.CASCADE)
    enfermedad = models.ForeignKey(Enfermedad, on_delete=models.CASCADE)
```

Fuente: Autor

A su vez se creó un modelo que hereda a todos los modelos e incluye la fecha de creación y de modificación, esto con el objetivo de llevar un seguimiento de todos los datos que se incluyan en la aplicación.

El admin web se compone de una variedad de interfaces las cuales hacen que la gestión de la información sea sencilla de manejar, un ejemplo de esto es la ventana de administración principal la cual muestra 3 listados, el primero relacionado con los grupos y usuarios que pueden acceder al admin web, el segundo donde se encuentran todos los modelos creados previamente cada uno de los ítems en el listado tiene accesos rápidos para añadir o cambiar información de cada uno de los

modelos disponibles, por último se encuentra un historial de los cambios recientes realizados por los administradores.

Figura 24 Menú principal web admin

The screenshot shows the Django administration interface. At the top, there's a dark blue header bar with the text "Django administration". Below it, a white page has a dark blue navigation bar with the text "Site administration". The main content area is divided into several sections:

- AUTHENTICATION AND AUTHORIZATION**: Contains links for "Groups" and "Users", each with "Add" and "Change" buttons.
- PRINCIPAL**: Contains links for various models: "Cultivo preguntas", "Cultivos", "Danos enfermedad cultivos", "Enfermedad cultivos", "Enfermedad preguntas", "Enfermedads", "Pasos tratamientos", "Pregunta foros", and "Rangos". Each link has "Add" and "Change" buttons.
- Recent actions**: A list of recent administrative actions, each preceded by a green plus sign and a small icon:
 - Cacao (Monilia) - ¿La verdad es que no estoy seguro de como hacer control cultural a mi cacao alguien dispuesto a enseñarme? Enfermedad pregunta
 - Cacao Cultivo pregunta
 - Cacao Cultivo pregunta
 - Cacao Cultivo pregunta
 - ¿La verdad es que no estoy seguro de como hacer control cultural a mi cacao alguien dispuesto a enseñarme? Pregunta foro
 - ¿Quien me puede enseñar a remover mazorcas?

Fuente: Autor

Cuando se accede a alguno de los modelos listados previamente en el menú principal, se mostrara un listado del respectivo modelo, el listado es fácilmente configurable desde el código para que muestre los campos que deseen los administradores, en este caso se puede ver las preguntas echas por los usuarios de la aplicación, esta interfaz también posee una barra de acciones donde se puede seleccionar las preguntas para posteriormente eliminarlas, también se tiene acceso a un botón el cual nos llevara a una interfaz para poder agregar preguntas por cuenta propia.

Figura 25 Listado preguntas web admin

Select pregunta foro to change

Action: 0 of 4 selected

<input type="checkbox"/>	PREGUNTA FORO
<input type="checkbox"/>	¿La verdad es que no estoy seguro de como hacer control cultural a mi cacao alguien dispuesto a enseñarme?
<input type="checkbox"/>	¿Quien me puede enseñar a remover mazorcas?
<input type="checkbox"/>	¿Como deberia limpiar mi cacao?
<input type="checkbox"/>	¿Como planto cacao?

4 pregunta foros

Fuente: Autor

El formulario de cada modelo permite registrar información en la base de datos, este a su vez puede ser usado para editar información de cada uno de los campos una vez se haya accedido directamente a algún dato en específico, cabe destacar que desde esta interfaz también se puede eliminar el dato al que se accedió.

Figura 26 formulario cultivo web admin

Change tabla imagen cultivo

Cultivo:

Nombre:	<input type="text" value="Cacao bonito"/>
Tipo:	<input type="text" value="banner"/>
Imagen:	Currently: cultivos/cacao_sano.jpg <input type="button" value="Clear"/> Change: <input type="button" value="Examinar..."/> No se ha seleccionado ningún archivo.

Fuente: Autor

Para el desarrollo del API rest se utilizó la librería “*Django rest framework*” la cual permite que a partir de los modelos creados previamente para el web admin se puedan generar URL’s que contienen la información de las diferentes tablas de la base de datos sin comprometerlas, estas URL’s suministran toda la información en formato JSON el cual posteriormente es interpretado en la aplicación móvil para poblar el contenido de la misma.

Para poder generar el API de manera correcta se deben crear Serializers. Los serializers son clases que transforman los datos de formatos de Django como objetos que extienden de las clases como Model o querysets, en formatos usados

en la web tales como JSON y XML, de esta manera podemos hacer translación de datos rápidamente al formato requerido en este caso se usó JSON

Figura 27 Serializers Django

```
class CultivoSerializer(serializers.ModelSerializer):
    class Meta:
        model = Cultivo
        fields = ['id', 'nombre', 'nombreCientifico', 'descripcion', 'ambienteCrecimiento']

class EnfermedadSerializer(serializers.ModelSerializer):
    class Meta:
        model = Enfermedad
        fields = ['id', 'nombre', 'nombreCientifico', 'descripcion']

class TratamientoSerializer(serializers.ModelSerializer):
    class Meta:
        model = Tratamiento
        fields = ['id', 'nombre', 'descripcion', 'tipoDeTratamiento', 'enfermedadCultivo', 'rango']
        depth = 2
```

Fuente: Autor

El viewset es una clase que permite al desarrollador concentrarse en todo lo relacionado con el backend sin tener que preocuparse por la construcción de URL's necesarias para consultar el API, esto pasa por el hecho de que esta clase construye las URL's basada en convenciones comunes de la programación web. Por esto mismo se hizo uso del mismo, facilitando la creación de URL's ahorra tiempo y esfuerzo a la hora de codificar el API.

Figura 28 ViewSets Django

```
class CultivoViewSet(viewsets.ModelViewSet):
    serializer_class = CultivoSerializer
    queryset = Cultivo.objects.all()

class EnfermedadViewSet(viewsets.ModelViewSet):
    serializer_class = EnfermedadSerializer
    queryset = Enfermedad.objects.all()

class TratamientoViewSet(viewsets.ModelViewSet):
    serializer_class = TratamientoSerializer
    queryset = Tratamiento.objects.all()
```

Fuente: Autor

Después de configurar los viewsets se tienen que registrar todas las rutas que se desean tener en el API así mismo se le pasa el viewset que contiene la forma en como se mostrara la información en el frontend.

Figura 29 Routers Django

```
router = DefaultRouter()
router.register(r'cultivos', cultivoViewSet)
router.register(r'enfermedades', EnfermedadViewSet)
router.register(r'tratamientos', TratamientoViewSet)
router.register(r'pasosTratamiento', PasosTratamientoViewSet)
router.register(r'imagenCultivo', ImagenCultivoViewSet)
router.register(r'imagenEnfermedad', ImagenEnfermedadViewSet)
router.register(r'imagenTratamiento', ImagenTratamientoViewSet)
router.register(r'usuarioForo', UsuarioForoViewSet)
```

Fuente: autor

En el frontend se muestra cada una de las rutas establecidas en el router y estas rutan están distribuidas en una estructura diccionario su clave es el nombre del modelo y su contenido es la dirección de la misma.

Figura 30 API Root

Django REST framework skysaku

Api Root

OPTIONS GET

The default basic root view for DefaultRouter

GET /

```
HTTP 200 OK
Allow: GET, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

{
    "cultivos": "http://192.168.1.41:8000/cultivos/",
    "enfermedades": "http://192.168.1.41:8000/enfermedades/",
    "tratamientos": "http://192.168.1.41:8000/tratamientos/",
    "pasosTratamiento": "http://192.168.1.41:8000/pasosTratamiento/",
    "ImagenCultivo": "http://192.168.1.41:8000/ImagenCultivo/",
    "ImagenEnfermedad": "http://192.168.1.41:8000/ImagenEnfermedad/",
    "ImagenTratamiento": "http://192.168.1.41:8000/ImagenTratamiento/",
    "usuarioForo": "http://192.168.1.41:8000/usuarioForo/",
    "preguntaForo": "http://192.168.1.41:8000/preguntaForo/",
    "respuestaForo": "http://192.168.1.41:8000/respuestaForo/",
    "sugerencia": "http://192.168.1.41:8000/sugerencia/",
    "ImagenEnfermedadCultivo": "http://192.168.1.41:8000/ImagenEnfermedadCultivo/",
    "cultivoPregunta": "http://192.168.1.41:8000/cultivoPregunta/",
    "enfermedadPregunta": "http://192.168.1.41:8000/enfermedadPregunta/",
    "TratamientoPregunta": "http://192.168.1.41:8000/TratamientoPregunta/",
    "enfermedadCultivo": "http://192.168.1.41:8000/enfermedadCultivo/",
    "rango": "http://192.168.1.41:8000/rango/",
    "tipoTratamiento": "http://192.168.1.41:8000/tipoTratamiento/",
    "danosEnfermedadCultivo": "http://192.168.1.41:8000/danosEnfermedadCultivo/"
}
```

Fuente: Autor

Cuando se accede a alguna de estas rutas se mostrará la información de las tablas de la base de datos en formato JSON, se escogió el formato JSON por su facilidad de tratamiento de datos por fuera del ámbito web. cuando este JSON llegue a la aplicación será procesado por una serie de funciones para extraer su información.

Figura 31 JSON cultivo

```
HTTP 200 OK
Allow: GET, POST, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

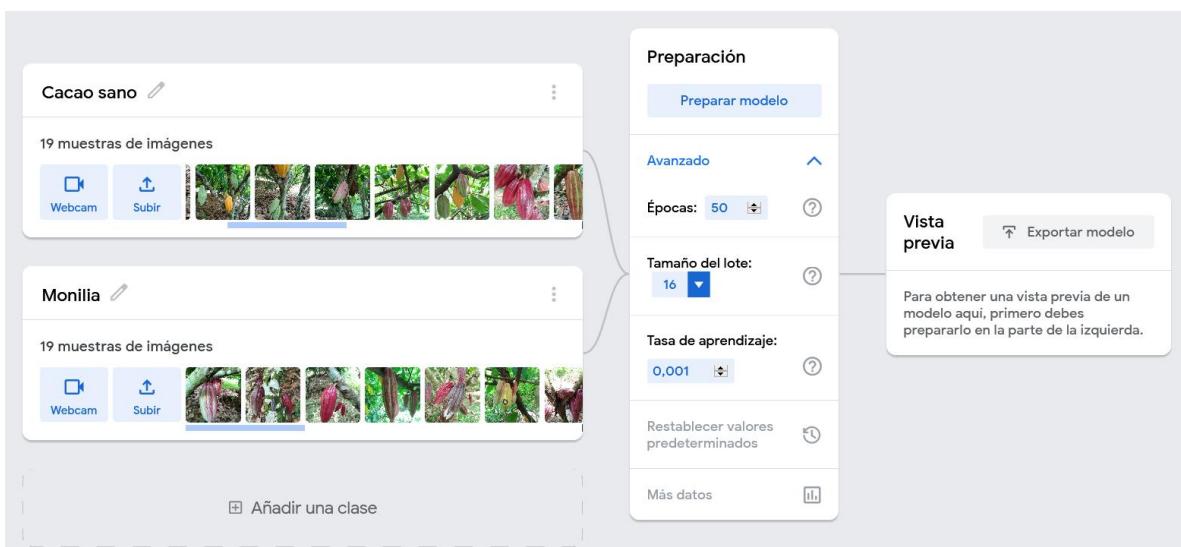
[
    {
        "id": 1,
        "nombre": "Cacao",
        "nombreCientifico": "Theobroma cacao",
        "descripcion": "árbol del cacao o cacaotero, nativo de regiones tropicales y subtropicales de America: América tropical, planta de hoja perenne",
        "ambienteCrecimiento": "Humedo, Templado"
    }
]
```

Fuente: Autor

8.2 ENTRENAMIENTO DEL MÓDULO DE RECONOCIMIENTO DE IMÁGENES

Para el entrenamiento del módulo de reconocimiento de imágenes se pusieron a prueba 2 servicios de terceros, el primero Teachable machines desarrollado por Google el cual permite preparar modelos de aprendizaje automático desde la comodidad del navegador, su uso es relativamente sencillo como primer paso se recopila y agrupa los elementos en una interfaz gráfica la cual pide que se agrupen estos en 2 clases diferentes o más, esto con el fin de separar los elementos y poder hacer una clasificación apropiada de los mismos, siguiente a esto se prepara el entrenamiento del modelo, el sitio web tiene unos valores predeterminados para este entrenamiento pero se puede acceder a ellos en el apartado de avanzado. Al estar en este apartado podemos variar diferentes valores como las épocas que controlan la cantidad de veces que se va a repetir el entrenamiento, el tamaño del lote que tiene como función distribuir la cantidad de fotos por entrenamiento o la taza de aprendizaje que afecta la velocidad en la que el algoritmo aprende todo esto puede ser cambiado allí, por último la exportación del modelo hace que se ejecute el entrenamiento del modelo para así obtener el resultado lo cual sería el modelo ya entrenado, este modelo se puede alojar directamente en la nube de Google o descargarlo para montarlo en donde el usuario o desarrollador desee.

Figura 32 Teachable machine entrenamiento

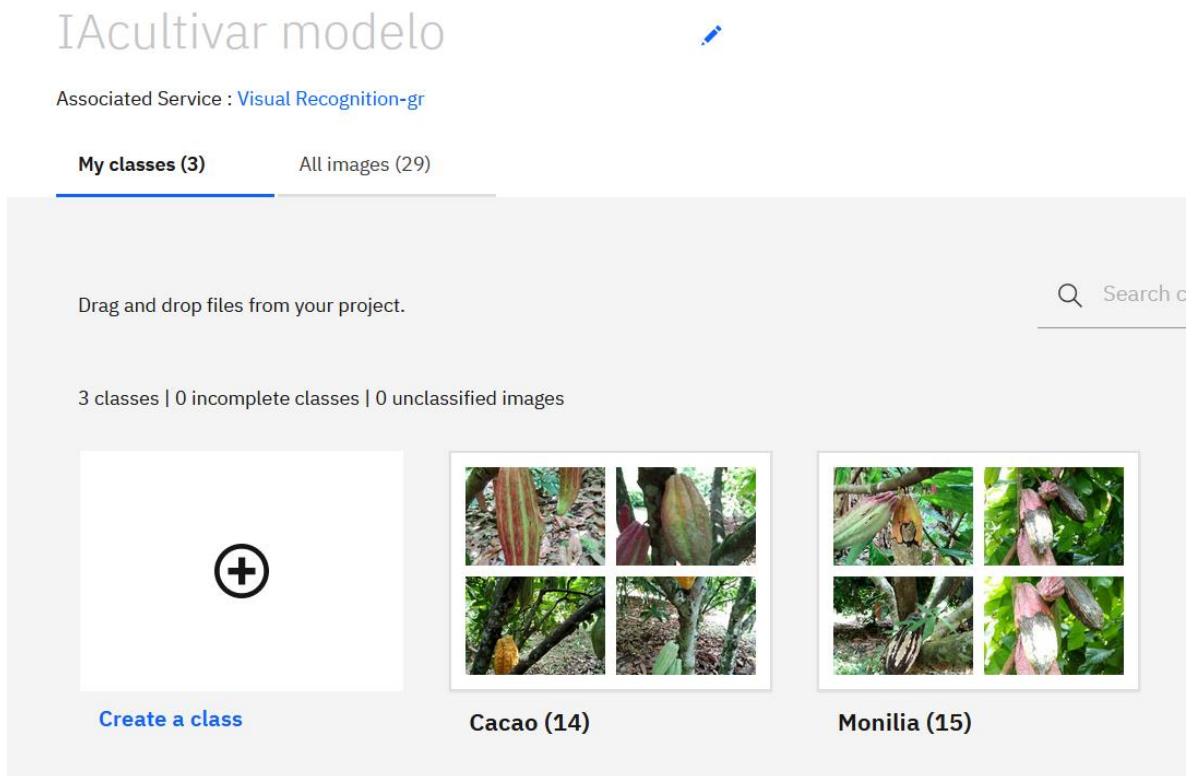


Fuente: Autor

La segunda opción es IBM Watson Studio desarrollado por IBM, permite hacer lo mismo que teachable machines, se crean dos clases para separar los objetos que se van a identificar, una vez creadas las clases con sus respectivas imágenes se puede entrenar el modelo.

Ya con el modelo entrenado se puede comenzar a hacer pruebas de su funcionamiento, se puede subir la cantidad de fotos que se desee para hacer estas pruebas.

Figura 33 IBM Watson entrenamiento



Para escoger el servicio a usar se hizo una tabla comparativa de los aspectos técnicos que el equipo de trabajo observó de los dos servicios

Tabla 16 Teachable machine VS IBM Watson

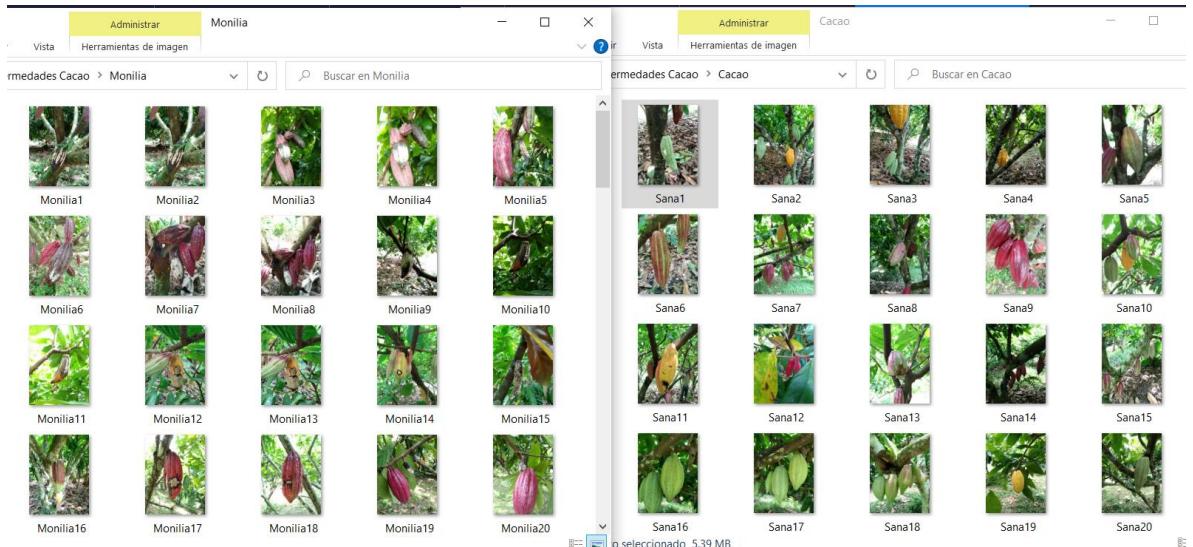
Tabla comparativa		
Aspecto técnico	Teachable machines	IBM watson
Capacidad de almacenamiento	Se puede conectar directamente con Google drive de esa manera se puede acceder a datos desde la nube, no tiene límites de en cuanto a la cantidad de información que se quiera almacenar en el modelo	Tiene un almacenamiento base de 250mb, mas allá del almacenamiento limite se debe pagar por el servicio, las imágenes deben ser subidas directamente desde el ordenador al sitio web, las imágenes quedan guardadas en la nube de IBM
Ajustabilidad	Permite modificar valores como épocas, tamaño del lote y tasa de aprendizaje	No permite hacer ninguna modificación al modelo
Facilidad de uso	Interfaz amigable e intuitiva, no se tiene que iniciar sesión en ningún servicio	Interfaz compleja para hacer uso de los modelos se debe iniciar sesión en diferentes servicios
Facilidad de implementación	Ofrece diferentes extensiones tales como Tensorflow.js el cual se puede implementar en JavaScript y P5.js, en Tensorflow normal con keras y para Android con Tensorflow lite	Ofrece implementación en diferentes lenguajes de programación como cURL, Java, Node, Python y Ruby, también tiene información extensa de cómo hacer la implementación a todos estos lenguajes
Seguridad	El modelo queda disponible públicamente para cualquiera que tenga la URL del modelo en la nube, los datos no se almacenan en los servidores de Google y cualquiera que tenga la URL del modelo no puede acceder a las imágenes del modelo.	Al ser almacenado directamente la plataforma de IBM es completamente aislado y únicamente el desarrollador puede acceder al modelo, lo cual lo hace una plataforma extremadamente segura.

Fuente: Autor

Se optó por usar teachable machines como servicio para el reconocimiento de imágenes ya que su capacidad de carga de información no tiene límites a comparación de IBM Watson y su vez teachable machines permite hacer modificaciones a la tasa de aprendizaje del modelo y las épocas a entrenar el modelo, para usarlo de manera apropiada se decidió alojarlo directamente en nuestro admin web, de esa forma se puede consultar de manera rápida y tener información a la mano.

Para el entrenamiento del modelo de reconocimiento de imágenes se hizo uso de un data set que contiene 100 imágenes del cacao y 105 imágenes de la monilia con esto es suficiente para el entrenamiento del modelo de reconocimiento.

Figura 34 Directorio de imágenes para entrenamiento



Fuente: Autor

En cuanto al entrenamiento del modelo, al variar los valores de entrenamiento cambia la precisión del modelo por esto mismo se hicieron una serie de cambios que pusieron a prueba la capacidad del servicio. Algo a tener en cuenta es que Teachable machines obtiene el valor de la precisión del modelo a partir de las imágenes que se le pasan al modelo, este toma el 85% de las fotos para hacer entrenamiento y el 15% restantes de las fotos las usa como fotos de prueba, por esto mismo no tenemos que hacer un data set independiente para probar el modelo, regresando al cambio de valores este se hizo de diferentes formas primero se varió el tamaño de los lotes por época con una tasa de aprendizaje fija y así mismo para la siguiente iteración.

Tabla 17 Tabla precisión tasa de aprendizaje 0,001

Precisión		
Tasa de aprendizaje: 0,001		
Valores	Cacao	Monilia
50 épocas 16 tamaño de lote	0,80	0,56
50 épocas 32 tamaño de lote	0,80	0,94
50 épocas 64 tamaño de lote	0,87	0,75
100 épocas 16 tamaño de lote	0,87	0,88
100 épocas 32 tamaño de lote	1,00	0,81
100 épocas 64 tamaño de lote	0,87	0,88
150 épocas 16 tamaño de lote	0,60	0,94

150 épocas 32 tamaño de lote	0,93	0,75
150 épocas 64 tamaño de lote	0,80	0,69
200 épocas 16 tamaño de lote	0,87	0,88
200 épocas 32 tamaño de lote	0,73	0,63
200 épocas 64 tamaño de lote	0,87	0,88

Fuente: Autor

Con una tasa de aprendizaje del 0,001 se observó que la precisión de predicción tiende a no variar demasiado con el data set del cacao a diferencia del data set de monilia donde 5 de los 12 modelos estuvieron por debajo del 75% de precisión.

Tabla 18 Tabla precisión tasa de aprendizaje 0,0001

Precisión		
Tasa de aprendizaje: 0,0001		
Valores	Cacao	Monilia
50 épocas 16 tamaño de lote	0,80	0,81
50 épocas 32 tamaño de lote	0,80	0,56
50 épocas 64 tamaño de lote	0,87	0,63
100 épocas 16 tamaño de lote	0,80	0,81
100 épocas 32 tamaño de lote	0,80	0,81
100 épocas 64 tamaño de lote	0,93	0,75
150 épocas 16 tamaño de lote	0,80	0,69
150 épocas 32 tamaño de lote	0,93	0,75
150 épocas 64 tamaño de lote	0,80	0,75
200 épocas 16 tamaño de lote	0,89	0,88
200 épocas 32 tamaño de lote	0,87	0,63
200 épocas 64 tamaño de lote	0,80	0,63

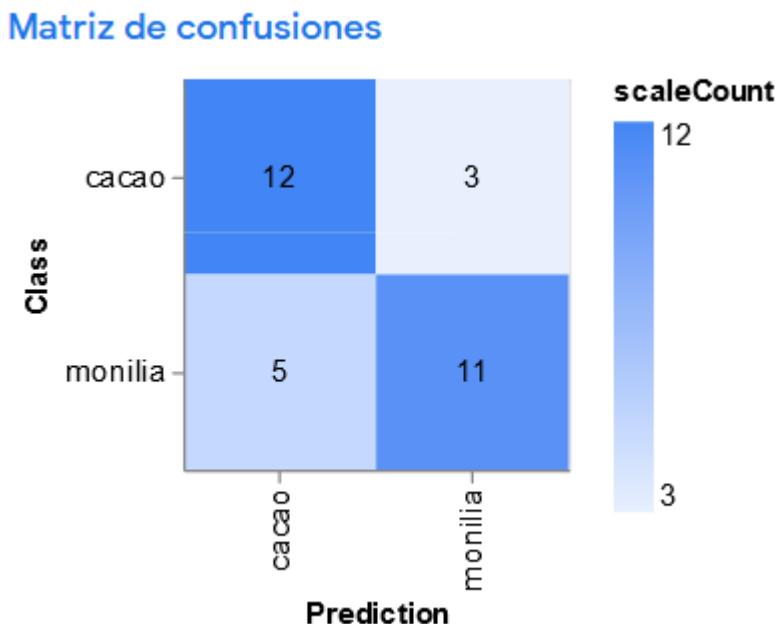
Fuente: Autor

Con una tasa de aprendizaje del 0,0001 se observó que la precisión de la predicción tiende a no variar demasiado con el data set del cacao, si se compara con la tasa de aprendizaje del 0,001 de ese mismo data set se denota una variabilidad muy mínima en los valores ya que ninguno de los modelos estuvo por debajo del 75% de precisión, en cambio las variaciones en la monilia empeoraron de manera drástica al únicamente tener 4 modelos por encima del 75% de precisión, uno de estos y el mas destacable es el de 200 épocas y 16 de tamaño del lote al ser el modelo más alto de toda la tabla.

Por porcentaje de precisión hubo 4 modelos que estuvieron a la par, estos están en la tabla de tasa de aprendizaje del 0,001 estos son los de 100 y 200 épocas con tamaño de lote de 16 y 64, por parte la tabla con tasa de aprendizaje de 0,0001 se encontró un modelo ligeramente superior a los demás con 0,89 de precisión en cacao y 0,88 de precisión en monilia.

En el análisis de la matriz de confusiones del modelo con mejor precisión de todos, la función de esta matriz es la de indicar la cantidad de veces que se pasaron datos al algoritmo e hizo la predicción de manera apropiada. Para hacer pruebas de si la clase de cacao hacia la detección de manera apropiada se le pasaron 17 imágenes donde todas estas eran cacao, de las 17 solo 5 fueron detectadas como monilia y 12 de ellas como cacao, a su vez de 14 pruebas que se hicieron en la clase monilia, 11 veces predijo que era monilia y 3 veces predijo que era cacao. Se puede afirmar que se obtuvo un modelo que puede hacer predicciones con un porcentaje de precisión bastante alto y por lo tanto este modelo fue el escogido para hacer las predicciones en la aplicación móvil.

Figura 35 Matriz de confusiones mejor modelo



Fuente: Autor

8.3 DESARROLLO DE LA APLICACIÓN MÓVIL

El desarrollo de la aplicación móvil se hizo en el framework Flutter que usa de lenguaje base a Dart, este framework desarrollado por Google está orientado al uso de widgets como herramienta principal, esto hace que sea sencillo de aprender y de usar. Flutter cuenta con una documentación extensa de todos los componentes que hacen parte de él, así como una comunidad en crecimiento la cual hace grandes aportes al lenguaje, así mismo Google en su afán de impulsar este framework ha desarrollado tutoriales y proyectos que ayudan al aprendizaje del mismo. El gran

punto a favor de Flutter por encima de otros frameworks de desarrollo es su gran cantidad de librerías que ayudan al desarrollador en todos los ámbitos de su trabajo, para instalar Flutter en el equipo u ordenador se debe descargar el SDK (Software Development Kit) de Android para aplicaciones en general, instalar VScode (Visual Studio Code) y correr unas líneas de comando en el símbolo del sistema (CMD) esto con el fin de comprobar que todos los elementos necesarios para correr Flutter en el equipo estén debidamente instalados. Una vez comprobado esto se está listo para codificar con el framework Flutter.

Figura 36 Comandos flutter

```
C:\Users\Usuario>flutter doctor
Doctor summary (to see all details, run flutter doctor -v):
[✓] Flutter (Channel stable, 1.22.1, on Microsoft Windows [VersiÃ³n 10.0.19041.572], locale es-CO)

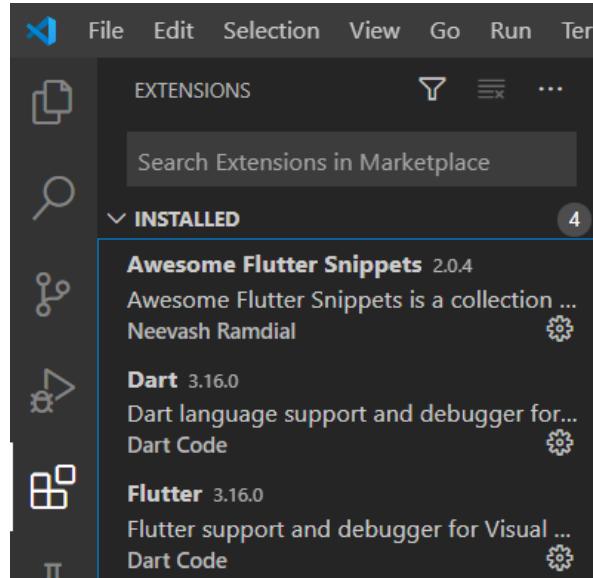
[✓] Android toolchain - develop for Android devices (Android SDK version 30.0.2)
[!] Android Studio (not installed)
[✓] VS Code (version 1.50.1)
[!] Connected device
    ! No devices available

! Doctor found issues in 2 categories.
```

Fuente: Autor

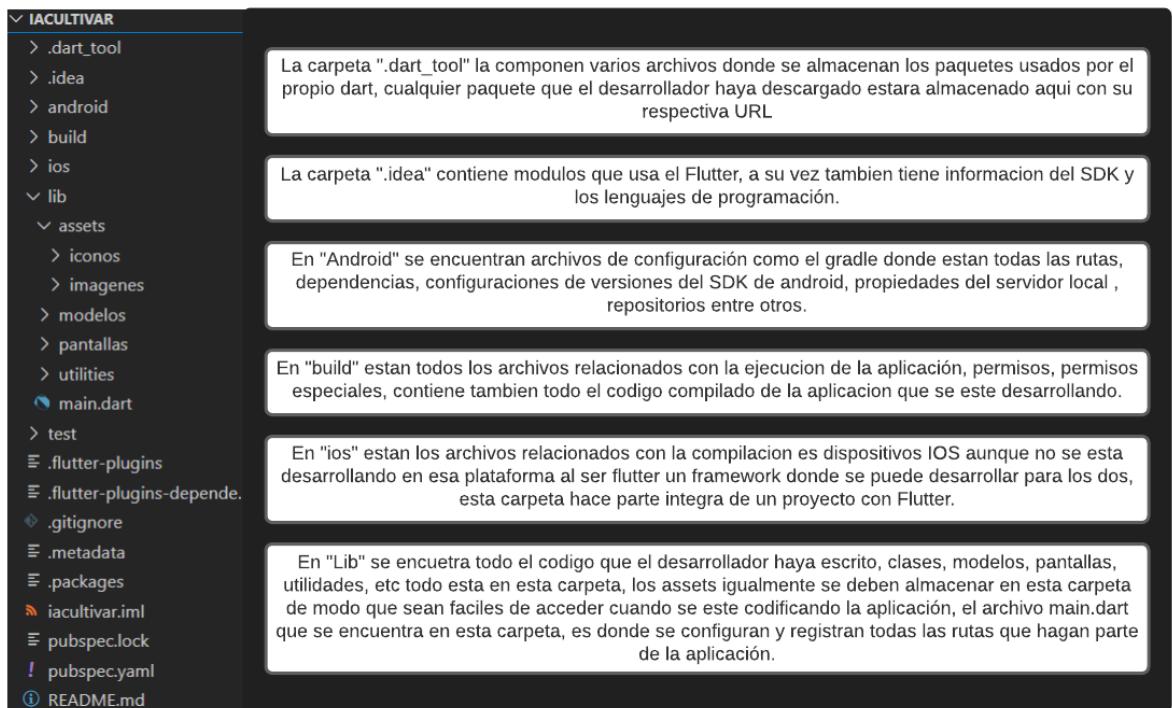
El editor de texto de código que se usó fue VScode (Visual Studio Code), plataforma que ha tenido un crecimiento exponencial en los últimos años ya que soporta la mayoría de lenguajes de programación, este a su vez posee con una gran cantidad de extensiones que ayudan al desarrollador a escribir código rápidamente. Para el desarrollo de la aplicación se instalaron las extensiones Flutter y Dart necesarias para poder codificar la aplicación y la extensión Awesome Flutter Snippets para generar bloques de código básicos como estructuras, constructores, entre otros.

Figura 37 Extensiones VScode



Fuente: Autor

Figura 38 Explicación directorio aplicación móvil



Fuente: Autor

Para la construcción de las interfaces se tomó un enfoque orientado a funciones, este enfoque orientado a funciones parte de la premisa de programar funciones con pequeñas partes de la interfaz y que una vez estas sean programadas por completo se inserten dentro del cuerpo de la app, esto se hace con el fin de tener un código organizado y fácil de depurar a la hora de hacer revisiones de errores.

En cuanto a la interfaz se trató de que no se vea opacada por la barra superior del teléfono por esto mismo se decidió usar el Widget SafeArea en todas las pantallas, este widget hace que la barra superior quede siempre fija y el contenido de la app empiece después de que se construye esta barra, así no se pierde información o espacio.

Para evitar desproporciones en diferentes dispositivos móviles se optó por usar la función *MediaQuery* la cual nos permite adaptar todo el contenido de la aplicación a medida que cambien los tamaños de los dispositivos móviles donde se instale la misma, esto hace que los usuarios en diferentes dispositivos no pierdan la experiencia de una interfaz amigable y sencilla hacia ellos.

Figura 39 Código e interfaz aplicación

```
@override  
Widget build(BuildContext context) {  
    return SafeArea(  
        child: Scaffold(  
            body: SingleChildScrollView(  
                child: Container(  
                    decoration: BoxDecoration(  
                        image: DecorationImage(  
                            image: AssetImage("lib/assets/imagenes/fondo1.jpg"),  
                            fit: BoxFit.cover)), // DecorationImage // BoxDecoration  
                    child: Column(  
                        children: [  
                            titulo(),  
                            card("lib/assets/iconos/camera.png", "Detección en vivo",  
                                context, "nada"),  
                            card("lib/assets/iconos/plant.png", "Cultivos", context,  
                                ListadoCultivos.listadoCultivos.toString()),  
                            card(  
                                "lib/assets/iconos/coronavirus.png",  
                                "Enfermedades",  
                                context,  
                                DescripcionEnfermedad.descripcionEnfermedad.toString()),  
                            botonesInferiores(context)  
                        ],  
                    ), // Column  
                ), // Container  
            ), // SingleChildScrollView  
        ), // Scaffold  
    ); // SafeArea
```



Fuente: Autor

La aplicación tiene funciones que le permiten hacer solicitudes http a el API desarrollada, para que estas funciones correctamente se debe tener una estructura de clases similar a la que esta en la API por esto mismo se desarrollaron todos los modelos del API en la aplicación móvil, un ejemplo de estos es la clase Cultivo la cual se compone de sus atributos básicos y constructores pero a su vez posee una función JSON que solicita un dato de tipo Map que posee los mismos campos que tiene el modelo en el API esto tiene como función almacenar los datos del API al objeto de tipo Cultivo creado por el desarrollador, al hacer esto podemos organizar los datos en la aplicación de forma fácil y sencilla.

Figura 40 función transformación JSON

```
factory Cultivo.fromJson(Map<String, dynamic> json) {  
    return Cultivo.completo(  
        int.parse(json['id'].toString()),  
        json['nombre'].toString(),  
        json['nombreCientifico'].toString(),  
        json['descripcion'].toString(),  
        json['ambienteCrecimiento'].toString());  
}
```

Fuente: Autor

Una vez creados todos los modelos que están en el API a la aplicación y creadas las funciones que extraen los datos de la misma se debe pasar directamente a como se usan estas funciones en la interfaz de usuario, esto se puede hacer por medio de otra función la cual permite tomar este resultado y lo convierte directamente en objeto, no obstante la función principal de la función es conectarse a la API por medio de una petición GET, esta petición GET debe tener la ruta específica del modelo a consultar en este caso será el cultivo, cuando se completa la petición se debe confirmar que esta petición fue exitosa por esto mismo, se genera un *if* que confirme el estado de la petición para peticiones exitosas el código de estado es el 200, una vez confirmado el código de estado se pasa a decodificar la información, esta información llega en formato JSON y pasada por un filtro que permita los caracteres especiales como tildes, apostrofes, entre otros. Esto con el fin de que la información no se altere al momento de llegar a la aplicación, en el momento que se complete esto se decodifica el JSON proveído por el API y se procesa en la función programada anteriormente que convierte esta información a un objeto, este

objeto será consultado por alguna función de la interfaz la cual tomara los datos necesarios para mostrar en la misma.

Figura 41 Función de recolección de información API

```
Future<List<Cultivo>> fetchDataFromJSON() async {
    http.Response cultivo = await cliente.get(ruta);

    if (cultivo.statusCode == 200) {
        List jsonData = json.decode(utf8.decode(cultivo.bodyBytes));

        for (Map<String, dynamic> cultivo in jsonData) {
            print(cultivo);
            cultivos.add(Cultivo.fromJSON(cultivo));
        }

        print(cultivos);
    } else {
        Exception("Ha ocurrido un error al consultar el API");
    }
    return cultivos;
}
```

Fuente: Autor

Para usar esta función de manera apropiada se debe implementar dentro de la aplicación en si por medio de un *FutureBuilder* esta clase nos permite esperar información traída desde algún sitio web como lo es la API, para después procesar la información directamente a la funciones que se hayan desarrollado para que consuman esta información, en este caso la función *_textoExpandible* solicita varios argumentos como nombre y descripción los cuales extrae de la variable *snapshot* que a su vez es de tipo *Cultivo* y usa las propiedades de la clase *Cultivo* para dar con estos datos.

Figura 42 Widget FutureBuilder para traer datos

```
child: FutureBuilder(
    future: fetchDataFromJSON(),
    builder: (context, AsyncSnapshot<List<Cultivo>> snapshot) {
        if (snapshot.hasData) {
            return Column(
                children: [
                    _titulo(),
                    _textoExpandible(
                        snapshot.data[0].nombre,
                        snapshot.data[0].descripcion,
                        context,
                        DescripcionCultivo.descripcionCultivo.toString()),
                ],
            ); // column
        } else {
            return Container(
                child: Text("No hay información para mostrar"),
            ); // Container
        }
    )), // FutureBuilder // SingleChildScrollView
```

Fuente: Autor

En el caso de enviar información a la API hicimos uso de una función parecida a la usada anteriormente para recibir información, esta vez se le pasan argumentos a la función para construir un objeto de tipo JSON y pasarlo a el API, esto se hace por medio de una petición post la cual solicita la ruta y las cabeceras del contenido, una vez se complete la petición el API mandara el código de respuesta 201 para confirmar que efectivamente se ha creado el objeto.

Figura 43 Función POST para mandar datos al API

```
Future<Sugerencia> crearSugerenciaPost(
    String titulo, String cuerpo, int puntuacion, String usuario) async {
final http.Response response = await http.post(
    ruta,
    headers: <String, String>{
        'Content-Type': 'application/json; charset=UTF-8',
    },
    body: jsonEncode(<String, dynamic>{
        'titulo': titulo,
        'cuerpo': cuerpo,
        'puntuacion': puntuacion,
        'usuarioForo': usuario,
    }),
);
if (response.statusCode == 201) {
    return Sugerencia.fromJson(jsonDecode(response.body));
} else {
    throw Exception("Ha ocurrido un error");
}
}
```

Fuente: Autor

Para la generación dinámica de objetos dentro de la aplicación se hace por medio de funciones que siguen una estructura similar, esta estructura tiene como base un *ListView.Builder* el cual se encarga de recorrer una lista de datos y ponerlos en su respectivo Widget y este después ser mostrado en la interfaz.

Figura 44 ListView Builder para crear objetos de manera dinámica

```
ListView _lista(List<Cultivo> cultivos) {
    return ListView.builder(
        itemCount: cultivos.length,
        itemBuilder: (BuildContext ctxt, int index) {
            return new Text(cultivos[index].nombre);
        });
}
```

Fuente: Autor

Para comprobar que la aplicación tenga conexión a internet se instaló el paquete connectivity, dentro del cual se pueden usar funciones para hacer peticiones a sitios web como Google y comprobar que el dispositivo tenga conexión a internet, una vez llamada esta función se mostrara la información correspondiente en cada una de las pantallas.

Figura 45 Función para comprobar internet

```
Future<bool> checkConnection() async {
    bool previousConnection = hasConnection;

    try {
        final result = await InternetAddress.lookup('google.com');
        if (result.isNotEmpty && result[0].rawAddress.isNotEmpty) {
            hasConnection = true;
        } else {
            hasConnection = false;
        }
    } on SocketException catch (_) {
        hasConnection = false;
    }

    if (previousConnection != hasConnection) {
        connectionChangeController.add(hasConnection);
    }

    return hasConnection;
}
```

Fuente: Autor

El llamado de la cámara se hace por medio del paquete *camera* e *ImagePicker*, se llama a la función *pickImage* para escoger de donde se va a escoger la imagen, en esta misma función se inserta la clase *ImageSource* y esta a su vez llama a la cámara para hacer uso de la misma, cuando se hace la petición de la cámara por primera, la aplicaciones genera un dialogo de permiso del uso de la cámara al usuario, una vez se confirme el permiso el dialogo desaparecerá y no volverá a solicitar el permiso nuevamente.

Figura 46 Función para abrir la cámara

```
void _openCamera(BuildContext context) async {
    var picture = await ImagePicker.pickImage(source: ImageSource.camera);

    this.setState(() {
        imageFile = picture;
    });

    Navigator.of(context).pop();
}
```

Fuente: Autor

Una vez tomada la foto se hace una petición http de tipo post, para enviarla y al igual que en las demás funciones se espera la respuesta de la API, una vez se confirme la respuesta del API la misma enviara el resultado a la aplicación, para darle el diagnostico al usuario

Figura 47 función POST para crear sugerencias

```
Future<Sugerencia> crearSugerenciaPost(
    String titulo, String cuerpo, int puntuacion, String usuario) async {
final http.Response response = await http.post(
    ruta,
    headers: <String, String>{
        'Content-Type': 'application/json; charset=UTF-8',
    },
    body: jsonEncode(<String, dynamic>{
        'titulo': titulo,
        'cuerpo': cuerpo,
        'puntuacion': puntuacion,
        'usuarioForo': usuario,
    }),
);
if (response.statusCode == 201) {
    return Sugerencia.fromJson(jsonDecode(response.body));
} else {
    throw Exception("Ha ocurrido un error");
}
}
```

Fuente: Autor

8.4 INTEGRACIÓN DEL SISTEMA

Para la integración e implementación del sistema se utilizo el software *MobaXterm* que permite correr comandos de Linux en Windows esto con el fin de configurar el servidor donde se va a alojar el administrador web y sus demás dependencias, una vez instalado MobaXterm se dispuso a hacer uso de los comandos de este y acompañado con el AWS (Amazon Web Services) poner el sistema en producción.

Como primer paso se registró un dominio y un subdominio a nombre de “*iacultivar*” para poder acceder al sitio web y al API, una vez registrado se configuró la raíz del dominio directamente desde la consola de AWS en esta consola se le asigna al servidor el nombre del dominio y el subdominio a usar para el sitio web, para evitar algún acceso intrusivo se hicieron configuraciones directamente con la misma consola AWS y bloqueando cualquier acceso a personal no autorizado.

Figura 48 bloqueo al acceso publico

Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Account settings for Block Public Access are currently turned on

Account settings for Block Public Access that are enabled apply even if they are disabled for this bucket.

Block all public access

Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

- Block public access to buckets and objects granted through new access control lists (ACLs)**
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
- Block public access to buckets and objects granted through any access control lists (ACLs)**
S3 will ignore all ACLs that grant public access to buckets and objects.
- Block public access to buckets and objects granted through new public bucket or access point policies**
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
- Block public and cross-account access to buckets and objects through any public bucket or access point policies**
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

Fuente: Autor

Una vez hechas estas configuraciones se pueden hacer pruebas directamente para saber si el sitio web está funcionando correctamente.

Figura 49 sitio registrado

The screenshot shows the Django admin interface for a registered site. The top bar includes the URL www.iacultivar.com/admin/. The main area is titled "Django administration" and "Site administration". It features two main sections: "AUTHENTICATION AND AUTHORIZATION" (Groups, Users) and "PRINCIPAL" (Cultivo preguntas, Cultivos, Danos enfermedad cultivos, Enfermedad cultivos, Enfermedad preguntas, Enfermedades). On the right, there's a sidebar with "Recent actions" and "My actions" logs.

Fuente: Autor

Figura 50 Cultivo en sitio registrado

The screenshot shows a "Change cultivo" form in the Django admin site. The URL is www.iacultivar.com/admin/principal/cultivo/1/change/. The form has fields for Nombre (Cacao), NombreCientifico (Theobroma cacao), Descripcion (árbol del cacao o cacaotero, nativo de region), and AmbienteCrecimiento (Humedo, Templado). At the bottom, there are buttons for Delete, Save and add another, Save and continue editing, and a large blue "SAVE" button.

Fuente: Autor

9. FASE DE PRUEBAS Y COMPROBACIÓN DEL SISTEMA

Un aspecto crucial en el control de calidad del desarrollo de la aplicación son las pruebas, en las cuales se hace una verificación dinámica del comportamiento del sistema, basada en la observación de un conjunto seleccionado de ejecuciones controladas o casos de prueba. Para hacer pruebas funcionales se requiere una planificación que consiste en definir los aspectos a revisar y la forma de verificar su correcto funcionamiento, punto en el cual adquieran sentido los casos de prueba. Es por esto que implemento un método para generar casos de prueba funcional a partir de casos de uso del sistema.

9.1 DISEÑO DE ESCENARIOS DE PRUEBAS

Los escenarios de prueba se diseñaron para consultar las secciones de la fase de planificación y fase de diseño, y asegurarse de probar todos los escenarios importantes. para derivar los escenarios de prueba se debe contar con insumos como: la especificación de casos de uso (diagramas y plantillas de descripción de cada caso de uso), una plantilla para diligenciar cada caso de prueba y la versión de la aplicación. Como salida se obtendrá un conjunto de plantillas de casos de prueba debidamente diligenciados.

Tabla 19 Ejemplo prueba

ID caso de prueba	04
Nombre caso de prueba	Foro
Descripción	Se probará la respuesta de la aplicación al cuando el usuario desee ingresar al listado de discusiones
Precondiciones	Tener instalada la aplicación es su dispositivo Android, estar en la pantalla principal de la aplicación
Relación de Caso de Uso	El caso de prueba está relacionado con el caso de uso “Ingresar listado de discusiones”
Pasos y condiciones de ejecución	Los pasos a seguir para llevar acabo este caso de prueba el usuario debe ingresar al menú principal de la aplicación y seleccionar el botón de discusiones el cual lo redirige a un listado de foros que esta dividido por cultivos y enfermedades asociadas al mismo
Resultado esperado	Con la ejecución de este caso de prueba se quiere ver como se comporta la aplicación cuando el usuario haga la petición de ingresar al listado de

	discusiones dando clic en el botón el usuario deberá ser dirigido a una pantalla con todos los foros de cultivos registrados en el admin y traídos a la aplicación mediante el API
Estado de caso de prueba	Ejecutado
Resultado obtenido	Se evidencio el correcto funcionamiento de la aplicación cuando se hace la petición de mostrar el listado de discusiones ya que esta muestra los foros categorizados por cultivos y enfermedades del mismo
Errores asociados	Ninguno
Responsable de diseño caso de prueba	Equipo de trabajo
Responsable de ejecución caso de prueba	Usuario

Fuente: Autor

La siguiente tabla muestra un resumen de todas las pruebas que debió realizar el equipo de trabajo.

Tabla 20 Resumen de las pruebas a realizar

Nº. De prueba	Descripción de prueba	Entrada para probar	Resultado esperado	Resultado
1-1	Navegación por cada una de las interfaces	Pantallas	Entrada a cada pantalla y salida de la misma, pasar entre pantallas	
1-2	Enviar información de formulario de registro a foro	Flujo de información entre el API, web admin, Base de datos	Recibir la información ingresada por el usuario, almacenarla en la base de datos y verificarla cada vez que intente ingresar	
1-3	Realizar preguntas y responder preguntas de un foro general y de una enfermedad específica	Flujo de información entre API, web admin y base de datos	Enviar las preguntas y respuestas hechas por los usuarios en el foro de cultivo y de una enfermedad de cultivo específico	
1-4	Enviar una foto para analizar el porcentaje de presencia de enfermedad	Conexión API, IA, App	Tomar la foto y enviarla para recibir el porcentaje de coincidencia de la imagen con la enfermedad y recibir el tratamiento de acuerdo con la etapa en la que se encuentre la enfermedad	
1-5	Tiempos respuesta peticiones realizadas desde la aplicación		Recibir la información de cada una de las peticiones y acciones en un tiempo prudente y asegurar que la aplicación no sea lenta	
1-6	Enviar información formulario de registro a foro		Información de registro recibida y almacenada en la base de datos de manera correcta	
1-7	Enviar información formulario registro a foro		Información de registro recibida y almacenada en la base de datos de manera correcta	

Fuente: Autor

9.2 EJECUCIÓN DE LOS ESCENARIOS DE PRUEBAS

Tabla 21 Resultados pruebas

Nº. De prueba	Descripción de prueba	Entrada para probar	Resultado esperado	Resultado
1-1	Navegación por cada una de las interfaces	Pantallas	Entrada a cada pantalla y salida de la misma, pasar entre pantallas	Se logra evidenciar que la aplicación permite entrar y salir de forma fácil de todas las interfaces que esta contiene.
1-2	Enviar la información de formulario de registro a foro	Flujo de información entre el API, web admin, Base de datos	Recibir la información ingresada por el usuario, almacenarla en la base de datos y verificarla cada vez que intente ingresar	La aplicación muestra que todas las partes del sistema se encuentran conectadas de forma correcta ya que nos permiten hacer peticiones y recibir respuestas de la misma.
1-3	Realizar preguntas y responder preguntas de un foro general y de una enfermedad específica	Flujo de información entre API, web admin y base de datos	Enviar las preguntas y respuestas hechas por los usuarios en el foro de cultivo y de una enfermedad de cultivo específico	Se observa que la aplicación carga toda la información registrada por los demás usuarios y permite subir nueva, esto nos muestra que la conexión API base de datos es la correcta
1-4	Enviar una foto para analizar el porcentaje de presencia de enfermedad	Conexión API, IA, App	Tomar la foto y enviarla para recibir el porcentaje de coincidencia de la imagen con la enfermedad y recibir el tratamiento de acuerdo	Se puede ver que la aplicación puede acceder a la cámara del dispositivo móvil, y captar la foto, enviarla al modulo de

		con la etapa en la que se encuentre la enfermedad	inteligencia artificial para el análisis, resultado que se muestra como porcentaje.
1-5	Tiempos de respuesta a peticiones realizadas desde la aplicación	Recibir la información de cada una de las peticiones y acciones en un tiempo prudente y asegurar que la aplicación no sea lenta	Se pudo evidenciar que la aplicación tiene un tiempo prudente en cada una de las peticiones hechas.

Fuente: Autor

Para generar los casos de prueba se utilizó una técnica que se basa en los casos de usos del sistemas (González Palacio, 2009), los módulos del sistema fueron sometidos a dichas pruebas las cuales se hicieron por pasos cada caso de prueba, los cuales tienen unas precondiciones e inicialmente es requerido que el usuario tenga un dispositivo móvil con sistema operativo API 21 lo que es equivalente a versiones con *Nougat*, tener instalada la aplicación en el mismo, se probó en 5 dispositivos móviles con características diferentes, tamaños diferentes, las personas que la probaron fueron personas ajenas al proyecto, personas no expertas en el área de sistema, pero a las cuales les interesa el cultivo de cacao, cada caso de prueba lleva descrito el paso a paso de su ejecución, los resultados del mismo. La aplicación respondió muy bien a cada una de las peticiones que se hicieron tanto en pruebas de unidad como en pruebas conjuntas, pudiendo evidenciar que el sistema se encuentra integrado de manera correcta.

9.3 ANÁLISIS DE LOS RESULTADOS DE LOS ESCENARIOS DE PRUEBAS

Figura 51 Análisis de pruebas

caso de prueba	Resultado	Fortalezas	Dificultades
Ver listado de enfermedades	Al llevar acabo la ejecución se vio que la aplicación realiza el caso de prueba de manera correcta ya que el usuario es dirigido a una pantalla donde se muestra el listado de las enfermedades registradas	Con la ejecución de este caso de prueba se logro evidenciar que la aplicación conecta de manera rápida y fácil al usuario y le permite visualizar todos los cultivos registrados	el usuario no sabía que el listado era cliqueable, y por esta razón no podía ingresar a la pantalla de detalle de enfermedades
Seleccionar enfermedades	Al llevar acabo la ejecución se vio que la aplicación realiza el caso de prueba de manera correcta ya que el usuario puede seleccionar de manera correcta la enfermedad	Muestra información de forma rápida	Ninguna
Información de enfermedad	Al llevar acabo la ejecución se vio que la aplicación realiza el caso de prueba de manera correcta ya que el usuario es dirigido a una pantalla donde se carga de	Muestra de manera rápida la información y la interfaz que lo contiene es entendible para el usuario.	Las imágenes tardaron un poco en mostrarse en pantalla.

	manera correcta la información de la enfermedad y las imágenes que describen la enfermedad		
Foro	Se evidencio el correcto funcionamiento de la aplicación cuando se hace la petición de mostrar el listado de discusiones ya que esta muestra los foros categorizados por cultivos y enfermedades del mismo.	La aplicación móvil muestra de manera rápida la pantalla del foro y el usuario se pudo identificar que se trataba de un foro de discusiones	ninguna
Registro	Al realizar esta prueba se vio que la aplicación trae de manera correcta los datos como el listado de los cultivos registrados y permite seleccionar uno de ellos para ser explorado con mayor y profundidad en un foro y participar del mismo	El formulario es entendible para el usuario y los datos son almacenados de forma correcta	Ninguna

Realizar pregunta	Al realizar esta prueba se vio que la aplicación trae de manera correcta los datos como las preguntas y respuestas de manera rápida, se pudo realizar la pregunta	Al ser un módulo filtrado por cultivos y enfermedades permite que el usuario pueda participar más activamente en los foros	Cuando se hacen un número de peticiones alto tiende a no mostrar de forma rápida las preguntas o respuesta hechas por los usuarios
Hacer sugerencia	Al realizar esta prueba se vio que la aplicación trae de manera correcta los datos como las pregunta y la almacena en la base de datos	Al tener un botón flotante para el usuario fue rápido poder hacer una nueva sugerencia	Ninguna
Diagnóstico	Al realizar la prueba se evidencio que la aplicación realiza de manera correcta el direccionamiento desde el menú principal hasta la pantalla que muestra la cámara del dispositivo y solicita permiso para utilizar la cámara el cual al ser otorgado por el usuario se puede hacer la captura de la foto	El botón de detección en vivo es muy ilustrativo y el usuario sabe que ahí deberá tomar una foto que será analizada	Ninguna

Imagen Diagnóstico	La aplicación conecta de manera correcta con el API y la IA para que la imagen sea subida y guardada dentro de una carpeta del proyecto llamada media que es donde se va almacenar todas las imágenes que los usuarios envían para análisis	La imagen se sube puede subir apenas es tomada	Cuando el dispositivo móvil no se encuentra con conexión a internet el usuario no podrá obtener el diagnóstico.
Detección de enfermedad	Se evidencia la correcta comunicación en el sistema y con esto se puede visualizar el diagnóstico de la enfermedad a modo de porcentaje	El diagnóstico es entendible para el usuario ya que sabe que cuando el número es un numero cercano a 100 es porque tiene una mazorca muy enferma y si es cercano a 0 tiene una mazorca sana	Al solo estar trabajando con la monilia algunas mazorcas que tengan un síntoma similar se puede llegar a confundir.

Fuente: Autor

CONCLUSIONES

A manera de conclusión se puede observar que el comportamiento del reconocimiento de imágenes aunque exitoso deja algunas cosas que desear ya que con una porcentaje de predicción del 88% en algunas pruebas realizadas se evidencio un poco de confusión entre la diferencia de cacao sano y monilia, aunque se buscó el modelo con una mayor porcentaje de predicción no se pudo encontrar una que satisfaga las necesidades del proyecto, en casos puntuales donde se presentan varias mazorcas de cacao en la misma imagen el modelo tiende a clasificarla como monilia cuando este es cacao, también se observa que en cambios de color como rojos oscuros se tiende a presentar confusiones para las mazorcas saludables.

La aplicación móvil cumple con su cometido al proveer información básica para los usuarios a su vez las recomendaciones de tratamiento fueron bien recibidas por el público en general, aunque estos a su vez manifiestan que se debe disponer de mas información ya que hay detalles en los que no se ahonda lo suficiente y lo cual es importante según ellos.

El sitio web y el API demuestran un rendimiento sobresaliente excepto cuando se hacen demasiadas peticiones post esto se debe a que al usar un servicio de hosting gratuito como el AWS se usa un servidor que no cumple con las capacidades apropiadas para la solución. Aunque también cabe destacar que en peticiones de tipo get no se presentó ningún inconveniente.

La integración de todos los servicio fue un éxito por el hecho de que al conectarlos no surgió ningún problema de tablas o modelos erróneos, a su vez el código al haber sido desarrollado con convenciones de cada uno de los lenguajes de programación usados puede ser mantenido con facilidad y usado para proyectos futuros.

RECOMENDACIONES O TRABAJOS FUTUROS

Como trabajos futuros se recomienda el uso de herramientas más especializadas para el reconocimiento de imágenes esto con el fin de mejorar la precisión de los modelos ya que se evidencio en diferentes tablas una varianza en la precisión de los modelos, se puede expandir los modelos del web admin para recopilar más información y se refuerce el aprendizaje dentro de la aplicación, como por ejemplo recomendaciones personalizadas por expertos en la agricultura, asociarse con entidades agricultoras puede reforzar la adquisición de datos ya que este es un punto crítico dentro del desarrollo de la arquitectura y este puede ralentizar el desarrollo de próximos proyectos.

El uso de información en guías agricultoras es de vital importancia para el desarrollo del reconocimiento de imágenes ya que estas contienen información de cómo se puede presentar las enfermedades en etapas previas, donde se es más difícil la detección de enfermedades.

Expandir el banco de enfermedades puede ayudar al reconocimiento de imágenes ya que al tener una variedad de enfermedades se puede ser más preciso con el diagnóstico y a su vez contribuir más en el ámbito agrícola.

Enfocarse en las teorías de color es importante para evitar confusiones entre cacao sano y cultivo ya que esto se presenta en algunos casos.

REFERENCIAS BILIOGRÁFICAS

- Schwaber, K., & Sutherland, J. (2017). *La guía de Scrum*.
- Google. (2020). *Dart*. <https://dart.dev/>
- Rocha, A., Hauagge, D. C., Wainer, J., & Goldenstein, S. (2010). Automatic fruit and vegetable classification from images. *Computers and Electronics in Agriculture*, 70(1), 96–104. <https://doi.org/10.1016/j.compag.2009.09.002>
- Google. (2020). *About the Android Open Source Project*.
<https://source.android.com/>
- Ricardo, R. (2018). *Biodiversidad de ecotipos de Moniliophthora roreri en cacao (*Theobroma cacao*) clon ccn-51 y la actividad antagonista de PGPR*.
- Jesus, S. C. (2015). *Diseño de una aplicación Android para la enseñanza de tropismos y adaptaciones vegetales*.
- Plantix. (2020). *Plantix*. <https://plantix.net/es/>
- Perea, J. A., Ramirez, O. L., & Villamizar, A. R. (2011). Caracterización fisicoquímica de materiales regionales de cacao Colombiano. *Biotechnology in the Agricultural and Agroindustrial Sector*.
<https://www.openaire.eu/search/publication?articleId=revistacauca::78c0e149f4731c2fd2d2f9df5ddabc12>
- Marzano, R. J. (2014). *Art and Science of Teaching / Investigation—The New Research Report*. <http://www.ascd.org/publications/educational-leadership/feb14/vol71/num05/Investigation—The-New-Research-Report.aspx>
- Wang, K., Zhang, S., Wang, Z., Liu, Z., & Yang, F. (2013). Mobile smart device-based vegetable disease and insect pest recognition method. *Intelligent Automation & Soft Computing*, 19(3), 263–273.
<https://doi.org/10.1080/10798587.2013.823783>
- Oracle. (2020). *What is a database*
. <https://www.oracle.com/database/what-is-database.html>

- Rouse, M., & Denman, J. (2017). *Desarrollo de aplicaciones móviles*.
<https://searchdatacenter.techtarget.com/es/definicion/Desarrollo-de-aplicaciones-moviles>
- contributors, M. D. N. (2020). *Detalles del modelo de objetos*.
https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Details_of_the_Object_Model
- Knowlton, & Jim. (2008). *Python* (1st ed.).
- Dubey, S. R., & Jalal, A. S. (2015). Fruit and vegetable recognition by fusing colour and texture features of the image using machine learning. *International Journal of Applied Pattern Recognition*, 2(2), 160–181.
<https://doi.org/10.1504/IJAPR.2015.069538>
- Javier, C., Sergio, C., & Jairo, C. (2014). *Estado de la moniliasis del cacao causada por Moniliophthora roreri en Colombia* (Vol. 63).
<https://www.redalyc.org/articulo.oa?id=169932435011>
- Group, T. P. G. D. (1996). *What is PostgreSQL?* <https://www.postgresql.org/about/>
- Tirado-Gallego, P. A., Lopera-Álvarez, A., & Ríos-Osorio, L. A. (2016). Estrategias de control de Moniliophthora roreri y Moniliophthora perniciosa en Theobroma cacao L.: revisión sistemática. *Ciencia y Tecnología Agropecuaria*, 17(3), 417–430. https://doi.org/10.21930/rcta.vol17_num3_art:517
- Deloitte. (2019). *Consumo móvil en Colombia*.
<https://www2.deloitte.com/co/es/pages/technology-media-and-telecommunications/articles/consumo-movil-en-colombia-2019.html>
- Andrés, T. V. J., & Mauricio, A. G. A. (n.d.). “DESARROLLO DE APLICACIÓN MÓVIL PARA LA POSTPRODUCCIÓN Y COMERCIALIZACIÓN DE LA MORA EN LA REGIÓN DE SANTUARIO, ASOCIACIÓN.
- MuleSoft, L. L. C. (2020). *What is a REST API*.
<https://www.mulesoft.com/resources/api/what-is-rest-api-design>
- PI@ntNet. (2020). *PI@ntNet Identify*. <https://identify.plantnet.org/>
- Fedecacao. (2015). *Guía técnica para el cultivo de cacao*.
- Xin, J., Mingyong, L., Kaixuan, Z., Jiangtao, J., Hao, M., & Zhaomei, Q. (2019). Development of vegetable intelligent farming device based on mobile APP.

Cluster Computing, 22(S4), 8847–8857. <https://doi.org/10.1007/s10586-018-1979-4>

Hernández, M. S., Saldaña, T. M., Castro, E. S., Avalos, V. C., Chulim, N. E., García, C. F. O., & Layalle, A. M. (2001). Factores que afectan la producción de cacao (*Theobroma cacao L.*) en el ejido Francisco I. Madero del plan chontalpa, Tabasco, Mexico. *Universidad y Ciencia*, 17(34), 93–100. <http://dialnet.unirioja.es/servlet/oaiart?codigo=2339735>

Nestor, C., Aquiles, D., Miguel, S., & Alia, R. (2017). ANÁLISIS ESPACIAL DE LA INCIDENCIA DE ENFERMEDADES EN DIFERENTES GENOTIPOS DE CACAO (*Theobroma cacao L.*) EN EL YOPAL (CASANARE), COLOMBIA
. *Acta Biologica Colombiana*, 22. <https://www.redalyc.org/articulo.oa?id=319051099010>

Mummah, S., Robinson, T. N., Mathur, M., Farzinkhou, S., Sutton, S., & Gardner, C. D. (2017). Effect of a mobile app intervention on vegetable consumption in overweight adults: a randomized controlled trial. *The International Journal of Behavioral Nutrition and Physical Activity*, 14(1), 125. <https://doi.org/10.1186/s12966-017-0563-2>

Khadabadi, G. C., Kumar, A., & Rajpurohit, V. S. (2015). *Identification and classification of diseases in carrot vegetable using Discrete Wavelet Transform*. 59–64. <https://doi.org/10.1109/ERECT.2015.7498988>

Foundation, D. S. (2020). *Why Django?* <https://www.djangoproject.com/start/overview/>

Gracia, J. F. H. (2018). Tipos de Investigación. *Boletín Científico de La Escuela Superior Atotonilco de Tula*, 5(9). <https://doi.org/10.29057/esat.v5i9.2885>

Google. (2020). *Teachable Machine*.

Acosta, K. R. (2017). Interfaz y experiencia de usuario: parámetros importantes para un diseño efectivo. *Tecnología En Marcha*. <https://www.scielo.sa.cr/pdf/tem/v30s1/0379-3982-tem-30-s1-49.pdf>

Riehle, & Dirk. (2000). *Framework Design: A Role Modeling Approach*
. www.riehle.org/computer-science/research/dissertation/diss-a4.pdf

Hope, C. (2019). *Programming language*. <https://www.computerhope.com/jargon/p/programming-language.htm>

Agricultura, M. (2017). *Principales Cultivos por Área Sembrada en 2017*.

Mummah, S. A., King, A. C., Gardner, C. D., & Sutton, S. (2016). Iterative development of Vegethon: a theory-based mobile app intervention to increase vegetable consumption. *The International Journal of Behavioral Nutrition and Physical Activity*, 13(1), 90. <https://doi.org/10.1186/s12966-016-0400-z>

Sebastián, V. P. (n.d.). *UNIVERSIDAD DE CHILE FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN.*

Microsoft. (2020). *What is Custom Vision?* <https://docs.microsoft.com/en-us/azure/cognitive-services/custom-vision-service/home>

ANEXOS

Anexo A Encuestas y Entrevistas a Personal con cercano con el manejo del cultivo de cacao

A partir de preguntas planteadas en la sección anterior se distribuyó la encuesta a agricultores de la zona de san Vicente de chucuri en el departamento de Santander, los resultados fueron los siguientes:

Se recogieron un total de 24 respuestas en un plazo de 1 semana:

Nivel socio económico

24 respuestas

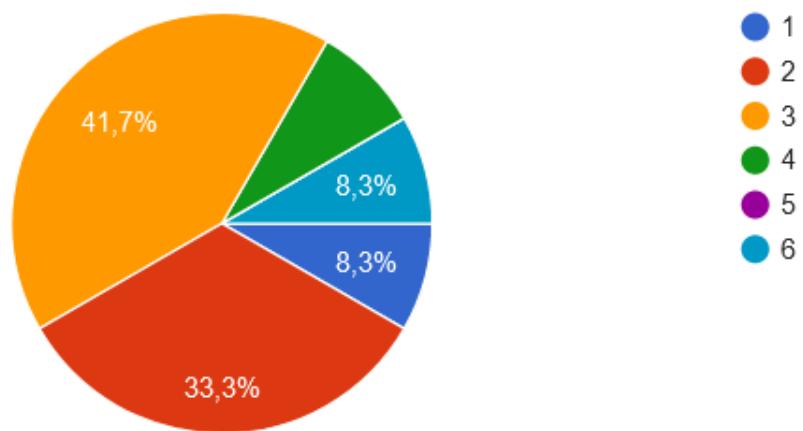


Figura 52 Respuestas nivel socio económico

Género

24 respuestas

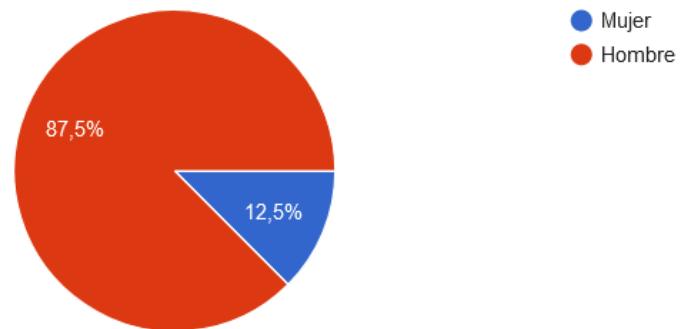


Figura 53 respuestas género

Edad

24 respuestas

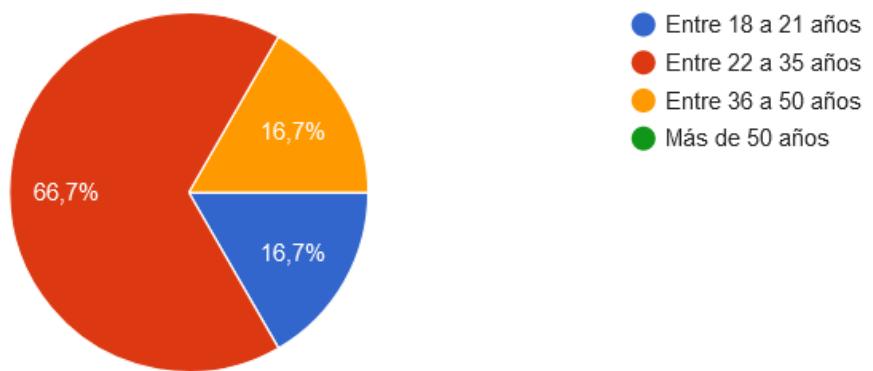


Figura 54 respuestas edad

- 1. En una escala del 1 al 5 siendo 5 la calificación más alta y 1 la más baja ¿Que tanto considera que cuida su cultivo?**

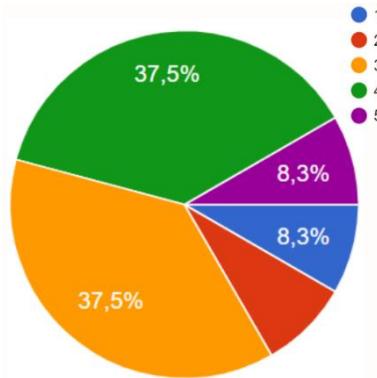


Figura 55 diagrama circular pregunta 1

Análisis:

Esta pregunta nos muestra que en la escala del 1 al 5 el

- 37,5% de los encuestados responde a 4 en la escala planteada son 5 hombres entre 22 y 35 años y 1 entre 18 y 21 años
- 37,5% a una calificación de 3, de los que son 6 hombres en un rango de edad de 22 a 35 años, de estado civil solteros y viven solos

con esto podemos concluir que la mayoría de los cacaoteros se preocupan por cuidar su cultivo y mantenerlo en óptimas condiciones para obtener una mayor producción

2. ¿Cuándo ve la presencia de alguna enfermedad en su cultivo busca ayuda en?

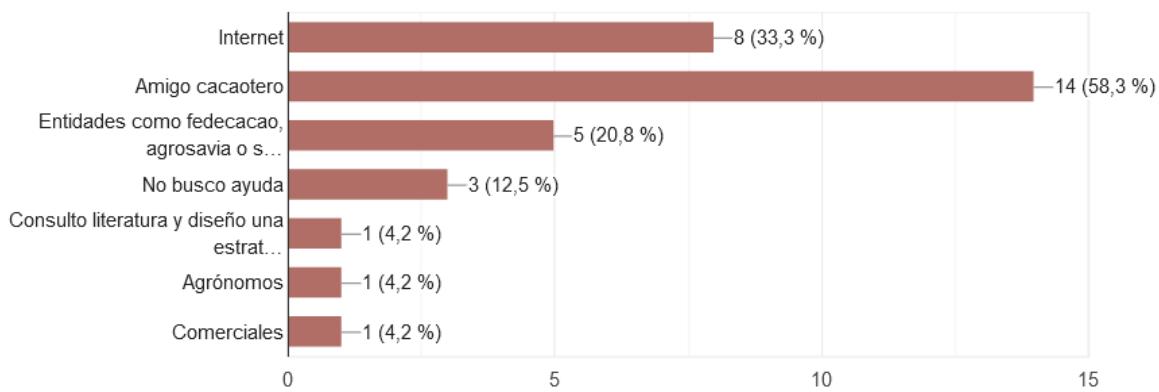


Figura 56 diagrama de barras pregunta 2

Análisis:

En esta pregunta se evidencia que

- 58,3% de los encuestados prefieren buscar ayuda en un amigo cacaotero concluyendo que lo hacen porque es información basada en experiencias y esta de primera mano para ellos
- un 33,3% prefiere busca información en Internet esto deja claro que intentan buscar soluciones en herramientas tecnológicas como aplicaciones web, documentos online, blogs etc.
- 20,8% de las personas encuestadas acuden a información de soluciones que les pueden brindar asociaciones como fedecacao, agrosavia, entre otras. Estas personas acuden a estas entidades buscando información en un poco más especializada.
- Los demás encuestados buscan ayuda en personas especializadas en la producción y enfermedades como lo son los agrónomos, pero este porcentaje es mucho menor ya que para que un cultivo sea tratado por un agrónomo requiere un costo mayor

3. ¿Su cultivo ha presentado enfermedades por hongos como?

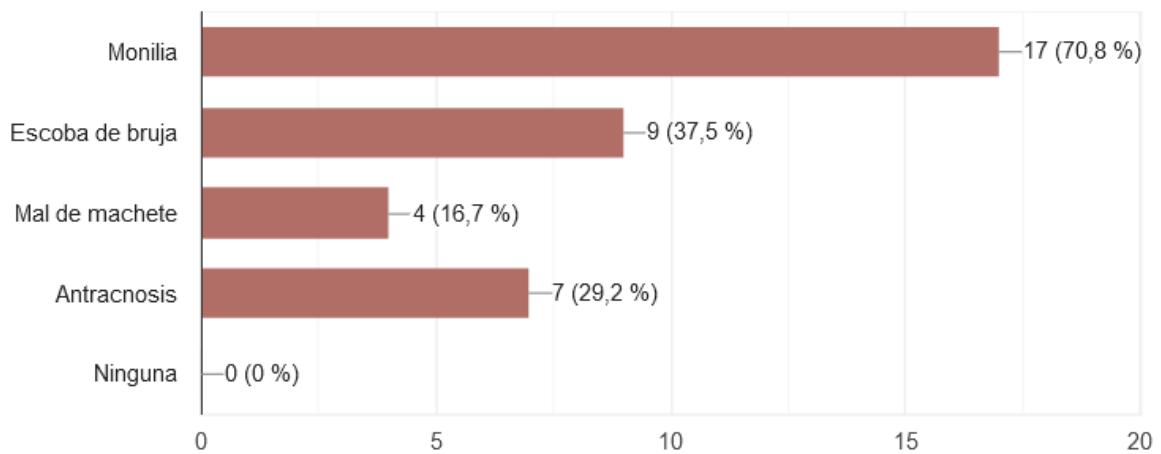


Figura 57 diagrama de barras pregunta 3

Análisis:

Las enfermedades por hongos son una de las más frecuentes en el cultivo de cacao, es por esto que se decidió preguntar cuál de estas enfermedades es la que tiene mayor presencia en el cultivo de acuerdo con los resultados

- un 70,8% dice que la monilia es la que tiene una mayor presencia en el cultivo, con lo que se puede concluir que al ser la mas frecuente es la que necesita un mayor control.
- 37,5% dice que su cultivo presenta escoba de bruja, 29.2% antracnosis, 16.7% mal de machete
- Ninguno de los encuestados señalo que no ha presentado enfermedades por hongos con lo que podemos decir que las enfermedades por hongos están presentes en la gran mayoría de cultivos

4. ¿Cuáles de los siguientes métodos emplea para controlar la proliferación de enfermedades como la monilia en su cultivo?

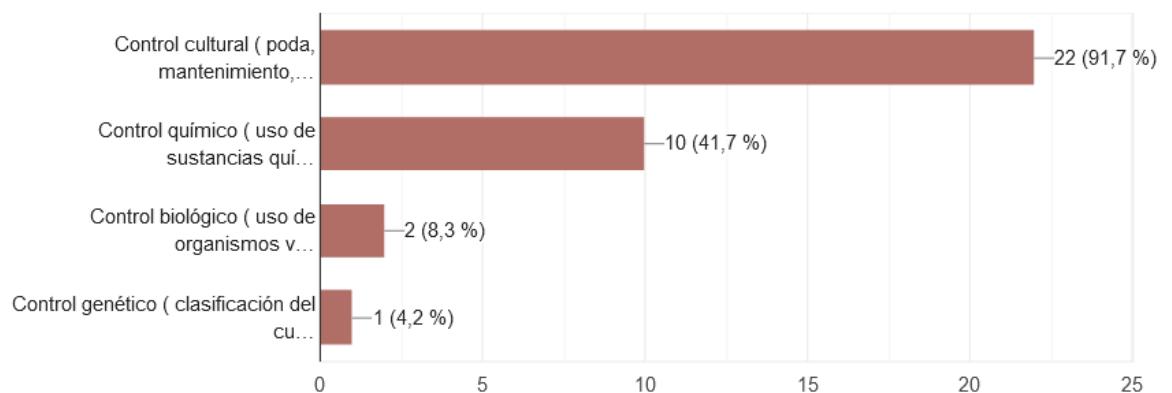


Figura 58 diagrama de barras pregunta 4

Análisis:

Existen diferentes métodos para controlar y evitar la proliferación de enfermedades como la Monilia, algunos de los mas usados según las personas encuestadas es el

- El Control cultural (**91,7%**) que consiste en una adecuada poda, mantenimiento, fertilización, remoción de frutos infectados, estos son tratos que el agricultor puede ejecutar con conocimientos propios y que no conllevan muchos gastos
- El control químico (**41,7%**) consiste en el uso de productos químicos que combaten la enfermedad y evitan que surja nuevamente, este es de un uso considerable, pero para aplicarlo se necesita saber exactamente qué beneficios traen y qué consecuencias porque en el mercado existen productos que pueden ser dañinos tanto para la planta como para el suelo
- El control biológico (**8,3%**) es el uso de organismo vivos que se utilizan como trampa para evitar la proliferación de organismos dañinos para la planta, este tipo de control no es llevado a cabo con frecuencia por los agricultores ya

que este requiere de personas especializadas que sepan cuales de estos organismos y las cantidades adecuadas para difuminar la enfermedad.

5. ¿Considera que una aplicación móvil ayudaría al control y detección de enfermedades como la Monilia?

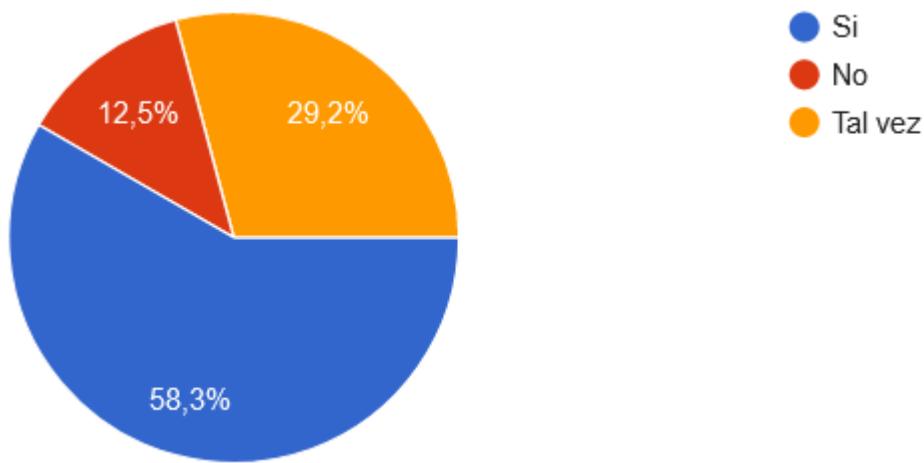


Figura 59 grafico circular pregunta 5

Análisis:

- El 58,3% de los encuestados dicen que si seria de ayuda una aplicación móvil para detectar enfermedades como la monilia de manera temprana gracias a estos resultados podemos decir que las personas ven a las aplicaciones móviles como herramientas que les permiten obtener información de manera oportuna.
- El 29,2% de los encuestados dice no sentirse completamente seguros de que una aplicación móvil pueda ayudar a la detección y control de unas enfermedades esto debido a que tal vez no conocen muy bien el funcionamiento que desarrollara la app
- 12,5% de los encuestados dice no sentir de ayuda una aplicación móvil, estas personas probablemente no sepan como será el funcionamiento de la app ya que esta solamente ayudara a dar posibles soluciones y diagnósticos mas no, por el solo usar la aplicación se halle la solución o finalice el problema.

6. ¿Cuenta con dispositivo móvil con cámara y acceso a Internet?

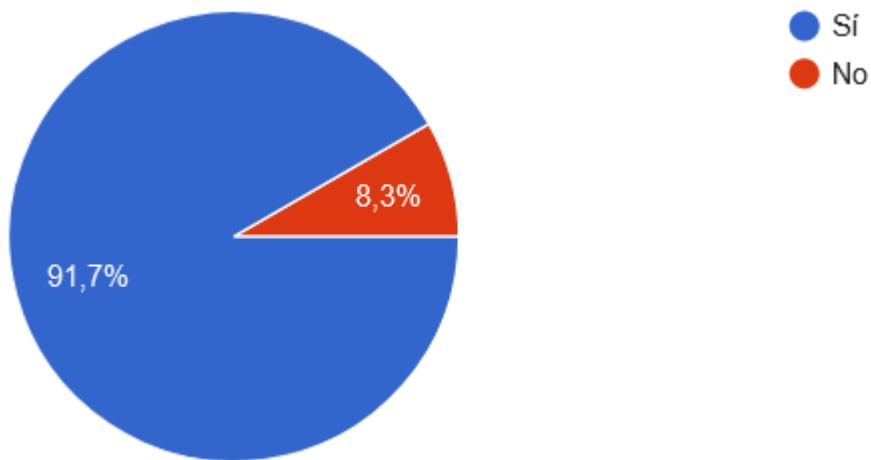


Figura 60 grafico circular pregunta 6

Análisis:

El 91,7% de los encuestados dice contar con un dispositivo móvil que tenga conexión a Internet, actualmente la mayor parte de la población mundial y nacional ha adquirido un teléfono inteligente, con diferentes tipos de cámaras y funcionamiento, para gozar ampliamente de estos dispositivos es necesario contar con conexión a Internet

El resultado de esta pregunta es favorable para la implementación de la aplicación ya que la mayoría dice poder acceder a estos requerimientos.

Sin embargo, se observa que existen personas que tal vez no cuentan con alguno de estos requerimientos lo cual lleva a concluir que no se podrá llegar a todas las personas que tengan un cultivo de cacao, pero por razones ajenas no cuentan con un dispositivo móvil con cámara y acceso a internet.

7. ¿Conoce alguna aplicación móvil que permita detectar enfermedades en su cultivo?

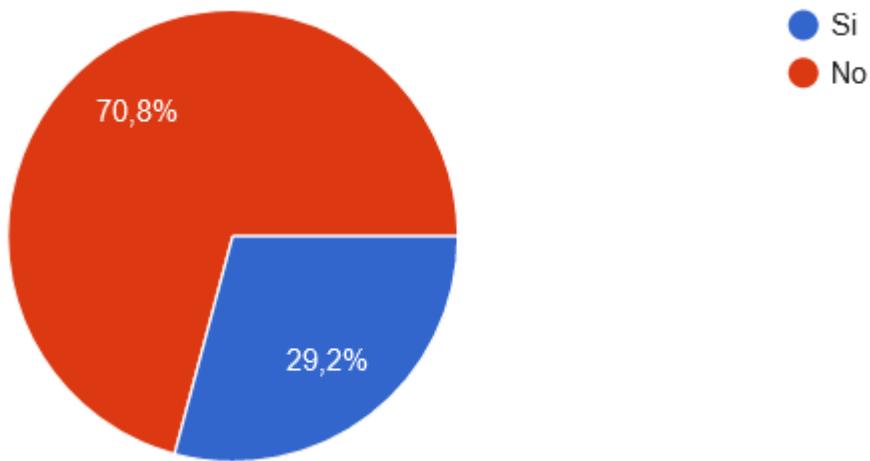


Figura 61 diagrama circular pregunta 7

Análisis:

- El 70,8% de los encuestados dice no tener conocimiento de una aplicación móvil que le permita hacer una detección de enfermedad lo cual indica que aplicaciones de este tipo no son un punto de interés para muchos desarrolladores o que tal vez no están al acceso de muchas personas
- 29,2% indican que si conocen aplicaciones de este tipo con lo cual se puede decir que a pesar de que no es un gran numero ya existen aplicaciones especializadas en encontrar enfermedades en plantas, cultivos, suelos, etc. Pero que para llegar a un mayor público se requiere pensar en el usuario y en sus necesidades.

Entrevistas

Se entrevistaron a 2 personas del sector cacaotero, siendo una el jefe de distribución de cacao en San Vicente de Chucuri del departamento de Santander asociado con Fedecacao y un profesor de agricultura experto en cacao, a ambos entrevistados se les realizaron las preguntas planteadas en la sección anterior y estas fueron las conclusiones que el equipo de trabajo recolectó después de las entrevistas.

Para este ejercicio se tomará al jefe de distribución como **Entrevistado 1** y al profesor de agricultura como **Entrevistado 2**

1. ¿Cuándo el cacao se ve afectado por la monilia cuales con los primeros síntomas que presenta?

Entrevistado 1: Usualmente se presentan varios puntos medio graseos son difíciles de identificar a simple vista, se tiene que revisar muy bien la mazorca del cacao por todos los lados, son puncos medio amarillos casi siempre están en las mazorcas más verdes, aunque en las rojas también se ven son más o menos naranjados, inclusive hay veces donde ya se ven bultos en la mazorca, ahí es cuando se ve que ya está entrando el hongo solamente hay que agarrar un machete y cortarle la parte que está abultada y con eso se evita que el hongo se disperse por todo el cultivo.

Entrevistado 2: Las mazorcas afectadas siempre son las de menor edad como un mes o menos y comienzan a madurarse más rápido que las demás, la mazorca comienza a presentar unos puntos ya sean verdes o naranjados, en las mazorcas que tienen 1 a 3 meses de edad se les ven deformaciones a veces son leves y a veces se ven a simple vista, también se comienzan a abultar después de los 2 o 3 meses también se ve que las deformaciones se comienzan a adquirir unos puntos verdes y hay que buscarlos si o sí.

2. ¿Qué procesos o controles son efectivos para combatir a la monilia?

2.1 ¿Cuándo el cacao se ve afectado por la monilia cuales con los primeros síntomas que presenta?

2.2 ¿Qué procesos o controles son efectivos para combatir a la monilia?

2.3 ¿Qué información deberían conocer los agricultores de primera mano cuando se enfrentan a enfermedades como la monilia?

2.4 ¿Qué opina de los controles químicos utilizados en el cacao?

2.5 ¿Qué otros controles se deben hacer frecuentemente al cacao para evitar la proliferación de enfermedades como la monilia?

Entrevistado 1: Personalmente hacemos rondas cada 2 semanas para revisar las mazorcas del cacao, cualquier bulto, deformación, puntos verdes extraños se deben revisar y si es el caso se arranca esa mazorca en específico, después de arrancar la mazorca usualmente se juntan todas las mazorcas afectada en un rango prudente ya que si se mueven mucho las mazorcas se puede expandir la monilia y es algo que no queremos, ya después de hacer un montón se les deja quietas o si están muy infectadas se amontonan la más infectadas a otro lado para que se les eche cal, algunos agricultores de la zona usan químicos cuando llega la temporada de producción y así se evitan estar haciendo la remoción de mazorcas.

Entrevistado 2: Como controles generales es efectivo hacer rondas en todas las semanas para revisar que si alguna mazorca está afectada por la monilia si se detecta se debe cortar y tumbar la mazorca, esto por el simple hecho de que 6 a 8 días toma para que reaparezcan las esporas he infecten nuevas mazorcas, también hay que tener en cuenta que en el proceso de cortado se deben dejar las mazorcas en la base del cacao si se pueden hacer montones con las mazorcas del cacao afectado mucho mejor después de amontonarlas se les puede echar urea o cal ya que esto ayuda a que se descompongan mas rápido.

3. ¿Qué información deberían conocer los agricultores de primera mano cuando se enfrentan a enfermedades como la monilia?

Entrevistado 1: Uno de los errores más grande que comenten algunos agricultores es el exceso de sombra en el cultivo y esto es un error grave ya que la mayor parte de los hongos proliferan en ambientes de alta humedad, al obstruir la entrada de sol claramente se aumenta la probabilidad de que llegue algún hongo, muchos agricultores no conocen procedimientos generales, controles entre otros, por eso creo que la información de cuidado general de la planta es de vital importancia para cualquier agricultor de cacao.

Entrevistado 2: Gran parte de los agricultores lidian con hongos diría que casi el 80% de los agricultores de cacao sufren de esto, la causa principal es la desinformación en nociones generales de los cultivos, una de ellas la remoción de malezas, las malezas usualmente se presentan en las ramas del cacao o veces enredadas en el tronco, al existir maleza en los cultivos se produce sombra la cual aumenta la humedad del cultivo en general, debido al aumento hongos como la fitoptora o la propia monilia atacan directamente al cacao por esto mismo pienso que estos mantenimientos son importantes, otro de los factores que afectan fuertemente al cultivo de cacao es el número de árboles por hectárea cuadrada si se tiene una densidad de árboles de cacao altas lo más probable es que vaya a

sufrir de enfermedades por el mismo hecho de que no hay espacio para que la luz solar penetre el cultivo y evite la humedad.

4. ¿Qué opina de los controles químicos utilizados en el cacao?

Entrevistado 1: A la mayoría de agricultores que vienen a pedir apoyo o algún consejo en cuanto a su cultivo y se acercan con ideas de usar controles químicos les recomiendo no usarlos, ya que usar químicos no solamente afecta al árbol si no al suelo, sin un suelo fértil se afecta el crecimiento del árbol y así sucesivamente por esto mismo lidiar con afectaciones químicas en el cultivo es de las peores situaciones a las que se puede enfrentar un agricultor.

Entrevistado 2: Personalmente recomiendo que para mayor producción se usen fungicidas en base a cobre ya que no perjudican tanto al cultivo, eso si esta solución, aunque aumenta la producción pero así como aumenta, aumentan los costos a su vez, existen varios tipos y como el cacao se ve mayormente afectado por hongos se recomienda siempre usar fungicidas eso con el cuidado de los químicos que componen al fungicida.

5. ¿Qué otros controles se deben hacer frecuentemente al cacao para evitar la proliferación de enfermedades como la monilia?

Entrevistado 1: Fertilización dos veces al año es los controles más importantes del cultivo del cacao no solo por el hecho de mantener los nutrientes del suelo si no para aumentar la producción y rendimiento de los cultivos, igualmente se ve beneficiada la calidad, el tamaño del fruto entre otros, pero sobre todo lo más importante que es vuelve al cultivo más tolerante a factores ambientales ya sean plagas o enfermedades, otro de estos es la poda tanto la poda de formación que ayuda a que los árboles de cacao mantengan la misma forma o la poda de rehabilitación que ayuda a corregir defectos de crecimiento.

Entrevistado 2: Controles como poda, fertilización, remoción de maleza, control de sombra son de vital importancia tanto para el cuidado general del cultivo como para evitar la proliferación de enfermedad, una de las cosas que siempre se debe recalcar a los agricultores es las actividades básicas del cultivo.

Análisis:

Los consejos que dan los entrevistados la mayoría hacen parte del control cultural general de los cultivos por lo tanto se debe tener en cuenta estos mismos, ya no que no solamente son ellos si no en gran parte de las guías consultadas y artículos que hacen parte de la investigación se nombra el control cultural como la piedra angular del cuidado de los cultivos de cacao, los comentarios acerca de la monilia son muy acertados y se tendrán en cuenta al momento de desarrollar el reconocimiento de imágenes y no solo eso sino también para secciones como información general del cultivo de cacao o la monilia.

ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE

Software: IAcultivar

Ficha del Documento

Fecha	Autor	Verificado
02/09/2020	Andres Rincon Maryury Rodríguez	Juan Sebastián Cárdenas

Documento Validado por las partes en

Gerente	Desarrollador
	Andres Camilo Rincon Caballero Maryury Julieth Rodríguez Calderón

Introducción

Se presenta la Especificación de Requisitos de Software (ERS) que deberá operar en la implementación de la solución informática denominado **IACULTIVAR** que será una plataforma software diseñada para la prevención y detección temprana de las enfermedades como la monilia la cual es una de las más frecuentes y severas en el cultivo de cacao.

Metas

- Determinar todos los requisitos y funcionalidades a desarrollar de la arquitectura de software propuesta (Admin web, API Rest y Aplicación móvil)
- Diseñar los diagramas y esquemas de la arquitectura para saber cómo se relacionan entre sí y que a sí mismo se de cumplimiento con los requerimientos a proponer.
- Desarrollar el API Rest que procesa los datos que se le suministren siendo estos datos fotos de mazorcas de cacao las cuales serán procesadas para posteriormente devolver los datos deseados.
- Desarrollar el admin web que contendrá toda la información de la aplicación esto incluye (información de cacao, consejos, diagnósticos, entre otros)
- Entrenar el modelo de reconocimiento de imágenes
- Construir la Aplicación móvil para sistema operativo Android la cual permita cargar la información que le provea el admin web.

Descripción General

Alcance

Este proyecto tiene como propósito el desarrollo de una aplicación móvil que integre reconocimiento de imágenes con herramientas como custom vision para proporcionar un diagnóstico de la monilia que es una enfermedad presente en el cultivo de cacao, dicho diagnóstico se representará como un porcentaje que indicará la presencia de la enfermedad, para poder visualizar este diagnóstico se deberá contar con acceso a internet y a partir de este dato se aconsejara al agricultor sobre los tratamientos adecuados para prevenir y evitar la propagación de la moniliasis.

Requerimientos del Sistema

Requerimientos No Funcionales

Generales de los 3 sistemas

- El sistema debe tener una disponibilidad del 99,99% de las veces en que un usuario intente acceder.
- El tiempo para iniciar o reiniciar el sistema no podrá ser mayor a 1 minutos.
- La tasa de tiempos de falla del sistema no podrá ser mayor al 0,5% del tiempo de operación total.
- El sistema debe contar con manuales de usuario estructurados adecuadamente.
- Incluir unos datos de contacto directo con un equipo de desarrollo.

Admin Web

- El sistema debe soportar la gestión y almacenamiento de la información de varios usuarios admin
- La aplicación deberá consumir menos de 500 Mb de memoria RAM
- Los tiempos de espera para actualizar información no debe ser mayor 3 segundos
- Los formularios para agregar información una vez finalizados no deben tomar más de 5 segundos para agregar la información a la base de datos

Aplicación móvil

- La aplicación móvil tendrá una interfaz fácil de usar para los usuarios, y coherente visualmente con la plataforma.

- La aplicación móvil será escalable para poder añadir más cultivos o enfermedades agregando nuevas casillas y funcionalidades.
- La aplicación debe poseer un diseño que permita garantizar la adecuada visualización en múltiples dispositivos, tablets y teléfonos inteligentes.
- Tendrá acceso desde cualquier dispositivo móvil con sistema Android que tenga acceso a internet utilizando con datos móviles o por medio de redes wifi entre otras.
- La aplicación debe ser compatible con distintas versiones de Android desde la versión 5.
- El tiempo para iniciar o reiniciar el sistema no podrá ser mayor a 2 minutos.
- La tasa de tiempos de falla del sistema no podrá ser mayor al 0,5% del tiempo de operación total.
- El sistema será desarrollado únicamente para dispositivos con sistema operativo Android.
- La aplicación móvil debe poder funcionar sin la necesidad de ningún tipo de conexión a internet, pero no podrá hacer el análisis de la foto capturada ya que este sí debe contar con un acceso a internet

Requerimientos Funcionales

Admin Web

- El sitio web debe tener un login donde solamente los administradores del sitio podrán ingresar para poder hacer una inspección a la información que se ha agregado o se quisiese añadir.
- El sitio web debe permitir agregar, editar o eliminar varios cultivos pero se iniciara con el cacao, este debe contar con un esquema definido, el cual estará compuesto por datos básicos como: nombre del cultivo, nombre científico del cultivo, descripción del cultivo, ambiente de crecimiento y sus variedades, esto con el objetivo de tener una arquitectura flexible a la hora de querer agregar más cultivos.
- El sitio web debe permitir agregar enfermedades esta contará con una relación directa a los cultivos, si no existe relación con algún cultivo no se podrá agregar la enfermedad (Ejemplo: Monilia afecta al cacao), las enfermedades agregadas deben tener unos datos básicos como: Nombre de la enfermedad, nombre científico, cultivo al que ataca, descripción de la enfermedad, daños hacia el cultivo (Se debe ser específico en el hecho de categorizar en donde se presenta la enfermedad, tallos, flores, hojas, entre otros) (cabe aclarar que solo se podrá relacionar una enfermedad a un cultivo si hay una enfermedad igual que afecta a otro cultivo se tendría que registrar como la misma enfermedad pero afectando a otro cultivo)

- El sitio web debe permitir agregar tratamiento a una enfermedad de un cultivo específico está al mismo tiempo debe contar con una categoría asignada como ‘Control químico’, ‘Control cultural’, ‘Control Biológico’ y ‘Control genético’ está a su vez debe tener datos básicos como: Nombre de la solución, nombre científico (Si lo tiene), descripción y cómo aplicarlo, así como imágenes que ilustran el proceso de aplicación de la solución.
- El sitio web debe mostrar las sugerencias, preguntas o respuestas a preguntas, que los usuarios han digitado en el apartado de sugerencias o el apartado de discusión de la aplicación para que los desarrolladores las tengan en cuenta en caso de alguna mejora.

API Rest

- El api debe tener múltiples conexiones tanto a la aplicación, como el admin web y la base de datos está misma debe asegurarse de que la conexión sea segura por ambas partes
- El api debe tener funciones que conecten a la aplicación con el admin web haciendo que este admin web provea información a la aplicación estas funciones no pueden actualizar la información de la base de datos solo solicitan datos de una a la otra
- Las funciones que tenga esta api deben ser explícitas en cuanto a su nombramiento por ejemplo (Listado cultivos, listado de enfermedades, listado de controles, entre otros) debería traer a la aplicación todos los cultivos que están registrados en la página web, si los cultivos o en este caso la información ya está almacenada localmente en la aplicación no se bajará ningún dato.
- Las funciones relacionadas con subidas de información tales como preguntas, respuestas a preguntas, sugerencias para la aplicación, entre otras deberán comunicarse con el web admin para actualizar esta información lo más pronto posible y así poderla presentar en la aplicación
- El api debe tener funciones para conectarse directamente con el servicio tercerizado de la IA este tendrá conexión entre la IA y la aplicación haciendo que desde la aplicación se suba una foto y esta sea manejada por una función para enviarla a la IA después de enviada a la IA se esperará una respuesta, una vez capturada esta respuesta será interpretada por la API la cual tendrá unos valores asignados para posibles respuestas de la IA y de esta manera dar un diagnóstico al usuario.

Aplicación móvil

- La aplicación en su inicio debe mostrar un listado de los cultivos disponibles en la aplicación siendo el principal el cacao, este listado debe tener un ítem tipo card para cada cultivo, cada ítem debe mostrar el nombre del cultivo acompañado por una pequeña miniatura y una pequeña descripción plegable.
- Cuando se acceda al cultivo seleccionado la pantalla tendrá que mostrar el nombre del cultivo, el nombre científico, descripción, imágenes ilustrativas, variedades de cultivo y aparte de esto se mostrarán las enfermedades que afectan a este cultivo.
- Cuando se acceda a la pestaña de enfermedades se mostrarán datos como el nombre de la enfermedad imágenes asociadas al cultivo donde se encuentre, nombre científico, descripción, junto con las diferentes soluciones a la enfermedad desde varios puntos tales como el 'Control químico', 'Control cultural', 'Control Biológico' y 'Control genético'
- La aplicación móvil debe tener un apartado para detección de la monilia en la mazorca del cacao, está debe pedir permisos para poder usar la cámara del teléfono, una vez concedidos el usuario podrá tomar una foto de la mazorca del cacao esta será procesada por el API para dar un diagnóstico al usuario, una vez dado el diagnóstico se le redirigirá al apartado de la enfermedad en caso de que llegue a estar enferma para que aplique los controles correspondientes.
- La aplicación móvil debe tener un apartado de sugerencias para que los usuarios puedan escribir qué mejoras se le pueden hacer a la aplicación, este apartado de sugerencias sólo pedirá el nombre de la persona y la sugerencia correspondiente, esta sugerencia será almacenada en la base de datos del proyecto y posteriormente será mostrada en la página web para qué los administradores la tengan en cuenta en dado caso de que haya que realizar alguna mejora
- Si el usuario no tiene conexión a internet podrá subir fotos desde un apartado fuera de la detección en vivo, este apartado permitirá subir una o varias imágenes para ser procesadas y tener un diagnóstico de las mismas.
- La aplicación manejará una base de datos interna directamente con el móvil donde almacenará los datos que suministrará la aplicación tales como: contenidos, imágenes, consejos, entre otros.
- La aplicación deberá tener una sección de discusión donde los usuarios puedan preguntar y responder preguntas hechas por otros usuarios acerca de los cultivos o funcionamiento de algún apartado de la aplicación.
- La sección de discusión deberá ser el único lugar donde el usuario deba estar logeado para participar de ella.

Roles Del Sistema

Usuario:

Rol por defecto, este rol permite utilizar la aplicación móvil para descargar la información de uno o más cultivos a través de una conexión a internet y posteriormente visualizar la información organizada según la estructura planteada en los requerimientos. Este rol no podrá ingresar al sistema de IAcultivar y no podrá ingresar ningún nuevo registro modificar información existente.

Usuario foro:

Este rol tiene como funcionalidad el poder preguntar y opinar en la sección de discusión, el usuario tiene qué estar logeado para hacer uso de esta funcionalidad.

Roles Del Sistema

Administrador (Solo admin web)

Rol otorgado a únicamente las personas dedicadas a actualizar la información de la base de datos desde la página web, aceptar sugerencias de los usuarios basados en los criterios personales para qué estos aparezcan en la aplicación móvil y evaluar cuáles preguntas deben ser puestas en la sección de preguntas frecuentes, este también podrá crear usuarios del foro.

Anexo C Documento de casos de uso

CASOS DE USO IACULTIVAR

ANDRES CAMILO RINCON CABALLERO

MARYURY JULIETH RODRIGUEZ CALDERON

**UNIVERSIDAD AUTÓNOMA DE BUCARAMANGA
FACULTAD INGENIERÍA DE SISTEMAS
TECNOLOGÍA Y SOCIEDAD
BUCARAMANGA
2020**

Tabla de Contenido

1. INTRODUCCIÓN	
251.1	
Propósito	
271.2	
Alcance	
1191.3	Definiciones, Siglas y
Abreviaturas	
1191.4	
Referencias	
382.	IDENTIFICACIÓN DE ACTORES DEL
SISTEMA	
1202.1	DESCRIPCIÓN DE ACTORES DEL
SISTEMA	
1202.1.1	
<i>Programador</i>	
1203.	DIAGRAMA DE CASOS DE
USO 1244.	ESPECIFICACIÓN DE CASO DE
USO 126	

1. INTRODUCCIÓN

1.1 Propósito

Este documento tiene como propósito mostrar cada uno de los casos de uso y los actores que componen el sistema de “IAcultivar”

1.2 Alcance

El alcance de este documento va desde la muestra de cómo se distribuyen los casos de uso a lo largo del sistema, hasta la descripción a detalle de los mismos y de quienes actúan sobre ellos.

1.3 Definiciones, Siglas y Abreviaturas

UML

IA

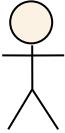
API

2. IDENTIFICACIÓN DE ACTORES DEL SISTEMA

- Administrador/Programador
- Usuario
- Admin web
- App

2.1 DESCRIPCIÓN DE ACTORES DEL SISTEMA

2.1.1 Administrador/Programador

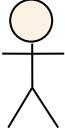
	<p><i>NOMBRE DEL ACTOR: Administrador / Programador</i></p>
	<p><i>Descripción:</i></p> <p>Se encarga de actualizar, editar, crear y eliminar información en el admin web, también leer y analizar las sugerencias hechas por los usuarios para después implementarlas dentro del sistema.</p>
	<p>CARACTERÍSTICAS:</p> <p>Crear información: Agrega información relevante desde el sitio web hacia la base de datos, para que esta se muestre en la aplicación móvil.</p> <p>Leer información: Lee y envía información a la aplicación móvil cada vez que se necesite, esta información la extrae directamente de la base de datos.</p> <p>Actualizar información: Actualiza cualquier tipo de información desde el sitio web, dependiendo de las necesidades del sistema, ya sea porque la información se errónea o porque la información ya no es relevante para el contexto actual del negocio.</p> <p>Eliminar información: Elimina cualquier tipo de información desde el sitio web, esto lo hace desde su criterio propio evaluando si la información a ser eliminada es errónea o puede ser dañina para los usuarios o no es relevante para el contexto actual del negocio.</p> <p>Analizar sugerencias: Analiza cualquier sugerencia hecha por los usuarios a través del móvil y evalúa si esta sugerencia puede ser válida para ser tomada en cuenta e implementarla en la aplicación móvil.</p>

Entrenamiento de la IA: Se encarga de suministrar al módulo de reconocimiento de imágenes los datos necesarios para que este funcione correctamente, también se encarga de verificar que el módulo esté funcionando correctamente y a su vez hace pruebas sobre este.

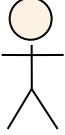
RELACIONES:

Admin web

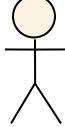
2.1.2 Aplicación móvil

	<p>NOMBRE DEL ACTOR: [Aplicación móvil]</p> <p>DESCRIPCIÓN: Se encarga de mostrar al usuario los diferentes cultivos, enfermedades de cultivos y las soluciones que se le pueden dar para el control y manejo de la enfermedad.</p>
<p>CARACTERÍSTICAS:</p> <p>[Explicación de las actividades en las que interviene relacionadas con el sistema]</p> <p>Mostrar información: la aplicación se encarga de mostrar al usuario los diferentes listados de cultivos y enfermedades del mismo, aparte de esto mostrará diferentes tipos de soluciones</p> <p>Diagnóstico: la aplicación móvil se comunica con el API y este a su vez con la IA la cual es la encargada de realizar un análisis especializado a las imágenes para arrojar un porcentaje de conciencia de la imagen con la enfermedad este porcentaje es el diagnóstico de la enfermedad y es el que se le mostrará al usuario.</p> <p>Discusiones: Apartado en el que los diferentes usuarios podrán realizar preguntas y responderlas esto, pero para poder realizar preguntas tendrá que registrarse con un nombre de usuario, cargo desempeñado y en caso de estar especializado en algún tipo de cultivo o enfermedad colocar esta información en forma de descripción personal, esto con el fin de que todos los usuarios puedan tener conocimiento de que tipo de información puede brindar dicha persona.</p> <p>Sugerencias: los usuarios podrán ingresar al apartado de sugerencias con el fin de hacer sugerencias respecto a al aspecto, funcionamiento entre otros de la aplicación.</p> <p>RELACIONES:</p> <p>Usuario, Admin web</p>	

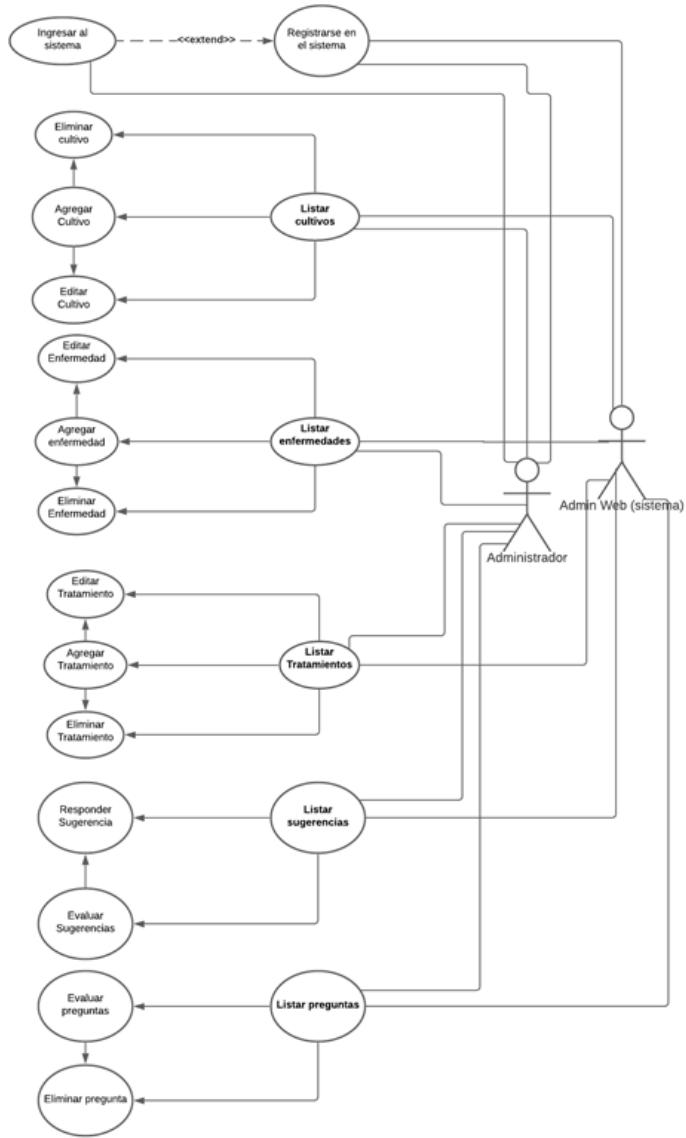
2.1.3 Admin web

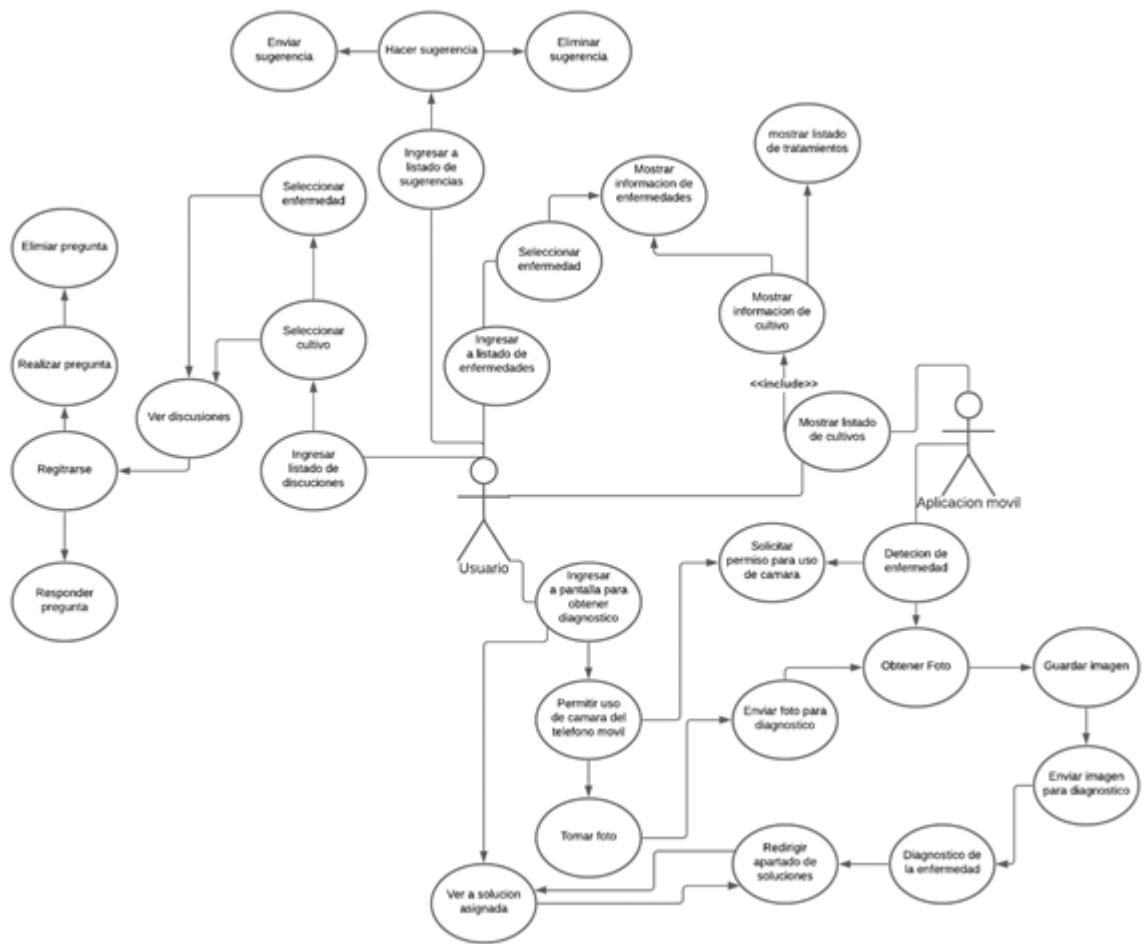
	<p><i>NOMBRE DEL ACTOR:</i> [Admin web]</p>
	<p><i>DESCRIPCIÓN:</i> Permite suministrar a la aplicación con todos los datos necesarios para que esta funcione de manera correcta, siendo estos datos cultivos, enfermedades, tratamientos, discusiones en el foro entre otros, al mismo tiempo actúa sobre la base de datos agregando y extrayendo información de la misma.</p>
	<p><i>CARACTERÍSTICAS:</i> Suministrar información: El admin web al poseer en su sistema al mismo API, suministra la información que va solicitando la aplicación móvil por medio de peticiones HTTP, esta información es acerca de cultivos, enfermedades, tratamientos, discusiones en foros y diagnóstico de enfermedades</p> <p><i>Comunicación con la base de datos:</i> se encarga de establecer la conexión entre la base de datos y al mismo tiempo actúa sobre ella, estas acciones van desde la edición de contenido de la misma, hasta agregación y eliminación de campos.</p>
	<p><i>RELACIONES:</i> Administrador / Programador</p>

2.1.4 Usuario

	<p><i>NOMBRE DEL ACTOR:</i> [Usuario]</p>
	<p><i>DESCRIPCIÓN:</i> Usuario final de la aplicación móvil y web, es quien tendrá acceso a información de los cultivos registrados, y a cada una de las enfermedades asociadas a este.</p>
	<p><i>CARACTERÍSTICAS:</i></p> <p>Información: Accede a información sobre los cultivos y enfermedades asociadas al mismo; dicha enfermedad cuenta con un apartado de tratamientos o soluciones a las que el usuario puede acceder.</p> <p>Diagnóstico: Encargado de dar acceso a la cámara del teléfono móvil y tomar la foto, esta información será redirigida y analizada para dar un diagnóstico del estado de la mazorca</p>
	<p><i>RELACIONES:</i></p>
	<p>App</p>

3. DIAGRAMA DE CASOS DE USO





4. ESPECIFICACIÓN DE CASO DE USO

Identificación del caso de uso	
ID	1
Nombre	Listar cultivos
Actores	Admin web
Descripción breve	Este caso de uso se encarga de listar los cultivos en el admin web, se activa cuando el administrador pulsa el botón de listado de cultivos y como resultado debe entregar una lista cultivos
Precondiciones	El admin debe estar logueado y haber seleccionado el listado de cultivos en la pantalla principal del admin web
Postcondiciones	En la pantalla debe aparecer la lista de cultivos disponibles en el sitio web
Flujo de eventos	Una vez seleccionado el listado de cultivos se mostrará la lista de los cultivos disponibles
Flujo básico	<p>El administrador selecciona el listado de cultivos.</p> <p>Se lee la base de datos en donde se encuentra el registro de los cultivos.</p> <p>Se muestra el resultado en la pantalla en forma de listado.</p> <p>Una vez mostrado el resultado el administrador puede:</p> <ul style="list-style-type: none"> -Organizarlo de manera alfabética

Frecuencia de uso	Caso de uso de frecuencia diaria
Include	Ninguno
Extend	Ninguno

Identificación del caso de uso	
ID	2
Nombre	Agregar Cultivo
Actores	Admin web, Administrador
Descripción breve	Este caso de uso se encarga de agregar cultivos al listado de cultivos del admin web
Precondiciones	El admin debe estar logueado, estar en el listado de cultivos y haber seleccionado la opción de agregar cultivo
Poscondiciones	En la pantalla debe aparecer creación exitosa y volver al listado de cultivos
Flujo de eventos	Una vez pulsado el botón de agregar cultivo, se mostrará un formulario para agregar cultivos.

	<p>Administrador selecciona en agregar cultivo</p> <p>El sistema lo redirecciona a un formulario con los datos para agregar un cultivo</p> <p>En pantalla debe estar claro lo que el administrador ha digitado</p>
Flujo básico	<p>Una vez llenó el formulario el administrador debe dar click en el botón agregar cultivo</p> <p>Una vez agregado el cultivo, se mostrará una pantalla la cual dirá creación exitosa y debe devolver al administrador al listado de cultivos para corroborar que efectivamente se agregó el cultivo</p>
Frecuencia de uso	Cada vez que sea necesario
Include	Listar cultivos
Extend	Ninguno

Identificación del caso de uso	
ID	3
Nombre	Eliminar cultivo
Actores	Admin web, administrador
Descripción breve	Este caso de uso se encarga de eliminar los cultivos que el administrador desee eliminar, para esto debe pulsar el botón de eliminar cultivo que estará posicionado en el cultivo a eliminar

Precondiciones	El admin debe estar logueado, estar en el listado de cultivos y haber seleccionado la opción de eliminar cultivo
Poscondiciones	En la pantalla debe aparecer el mensaje eliminación exitosa y devolver al administrador a la pantalla de cultivos
Flujo de eventos	Una vez pulsado el botón de eliminar cultivo, se mostrará un mensaje de confirmación de eliminación, una vez pulsado se debe mostrar la lista de cultivos sin el cultivo eliminado
Flujo básico	<p>Administrador selecciona en eliminar cultivo</p> <p>El sistema le muestra un mensaje de confirmación de eliminación</p> <p>Una vez el administrador confirme la eliminación, se mostrará un mensaje el cual dirá eliminación exitosa y debe devolver al administrador al listado de cultivos para corroborar que efectivamente se eliminó el cultivo</p>
Frecuencia de uso	Cada vez que sea necesario
Include	Listar cultivos
Extend	Ninguno

Identificación del caso de uso	
ID	4
Nombre	Editar cultivo
Actores	Admin web, Administrador
Descripción breve	Este caso de uso se encarga de editar la información de un cultivo en específico que seleccione el administrador en el admin web
Precondiciones	El admin debe estar logueado, estar en el listado de cultivos y haber seleccionado la opción de editar cultivo
Poscondiciones	En la pantalla debe aparecer cambios realizados y volver al listado de cultivos
Flujo de eventos	Una vez pulsado el botón de editar cultivo, se mostrará un formulario para editar cultivos.
Flujo básico	<p>Administrador selecciona en editar cultivo</p> <p>El sistema lo redirecciona a un formulario con los datos para editar un cultivo</p> <p>En pantalla debe estar claro lo que el administrador ha digitado</p> <p>Una vez llenó el formulario el administrador debe dar click en el botón guardar cambios</p> <p>Una vez editado el cultivo, se mostrará una pantalla la cual dirá edición exitosa y debe devolver al administrador al listado de cultivos para corroborar que efectivamente se editó el cultivo</p>

Frecuencia de uso	Cada vez que sea necesario
Include	Listar cultivos
Extend	Ninguno

Identificación del caso de uso	
ID	
Nombre	Listar enfermedades
Actores	Admin web
Descripción breve	Este caso de uso se encarga de listar las enfermedades en el admin web, se activa cuando el administrador pulsa el botón de listado de enfermedades y como resultado debe entregar una lista enfermedades
Precondiciones	El admin debe estar logueado y haber seleccionado el listado de enfermedades en la pantalla principal del admin web
Poscondiciones	En la pantalla debe aparecer la lista de enfermedades disponibles en el sitio web
Flujo de eventos	Una vez seleccionado el listado de enfermedades se mostrará la lista de las enfermedades disponibles

Flujo básico	<p>Administrador selecciona el listado de enfermedades</p> <p>Se lee la base de datos en donde se encuentra el registro de las enfermedades.</p> <p>Se muestra el resultado en la pantalla en forma de listado.</p> <p>Una vez mostrado el resultado el administrador puede:</p> <ul style="list-style-type: none"> -Organizarlo de manera alfabética
Frecuencia de uso	Caso de uso de frecuencia diaria
Include	Ninguno
Extend	Ninguno

Identificación del caso de uso	
ID	2
Nombre	Agregar enfermedades
Actores	Admin web, Administrador
Descripción breve	Este caso de uso se encarga de agregar enfermedades al listado de enfermedades del admin web

Precondiciones	El admin debe estar logueado, estar en el listado de enfermedades y haber seleccionado la opción de agregar enfermedad
Poscondiciones	En la pantalla debe aparecer creación exitosa y volver al listado de enfermedades
Flujo de eventos	Una vez pulsado el botón de agregar enfermedad, se mostrará un formulario para agregar enfermedades.
Flujo básico	<p>Administrador selecciona en agregar enfermedad</p> <p>El sistema lo redirecciona a un formulario con los datos para agregar una enfermedad</p> <p>En pantalla debe estar claro lo que el administrador ha digitado</p> <p>Una vez llenó el formulario el administrador debe dar click en el botón agregar enfermedad</p> <p>Una vez agregada la enfermedad, se mostrará una pantalla la cual dirá creación exitosa y debe devolver al administrador al listado de enfermedades para corroborar que efectivamente se agregó la enfermedad</p>
Frecuencia de uso	Cada vez que sea necesario
Include	Listar enfermedades
Extend	Ninguno

Identificación del caso de uso

ID	3
Nombre	Eliminar enfermedad
Actores	Admin web, Administrador
Descripción breve	Este caso de uso se encarga de eliminar las enfermedades que el administrador desee eliminar, para esto debe pulsar el botón de eliminar enfermedad que estará posicionado en la enfermedad a eliminar
Precondiciones	El admin debe estar logueado, estar en el listado de enfermedades y haber seleccionado la opción de eliminar enfermedad
Poscondiciones	En la pantalla debe aparecer el mensaje eliminación exitosa y devolver al administrador a la pantalla de enfermedades
Flujo de eventos	Una vez pulsado el botón de eliminar enfermedad se mostrará un mensaje de confirmación de eliminación, una vez pulsado se debe mostrar la lista de enfermedad sin la enfermedad eliminado
Flujo básico	Administrador selecciona en eliminar enfermedad El sistema le muestra un mensaje de confirmación de eliminación Una vez el administrador confirme la eliminación, se mostrará un mensaje el cual dirá eliminación exitosa y debe devolver al administrador al listado de enfermedades para corroborar que efectivamente se eliminó la enfermedad
	Cada vez que sea necesario

Frecuencia de uso	
Include	Listar enfermedades
Extend	Ninguno

Identificación del caso de uso	
ID	4
Nombre	Editar enfermedades
Actores	Admin web, Administrador
Descripción breve	Este caso de uso se encarga de editar la información de una enfermedad en específico que seleccione el administrador en el admin web
Precondiciones	El admin debe estar logueado, estar en el listado de enfermedades y haber seleccionado la opción de editar enfermedad
Poscondiciones	En la pantalla debe aparecer cambios realizados y volver al listado de enfermedades
Flujo de eventos	Una vez pulsado el botón de editar enfermedad se mostrará un formulario para editar enfermedades
Flujo básico	Administrador selecciona en editar enfermedad El sistema lo redirecciona a un formulario con los datos para editar una enfermedad En pantalla debe estar claro lo que el administrador ha digitado

	<p>Una vez llenó el formulario el administrador debe dar click en el botón guardar cambios</p> <p>Una vez editada la enfermedad se mostrará una pantalla la cual dirá edición exitosa y debe devolver al administrador al listado de enfermedades para corroborar que efectivamente se editó la enfermedad</p>
Frecuencia de uso	Cada vez que sea necesario
Include	Listar enfermedades
Extend	Ninguno

Identificación del caso de uso	
ID	
Nombre	Listar tratamientos
Actores	Admin web
Descripción breve	Este caso de uso se encarga de listar los tratamientos en el admin web, se activa cuando el administrador pulsa el botón de listado de tratamientos y como resultado debe entregar una lista tratamientos
Precondiciones	El admin debe estar logueado y haber seleccionado el listado de tratamientos en la pantalla principal del admin web
	En la pantalla debe aparecer la lista de tratamientos disponibles en el sitio web

Poscondiciones	
Flujo de eventos	Una vez seleccionado el listado de tratamientos se mostrará la lista de los tratamientos disponibles
Flujo básico	<p>Administrador selecciona el listado de tratamientos</p> <p>Se lee la base de datos en donde se encuentra el registro de los tratamientos</p> <p>Se muestra el resultado en la pantalla en forma de listado.</p> <p>Una vez mostrado el resultado el administrador puede:</p> <ul style="list-style-type: none"> -Organizarlo de manera alfabética
Frecuencia de uso	Caso de uso de frecuencia diaria
Include	Ninguno
Extend	Ninguno

Identificación del caso de uso	
ID	2
Nombre	Agregar tratamientos
Actores	Admin web, Administrador

Descripción breve	Este caso de uso se encarga de agregar tratamientos al listado de tratamientos del admin web
Precondiciones	El admin debe estar logueado, estar en el listado de tratamientos y haber seleccionado la opción de agregar tratamiento
Poscondiciones	En la pantalla debe aparecer creación exitosa y volver al listado de tratamientos
Flujo de eventos	Una vez pulsado el botón de agregar tratamientos, se mostrará un formulario para agregar tratamientos
Flujo básico	<p>Administrador selecciona en agregar tratamiento</p> <p>El sistema lo redirecciona a un formulario con los datos para agregar un tratamiento</p> <p>En pantalla debe estar claro lo que el administrador ha digitado</p> <p>Una vez llenó el formulario el administrador debe dar click en el botón agregar tratamiento</p> <p>Una vez agregado el tratamiento, se mostrará una pantalla la cual dirá creación exitosa y debe devolver al administrador al listado de tratamientos para corroborar que efectivamente se agregó el tratamiento</p>
Frecuencia de uso	Cada vez que sea necesario
Include	Listar tratamientos

Extend	Ninguno
--------	---------

Identificación del caso de uso	
ID	3
Nombre	Eliminar tratamiento
Actores	Admin web, Administrador
Descripción breve	Este caso de uso se encarga de eliminar los tratamientos que el administrador desee eliminar, para esto debe pulsar el botón de eliminar tratamiento que estará posicionado en el tratamiento a eliminar
Precondiciones	El admin debe estar logueado, estar en el listado de tratamientos y haber seleccionado la opción de eliminar tratamiento
Poscondiciones	En la pantalla debe aparecer el mensaje eliminación exitosa y devolver al administrador a la pantalla de tratamientos
Flujo de eventos	Una vez pulsado el botón de eliminar tratamiento se mostrará un mensaje de confirmación de eliminación, una vez pulsado se debe mostrar la lista de tratamientos sin el tratamiento eliminado
Flujo básico	Administrador selecciona en eliminar tratamiento El sistema le muestra un mensaje de confirmación de eliminación Una vez el administrador confirme la eliminación, se mostrará un mensaje el cual dirá eliminación exitosa y debe devolver al administrador al listado de tratamientos para corroborar que efectivamente se eliminó el tratamiento

Frecuencia de uso	Cada vez que sea necesario
Include	Listar tratamientos
Extend	Ninguno

Identificación del caso de uso	
ID	4
Nombre	Editar tratamientos
Actores	Admin web, Administrador
Descripción breve	Este caso de uso se encarga de editar la información de un tratamiento en específico que seleccione el administrador en el admin web
Precondiciones	El admin debe estar logueado, estar en el listado de tratamientos y haber seleccionado la opción de editar tratamiento
Poscondiciones	En la pantalla debe aparecer cambios realizados y volver al listado de tratamientos

Flujo de eventos	Una vez pulsado el botón de editar tratamiento se mostrará un formulario para editar tratamientos
Flujo básico	<p>Administrador selecciona en editar tratamientos</p> <p>El sistema lo redirecciona a un formulario con los datos para editar un tratamiento</p> <p>En pantalla debe estar claro lo que el administrador ha digitado</p> <p>Una vez llenó el formulario el administrador debe dar click en el botón guardar cambios</p> <p>Una vez editada el tratamiento se mostrará una pantalla la cual dirá edición exitosa y debe devolver al administrador al listado de tratamientos para corroborar que efectivamente se editó el tratamiento</p>
Frecuencia de uso	Cada vez que sea necesario
Include	Listar tratamientos
Extend	Ninguno

Identificación del caso de uso	
ID	3
Nombre	Ingresar al sistema
Actores	Administrador

Descripción breve	Este caso de uso permite que el administrador se puede logear en el admin web para poder realizar sus tareas de administración del sitio
Precondiciones	El administrador debe haber ingresado a la página web y estar en la ventana de logeo
Poscondiciones	En la pantalla debe aparecer el mensaje log in exitoso y dirigir al administrador a la ventana de administración del sitio
Flujo de eventos	Una vez digitadas las credenciales en la ventana de login, se confirmara si el usuario es válido o no, una vez validado se redirigirá a la ventana de administración
Flujo básico	<p>El administrador debe ingresar al sitio</p> <p>una vez en el sitio debe ingresar a la ventana de log in</p> <p>una vez digitadas las credenciales en la ventana de login se evaluará si el usuario es correcto o no</p> <p>una vez se valide que el usuario es correcto se redirigirá al usuario a la ventana de administración</p>
Frecuencia de uso	Uso diario
Include	Ninguno
Extend	Registrarse en el sistema

Identificación del caso de uso	
ID	3
Nombre	Registrarse en el sistema
Actores	Admin web, Administrador
Descripción breve	Este caso de uso permite que los administradores nuevo se registren en el sistema
Precondiciones	El nuevo administrador debe haber ingresado a la página web, estar en la ventana de logeo y haber pulsado el botón registrarse
Poscondiciones	En la pantalla debe aparecer el mensaje registro exitoso y dirigir al nuevo administrador a la ventana de login del sitio
Flujo de eventos	Una vez digitadas las credenciales en la ventana de registro, se confirmará si los datos son válidos o no, una vez validados los datos se redirigirá al nuevo administrador a la ventana de log in.
Flujo básico	<p>El nuevo administrador debe ingresar al sitio</p> <p>una vez en el sitio debe ingresar a la ventana de log in</p> <p>una vez en la ventana de login debe presionar el botón de registrarse</p> <p>una vez digitadas las credenciales en la ventana de registro se evaluará si los datos son correctos o no</p> <p>una vez se validen los datos del nuevo usuario se redirigirá al nuevo administrador a la ventana de log in</p>

Frecuencia de uso	Cada vez que sea necesario
Include	Ingresar al sistema
Extend	Ninguno

Identificación del caso de uso	
ID	3
Nombre	Listar preguntas
Actores	Admin web
Descripción breve	Este caso de uso se encarga de listar las preguntas realizadas por los usuarios de la aplicación en el admin web, se activa cuando el administrador pulsa el botón de listado de preguntas y como resultado debe entregar una lista de preguntas
Precondiciones	El admin debe estar logueado y haber seleccionado el listado de preguntas en la pantalla principal del admin web
Poscondiciones	En la pantalla debe aparecer la lista de preguntas disponibles en el sitio web
Flujo de eventos	Una vez seleccionado el listado de preguntas se mostrará la lista de las preguntas hechas por los usuarios

Flujo básico	<p>Administrador selecciona el listado de preguntas</p> <p>Se lee la base de datos en donde se encuentra el registro de las preguntas</p> <p>Se muestra el resultado en la pantalla en forma de listado.</p> <p>Una vez mostrado el resultado el administrador puede:</p> <ul style="list-style-type: none"> -Organizarlo de manera alfabética -Organizarlo por el número de respuestas a esa pregunta
Frecuencia de uso	Caso de uso de frecuencia diaria
Include	Ninguno
Extend	Ninguno

Identificación del caso de uso	
ID	3
Nombre	Evaluuar preguntas
Actores	Admin web, Administrador
Descripción breve	Este caso de uso se encarga de evaluar las preguntas de modo que el administrador identifique si son apropiadas para la aplicación.

Precondiciones	El admin debe estar logueado, haber seleccionado el listado de preguntas en la pantalla principal del admin web y seleccionar la pregunta a evaluar
Poscondiciones	En la pantalla debe aparecer el contenido de la pregunta.
Flujo de eventos	Una vez seleccionado el listado de preguntas, se debe seleccionar una de las preguntas para evaluar su contenido.
Flujo básico	<p>Administrador selecciona el listado de preguntas</p> <p>Se lee la base de datos en donde se encuentra el registro de las preguntas</p> <p>Se muestra el resultado en la pantalla en forma de listado.</p> <p>El administrador selecciona una de las preguntas</p> <p>A su vez evalúa el contenido de la misma con un sistema de puntaje</p> <p>Una vez dado el puntaje se guardarán los cambios y se redirigirá a la ventana del listado de preguntas.</p>
Frecuencia de uso	Caso de uso de frecuencia diaria
Include	Listar preguntas
Extend	Ninguno

Identificación del caso de uso

ID	3
Nombre	Eliminar preguntas
Actores	Admin web, Administrador
Descripción breve	Este caso de uso se encarga de eliminar las preguntas que el administrador desee eliminar, para esto debe pulsar el botón de eliminar pregunta que estará posicionado en la pregunta a eliminar
Precondiciones	El admin debe estar logueado, estar en el listado de preguntas y haber seleccionado la opción de eliminar pregunta
Poscondiciones	En la pantalla debe aparecer el mensaje eliminación exitosa y devolver al administrador a la pantalla de preguntas
Flujo de eventos	Una vez pulsado el botón de eliminar pregunta se mostrará un mensaje de confirmación de eliminación, una vez pulsado se debe mostrar la lista de preguntas sin la pregunta eliminada
Flujo básico	<p>Administrador selecciona en eliminar pregunta</p> <p>El sistema le muestra un mensaje de confirmación de eliminación</p> <p>Una vez el administrador confirme la eliminación, se mostrará un mensaje el cual dirá eliminación exitosa y debe devolver al administrador al listado de preguntas para corroborar que efectivamente se eliminó la pregunta</p>
	Cada vez que sea necesario

Frecuencia de uso	
Include	Listar preguntas
Extend	Ninguno

Identificación del caso de uso	
ID	3
Nombre	Listar sugerencias
Actores	Admin web
Descripción breve	Este caso de uso se encarga de listar las sugerencias realizadas por los usuarios de la aplicación en el admin web, se activa cuando el administrador pulsa el botón de listado de preguntas y como resultado debe entregar una lista sugerencias
Precondiciones	El admin debe estar logueado y haber seleccionado el listado de sugerencias en la pantalla principal del admin web
Poscondiciones	En la pantalla debe aparecer la lista de sugerencias disponibles en el sitio web
Flujo de eventos	Una vez seleccionado el listado de sugerencias se mostrará la lista de las sugerencias hechas por los usuarios

Flujo básico	<p>Administrador selecciona el listado de sugerencias</p> <p>Se lee la base de datos en donde se encuentra el registro de las sugerencias</p> <p>Se muestra el resultado en la pantalla en forma de listado.</p> <p>Una vez mostrado el resultado el administrador puede:</p> <ul style="list-style-type: none"> -Organizarlo de manera alfabética
Frecuencia de uso	Caso de uso de frecuencia diaria
Include	Ninguno
Extend	Ninguno

Identificación del caso de uso	
ID	3
Nombre	Evaluuar sugerencias
Actores	Admin web, Administrador
Descripción breve	Este caso de uso se encarga de evaluar las sugerencias de modo que el administrador identifique qué tan importantes son para el desarrollo de la aplicación o su estructura de negocio

Precondiciones	El admin debe estar logueado, haber seleccionado el listado de sugerencias en la pantalla principal del admin web y seleccionar la sugerencia a evaluar
Poscondiciones	En la pantalla debe aparecer el contenido de la sugerencia
Flujo de eventos	Una vez seleccionado el listado de sugerencias, se debe seleccionar una de las sugerencias para evaluar su nivel de importancia.
Flujo básico	<p>Administrador selecciona el listado de sugerencias</p> <p>Se lee la base de datos en donde se encuentra el registro de las sugerencias</p> <p>Se muestra el resultado en la pantalla en forma de listado.</p> <p>El administrador selecciona una de las sugerencias</p> <p>A su vez evalúa el contenido de la misma con un sistema de puntaje</p> <p>Una vez dado el puntaje se guardarán los cambios y se redirigirá a la ventana del listado de sugerencias</p>
Frecuencia de uso	Caso de uso de frecuencia diaria
Include	Listar sugerencias
Extend	Ninguno

Identificación del caso de uso

ID		3
Nombre	Responder sugerencia	
Actores	Admin web, Administrador	
Descripción breve	Este caso de uso se encarga de dar una respuesta a la sugerencia por parte del administrador, para esto debe pulsar el botón de responder sugerencia que estará posicionado en la sugerencia a responder	
Precondiciones	El admin debe estar logueado, estar en el listado de preguntas y haber seleccionado la opción de responder sugerencia, la sugerencia debe estar evaluada previamente para poder dar una respuesta	
Poscondiciones	En la pantalla debe aparecer el mensaje respuesta exitosa y devolver al administrador a la pantalla de sugerencias	
Flujo de eventos	Una vez pulsado el botón de responder sugerencia se mostrará un panel para dar respuesta a la sugerencia del usuario, una vez pulsado se debe mostrar la lista de sugerencias con un check marcando que la sugerencia ha sido respondida	
Flujo básico	<p>Administrador selecciona la sugerencia a responder</p> <p>se mostrará un pequeño panel para dar respuesta a la sugerencia</p> <p>El sistema le muestra un mensaje de respuesta exitosa al administrador</p> <p>una vez dada la respuesta el sistema debe devolver al administrador al listado de sugerencias</p>	

Frecuencia de uso	Cada vez que sea necesario
Include	Listar sugerencias
Extend	Ninguno

Identificación del caso de uso	
ID	1
Nombre	Mostrar listado de cultivos
Actores	App
Descripción breve	Este caso se encarga de listar los cultivos ya registrados por el administrador, en el momento que el usuario haga click sobre cultivos el podrá ver el listado de los diferentes cultivos en modo de lista con una pequeña imagen que lo identifica
Precondiciones	El usuario debe tener la aplicación instalada en su dispositivo móvil, hacer clic en el apartado de Cultivos
Postcondiciones	Al dar click sobre la casilla de cultivos se desplegará los diferentes cultivos registrados
Flujo de eventos	Elegir el cultivo del cual quiere obtener mayor información
Flujo básico	1. Entrar a la aplicación 2. dirigirse a la casilla de cultivos 3. elegir el cultivo del que desea tener más información
Frecuencia de uso	Frecuencia diaria

Include	No hay
Extend	No hay

Identificación del caso de uso	
ID	2
Nombre	Mostrar información de cultivo
Actores	App
Descripción breve	Cuando se acceda al cultivo seleccionado la pantalla que mostrará el nombre del cultivo, el nombre científico, descripción, imágenes ilustrativas, variedades de cultivo y aparte de esto, se mostrarán las enfermedades que afectan a este cultivo.
Precondiciones	El usuario debe haber seleccionado el cultivo del que desea obtener información
Postcondiciones	Al seleccionar el cultivo y ver toda la información después se podrá acceder al apartado de enfermedades
Flujo de eventos	Elegir un cultivo, Revisar la información
Flujo básico	Ingresar al apartado de cultivos, Elegir un cultivo del que desea obtener más información, elegir las enfermedades que lo afectan

Frecuencia de uso	DIARIA
Include	
Extend	

Identificación del caso de uso	
ID	3
Nombre	Mostrar información de enfermedades
Actores	App
Descripción breve	Cuando se acceda a la pestaña de enfermedades se mostraran datos como el nombre de la enfermedad, imágenes asociadas al cultivo donde se encuentre, nombre científico, parte de la planta a la que ataca, etapas de la enfermedad y descripción de desarrollo de la misma
Precondiciones	El usuario previamente debe haber elegido el cultivo del que desea obtener más información y la enfermedad de la que desea conocer más
Poscondiciones	Revisar la información de la enfermedad previamente seleccionada

Flujo de eventos	Ingresar al apartado de los cultivos y seleccionar la enfermedad de la que se desea saber más información
Flujo básico	1. ingresar a los cultivos 2. ingresar a un cultivo 3. ingresar a enfermedades 4. ingresar a una enfermedad específica
Frecuencia de uso	DIARIA
Include	
Extend	

Identificación del caso de uso	
ID	4
Nombre	Mostrar listado de tratamientos
Actores	App
Descripción breve	en este apartado se encuentran las soluciones a la enfermedad desde varios puntos tales como el 'Control químico', 'Control cultural', 'Control Biológico' y 'Control genético'
Precondiciones	Tener seleccionado un cultivo, la enfermedad que lo afecta para así poder mostrar la información de el tratamiento para dicha enfermedad

Postcondiciones	Revisar la información proporcionada y evaluar cuál es el tratamiento que más se ajusta al usuario.
Flujo de eventos	Una vez seleccionado el apartado de tratamiento queda a criterio y evaluación del usuario que tratamiento elegir
Flujo básico	ingresar al listado de cultivos, seleccionar cultivo, seleccionar el apartado enfermedades, seleccionar una enfermedad
Frecuencia de uso	DIARIA
Include	
Extend	

Identificación del caso de uso	
ID	7
Nombre	Seleccionar Enfermedad
Actores	Usuario
Descripción breve	Este caso de uso permite que el usuario pueda seleccionar del listado de enfermedades una de la que desee conocer más información
precondiciones	El usuario debe haber ingresado a la app y dirigido al apartado de cultivos y enfermedades del mismo

Postcondiciones	En la pantalla aparecerá la toda la información de la enfermedad como lo es nombre tradicional y científico, parte de la planta que más se ven afectadas, ciclo de la enfermedad entre otros
Flujo de eventos	Una vez seleccionada la enfermedad se le mostrará la información de la misma y el usuario tendrá la opción de regresar al listado de enfermedades.
Flujo básico	El usuario ingresara a la aplicación móvil, Dirigirse al apartado de cultivos, observar el listado de cultivos ya registrados, seleccionar un cultivo, ver el listado de enfermedades y seleccionar una enfermedad
Frecuencia de uso	Uso diario
Include	Ninguno
Extend	Ninguno

Identificación del caso de uso	
ID	8
Nombre	Detección de la enfermedad
Actores	APP
Descripción breve	Este caso de uso muestra una pantalla donde podemos subir la imagen que se quiere analizar para recibir un diagnóstico del nivel de coincidencia de la imagen de la planta con la enfermedad a la que se le asocia
Precondiciones	Ingresar a la aplicación móvil y elegir la casilla de diagnosticar
Postcondiciones	En la pantalla debe aparecer un ícono con una cámara para al momento de seleccionar tomar una foto y que esta sea analizada
Flujo de eventos	Una vez ingresado a la detección buscará acceder a la cámara del teléfono móvil

Flujo básico	Ingresar al sistema, Ingresar a detección de la enfermedad
Frecuencia de uso	Uso diario
Include	Ninguno
Extend	Registrarse en el sistema

Identificación del caso de uso	
ID	9
Nombre	Solicita permiso para uso de cámara
Actores	App
Descripción breve	Para hacer la detección se requiere de el uso de la cámara para poder tomar las fotos que serán analizadas por la IA
Precondiciones	Se necesita haber ingresado al sistema y al apartado de detección de enfermedad
Postcondiciones	En la pantalla aparecerá un mensaje tipo emergente solicitando al usuario permiso para acceder a la cámara de su teléfono celular donde tendrá dos botones uno que confirme el uso de la cámara y otro que sea para rechazar la petición
Flujo de eventos	Una vez confirmado el acceso de la cámara se redirigirá a la cámara del teléfono móvil
Flujo básico	Ingresar a la APP, Ingresar a Detección de enfermedad, ingresar al logo de la cámara, Enviar mensaje emergente solicitando permiso y Acceder a la cámara

Frecuencia de uso	Uso diario
Include	Ninguno
Extend	Registrarse en el sistema

Identificación del caso de uso	
ID	10
Nombre	Ingresar a la pantalla para obtener diagnóstico
Actores	Usuario
Descripción breve	Ingresar a al apartado de diagnóstico de enfermedad allí se tomará el registro fotográfico y se enviará para un análisis
Precondiciones	Ingresar a la aplicación móvil y buscar la casilla diagnóstica de enfermedad
Postcondiciones	En la pantalla contiene el icono de una cámara en el cual al momento de hacer clic nos dirigirá a cámara del teléfono móvil
Flujo de eventos	Una vez ingresado al apartado de obtener diagnóstico y podemos tomar la foto y enviarla para un análisis
Flujo básico	Ingresar al APP, seleccionar en la casilla de diagnóstico de enfermedad, ingresar pantalla de diagnóstico de enfermedad
Frecuencia de uso	Uso diario
Include	Ninguno
Extend	Ninguna

Identificación del caso de uso	
ID	11

Nombre	Permitir uso de la cámara del teléfono móvil
Actores	Usuario
Descripción breve	para poder hacer uso de la cámara del dispositivo móvil se debe pedir permiso primero al usuario por lo tanto a este le aparecerá una casilla emergente con la petición de acceso a la cámara de su dispositivo
Precondiciones	ingresar al apartado de diagnóstico de enfermedad y hacer clic en el icono de la cámara
Postcondiciones	En la pantalla aparecerá un texto con la solicitud de acceso a la cámara del dispositivo con dos botones en uno de ellos se permitirá el acceso en el otro se rechazará la petición
Flujo de eventos	Una vez confirmada la petición de acceso a la cámara se redirigirá al usuario a la aplicación de su cámara y luego se le dará la opción de enviar la imagen tomada
Flujo básico	Ingresar a la app, ingresar a diagnóstico de enfermedad, Hacer clic en el botón de la cámara, Enviar ventana emergente con solicitud
Frecuencia de uso	Uso diario
Include	Ninguno
Extend	Ninguno

Identificación del caso de uso	
ID	12
Nombre	Enviar foto para diagnóstico
Actores	Usuario
Descripción breve	Tomar una foto de la mazorca del cacao esta será procesada por el API para dar un diagnóstico

Precondiciones	ingresar al apartado de diagnóstico de enfermedad y hacer clic en el icono de la cámara y tomar la foto y luego fijar clic en el icono de enviar
Poscondiciones	Cuando se de clic en enviar aparecerá un mensaje emergente al usuario donde se le dará aviso de que el proceso se está llevando a cabo y de que puede tardar un momento
Flujo de eventos	Una vez se hace clic se hace la captura de la imagen se envía a análisis con la IA
Flujo básico	Ingresar a la app, ingresar a diagnóstico de enfermedad, Hacer clic en el botón de la cámara, Ingresar a la aplicación de cámara del dispositivo, tomar la foto, enviar foto.
Frecuencia de uso	Uso diario
Include	Ninguno
Extend	Ninguno

Identificación del caso de uso	
ID	13
Nombre	Obtener foto
Actores	App
Descripción breve	El usuario hace una captura de una imagen con la cámara de su dispositivo móvil y la envía para un análisis la aplicación la atrapa y la envía a la api para que sea esta la encargada de compartirla con la IA
Precondiciones	El usuario debe haber tomado una foto de buena calidad y haber hecho clic en enviar
Poscondiciones	Una vez se recibe la imagen en archivo jpg el la aplicación móvil la captura y la envía a la API
Flujo de eventos	Una vez se hace clic en el icono de la cámara se redirige al usuario a la aplicación de la cámara de su dispositivo y desde ahí tomar la foto una vez se tiene la foto se redirigirá nuevamente al usuario para a la aplicación para que le de clic en enviar
Flujo básico	Ingresar a la app, ingresar a diagnóstico de enfermedad, Hacer clic en el botón de la cámara, Ingresar a la aplicación de cámara del dispositivo, tomar la foto, enviar foto, capturar foto

Frecuencia de uso	Uso diario
Include	Ninguno
Extend	Ninguno

Identificación del caso de uso	
ID	14
Nombre	Guardar imagen
Actores	App
Descripción breve	Se guardará la imagen en la base de datos interna del celular y de ahí se subirá al API para que este se la envíe a la IA y así obtener un diagnóstico
Precondiciones	Haber capturado la imagen enviada por el usuario
Postcondiciones	Después de que la imagen haya sido guardada y analizada la aplicación mostrará un diagnóstico en forma de mensaje
Flujo de eventos	Teniendo la imagen se envía a la base de datos interna del dispositivo móvil y ahí se guarda y se verifica si tiene conexión a internet de ser así se envía al API de lo contrario se quedará ahí almacenada hasta que se tenga la conexión necesaria
Flujo básico	Ingresar a la aplicación de cámara del dispositivo, tomar la foto, enviar foto, capturar foto, guardar foto
Frecuencia de uso	Uso diario
Include	Ninguno
Extend	Ninguno

Identificación del caso de uso	
ID	11
Nombre	Enviar imagen para diagnóstico
Actores	App
Descripción breve	La imagen guardada en la base de datos interna del celular pasará cuando el dispositivo cuente con la conexión a internet necesaria para que la IA haga el

	análisis respectivo de no ser así las imágenes enviadas para analizar se quedarán guardadas ahí en la base de datos del dispositivo y no serán liberadas hasta que tenga conexión
Precondiciones	El la imagen debe estar en buena calidad y debe haber conexión a internet para que la imagen pueda ser enviada para un diagnóstico
Postcondiciones	Después de enviar la imagen para diagnóstico se recibirá la respuesta del Api con el diagnóstico en forma de porcentaje
Flujo de eventos	Ingresar a la base de datos interna del dispositivo móvil, confirmar de que existe una conexión al servicio de internet, comunicarse con el api, después el api se comunica con la IA y esta nos arroja un resultado del análisis.
Flujo básico	Capturar foto, enviar a la base de datos del dispositivo móvil, verificar la conexión, recibir confirmación de envío
Frecuencia de uso	Uso diario
Include	Ninguno
Extend	Ninguno

Identificación del caso de uso	
ID	11
Nombre	Diagnóstico de la enfermedad
Actores	App
Descripción breve	la imagen enviada por el API a la IA es analizada y comparada con las imágenes del Dataset que se tiene almacenado en la IA, como resultado se arroja un diagnóstico en modo de porcentaje
Precondiciones	El la imagen debe estar en buena calidad y debe haber conexión a internet para que la imagen pueda ser enviada para un diagnóstico
Postcondiciones	Después de enviar la imagen para diagnóstico se recibirá la respuesta del Api con el diagnóstico en forma de porcentaje y se mostrará al usuario el porcentaje
Flujo de eventos	confirmar que existe una conexión al servicio de internet comunicarse con el api, después el api se comunica con la IA y esta nos arroja un resultado del análisis en forma de porcentaje
Flujo básico	verificar la conexión, recibir confirmación de envío, recibir diagnóstico

Frecuencia de uso	Uso diario
Include	Ninguno
Extend	Ninguno

Identificación del caso de uso	
ID	11
Nombre	Redirigir al apartado de Soluciones
Actores	App
Descripción breve	Después de tener el diagnóstico se redirigirá al usuario al apartado de soluciones donde se mostrará la solución más acertada para dicho diagnóstico
Precondiciones	Se mostrará una pantalla con la solución que se cree más apropiada para la etapa de la enfermedad, se mostrará información de los tratamientos, producto a utilizar y cómo evitar su propagación e incidencia
Postcondiciones	Después el usuario será redirigido a la pantalla de inicio nuevamente
Flujo de eventos	Obtener el diagnóstico de la IA en modo de porcentaje con el que se le hace un análisis para determinar cuál es el tratamiento más adecuado para la etapa en la que se encuentra la enfermedad y después de saber cual es se le mostrará al usuario en el apartado de soluciones al cual es redirigido.
Flujo básico	obtener diagnóstico, Analizar mejor tratamiento de acuerdo a las condiciones de la planta, Redirigir al usuario apartado de soluciones
Frecuencia de uso	Uso diario
Include	Ninguno
Extend	Ninguno

Identificación del caso de uso	
ID	11
Nombre	Ingresar a solución asignada corregir

Actores	Usuario
Descripción breve	El usuario va ser dirigido al apartado de soluciones a una solución asignada de acuerdo con el análisis hecho al diagnóstico de la enfermedad
Precondiciones	Se mostrará una pantalla con la solución que se cree más apropiada para la etapa de la enfermedad, se mostrará información de los tratamientos, producto a utilizar y cómo evitar su propagación e incidencia
Postcondiciones	Después el usuario será redirigido a la pantalla de inicio nuevamente
Flujo de eventos	Obtener el diagnóstico de la IA en modo de porcentaje con el que se le hace un análisis para determinar cuál es el tratamiento más adecuado para la etapa en la que se encuentra la enfermedad y después de saber cuales se le mostrará al usuario en el apartado de soluciones al cual es redirigido.
Flujo básico	obtener diagnóstico, Analizar mejor tratamiento de acuerdo a las condiciones de la planta, Redirigir al usuario apartado de soluciones
Frecuencia de uso	Uso diario
Include	Ninguno
Extend	Ninguno

Identificación del caso de uso	
ID	11
Nombre	Ingresar listado de discusiones
Actores	usuario
Descripción breve	El usuario tendrá la opción de ingresar a un apartado de discusión filtrado por cultivo y enfermedad en el que podrá hacer preguntas y a su vez responder
Precondiciones	Ingresar a la aplicación móvil y seguidamente buscar la sección de foros y buscar un cultivo y la enfermedad ahí podrá ingresar sus preguntas
Postcondiciones	se mostrará una pantalla como una especie de chat o foro donde los usuarios pueden responder preguntas y hacer preguntas, el usuario hará su pregunta y tendrá la opción de dar enviar y si no se siente conforme con lo escrito tendrá la opción de eliminar
Flujo de eventos	ingresar al apartado de discusiones

Flujo básico	ingresar a la aplicación móvil, seleccionar casilla de discusiones
Frecuencia de uso	Uso diario
Include	Ninguno
Extend	Ninguno

Identificación del caso de uso	
ID	11
Nombre	Seleccionar cultivo
Actores	usuario
Descripción breve	El usuario tendrá la opción de ingresar a un apartado de discusión filtrado por cultivo ya que en el foro se tratarán solo temas relacionados a dicho cultivo
Precondiciones	Ingresar a la aplicación móvil y seguidamente buscar la sección de foros y buscar un cultivo del cual desea opinar
Postcondiciones	se mostrará una pantalla como una especie de chat o foro donde los usuarios pueden responder preguntas y hacer preguntas, el usuario hará su pregunta y tendrá la opción de dar enviar y si no se siente conforme con lo escrito tendrá la opción de eliminar
Flujo de eventos	ingresar al apartado de discusiones
Flujo básico	ingresar a la aplicación móvil, seleccionar casilla de discusiones, seleccionar cultivo
Frecuencia de uso	Uso diario
Include	Ninguno
Extend	Ninguno

Identificación del caso de uso

ID	11
Nombre	Seleccionar enfermedad
Actores	usuario
Descripción breve	El usuario tendrá la opción de ingresar a un apartado de discusión filtrado por cultivo y enfermedad en el que podrá hacer preguntas y a su vez responder
Precondiciones	Ingresar a la aplicación móvil y seguidamente buscar la sección de foros y buscar un cultivo y la enfermedad ahí podrá ingresar sus preguntas
Poscondiciones	se mostrará una pantalla como una especie de chat o foro donde los usuarios pueden responder preguntas y hacer preguntas, el usuario hará su pregunta y tendrá la opción de dar enviar y si no se siente conforme con lo escrito tendrá la opción de eliminar
Flujo de eventos	ingresar al apartado de discusiones
Flujo básico	ingresar a la aplicación móvil, seleccionar casilla de discusiones, seleccionar cultivo, seleccionar enfermedad de cultivo
Frecuencia de uso	Uso diario
Include	Ninguno
Extend	Ninguno

Anexo D Documento de arquitectura

ARQUITECTURA DEL SISTEMA

PROYECTO

IACULTIVAR: PROTOTIPO DE APLICACIÓN MÓVIL PARA LA DETECCIÓN DE LA MONILIA EN LOS CULTIVOS DE CACAO MEDIANTE RECONOCIMIENTO DE IMÁGENES

PONENTES

ANDRES CAMILO RINCON CABALLERO - U00111082

MARYURY JULIETH RODRIGUEZ CALDERON - U00111229

DIRECTOR

JUAN SEBASTIÁN CÁRDENAS ARENAS

UNIVERSIDAD AUTÓNOMA DE BUCARAMANGA

PROYECTO DE GRADO

BUCARAMANGA

202

INTRODUCCIÓN

El presente documento muestra evidencia de los componentes y variedad de sistemas qué se van a usar para el desarrollo del proyecto “IAcultivar: prototipo de aplicación móvil para la detección de la monilia en los cultivos de cacao mediante reconocimiento de imágenes” esto con el fin de documentar cada uno de los componentes del sistema y ver cómo se relacionan entre sí.

El sistema estará compuesto primeramente de una aplicación móvil la cual extraerá los datos de una API REST el cual se encarga de gestionar todos los datos qué se suministraran a la aplicación, este API REST igualmente se relaciona directamente con el administrador web, este admin se encargará de la visualización, agregación y edición de información a la base datos, siendo esta información datos de los cultivos, enfermedades qué afectan a los cultivos, soluciones, controles, entre otros, cabe destacar qué el admin web es el único qué tiene acceso directo a la base de datos, la base de datos será relacional por el hecho de la facilidad de organizar la información.

Para cubrir la necesidad del reconocimiento de imágenes se usará un servicio tercerizado en la nube el cual será alimentado y probado por el administrador , por último la aplicación se comunica únicamente con el API REST está proveerá información al usuario final el cual disfrutara del producto mencionado anteriormente.

DESCRIPCIÓN GENERAL

Posicionamiento y alcance:

Este proyecto tiene como propósito el desarrollo de una aplicación móvil que integre reconocimiento de imágenes con herramientas como custom visión o teachable machines para proporcionar un diagnóstico de la monilia qué es una enfermedad presente en el cultivo de cacao, dicho diagnóstico se representará como un porcentaje que indicará la presencia de la enfermedad, para poder visualizar este diagnóstico se deberá contar con acceso a internet y a partir de este dato se aconsejara al agricultor sobre los tratamientos adecuados para prevenir y evitar la propagación de la moniliasis.

El alcance del proyecto no abarca solucionar el problema del agricultor sino dar una herramienta la cual ayude a qué su toma de decisiones frente a su cultivo sea óptima, esto proveyendo al mismo de información la cual estará a su disposición.

La información anteriormente hablada comprende conocimientos sobre el cultivo a tratar siendo cacao el cultivo objetivo, las enfermedades qué pueden afectar el cultivo, consejos desde diferentes vectores de soluciones siendo estos, controles culturales, biológicos, entre otros.

Objetivos del diseño

- Evidenciar el software qué se usará para la creación del sistema
- Documentar los procesos de desarrollo del proyecto.
- Dar una perspectiva del sistema en general

Características principales

1. Aplicación móvil

La aplicación móvil será desarrollada en el framework Flutter qué hace uso del lenguaje de programación Dart, a su vez se implementará el patrón bloc para el desarrollo de la aplicación en general. bloc siendo un sistema de gestión de estado para Flutter recomendado por los desarrolladores de Google. Ayuda en la gestión del estado y en el acceso a los datos desde un lugar central del proyecto.



Fig. 1 Flutter y dart.

2. Admin web y API:

El admin web se desarrollara en Django qué hace uso de Python como lenguaje de programación esto con el objetivo de facilitar la creación del admin web ya que este framework nos brinda rapidez a la hora de trabajar con CRUD's aparte de qué su integración con las bases de datos y sus interfaces preestablecidas ayudan a trabajar de manera eficiente.



Fig. 2 Python y django.

3. Base de datos

El servicio de base de datos MySQL y MySQLLite son servicios de bases de datos totalmente administrados para implementar aplicaciones nativas en la nube, aparte de qué son Open source lo cual las herramientas excelentes para cualquier desarrollador junior, al igual la documentación al respecto es completa lo cual ayuda al desarrollador a tener un sitio al que acceder cuando hay problemas.



Fig. 3 MySQL.

4. Módulo de reconocimiento de imágenes

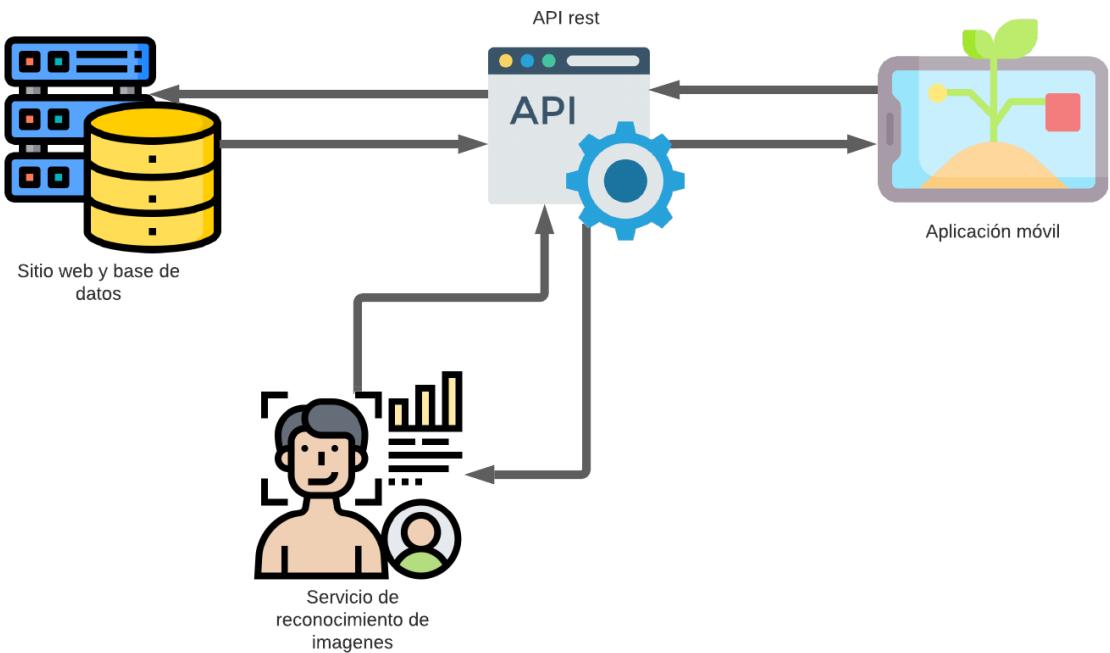
Para el reconocimiento de imágenes se usará un servicio tercerizado llamado Teachable Machine, una herramienta basada en la Web que hace posible crear modelos de aprendizaje automático de manera rápida, sencilla, accesible para todos y sin necesidad de programar.

Teachable Machine

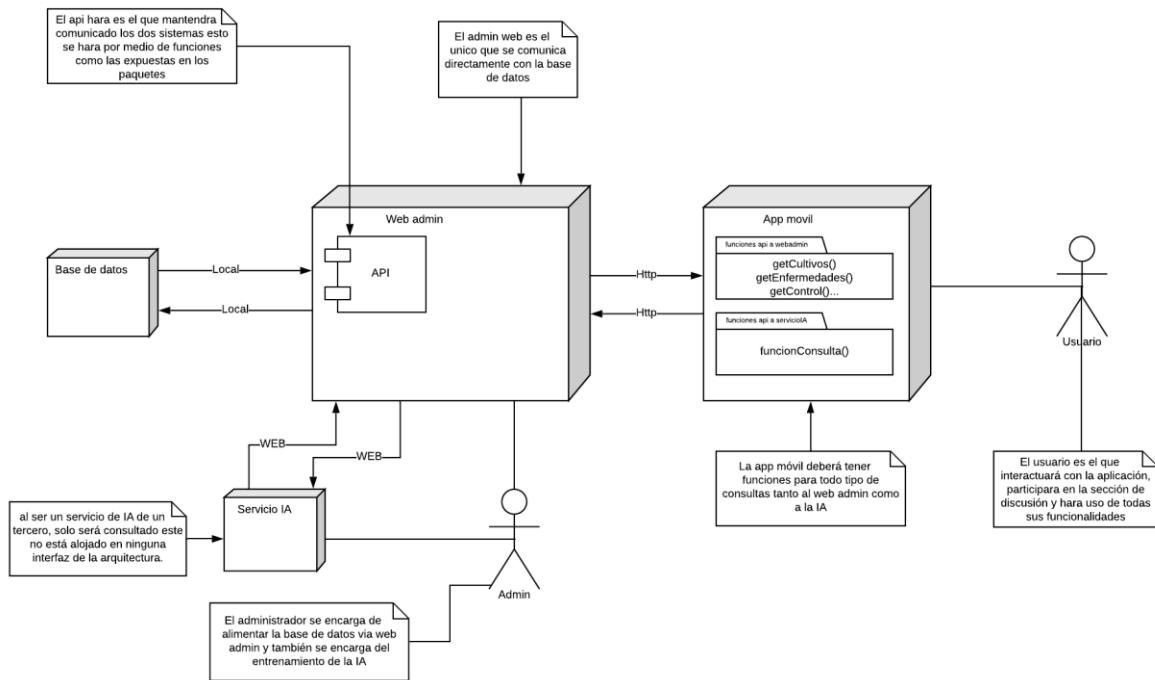
Fig. 4 Teachable machine

VISTAS Y PLANOS

Vista general



Vista de despliegue



DETALLES DE LA IMPLEMENTACIÓN

Plataformas, lenguajes y organización del proyecto

1. Desarrollo API REST y web admin
 - Plataformas: Django
 - Lenguajes: Python
2. Entrenamiento del módulo de reconocimiento de imágenes:
 - Plataformas: Teachable machine
3. Fase de desarrollo de aplicación móvil:
 - Plataformas: Flutter
 - Lenguajes: Dart
4. Fase de creación de integración:
 - Plataformas: Amazon web services

La organización del proyecto estará regida por el respaldo de GitHub, para el manejo de versiones en cuanto a la creación de software, en el caso del API y la aplicación móvil.

REFERENCIAS

Django Software Foundation. (2020). Why django? Obtenido de <https://www.djangoproject.com/start/overview/>

Knowlton, & Jim. (2008). *Python* (1st ed.)

MuleSoft, L. (2020). What is a REST API. Obtenido de <https://www.mulesoft.com/resources/api/what-is-rest-api-design>

Oracle. (2020). What is a database. Obtenido de <https://www.oracle.com/database/what-is-database.html>

Google. (2020b). Dart. Obtenido de <https://dart.dev/>

Google. (2020c). Teachable machine. Obtenido de <https://teachablemachine.withgoogle.com/>

Anexo E Interfaces



<p>Descripción</p>  <p>← cacao</p> <div style="display: flex; justify-content: space-around; margin-top: 10px;">  Detección en vivo  Tratamientos  Enfermedades </div> <p>Preguntas acerca del cultivo</p> <p>¿Qué es el cacao?</p> <p>El cacao es un árbol procedente de América que produce un fruto del mismo nombre que se puede utilizar como ingrediente para alimentos entre los que destaca el chocolate. Su uso se remonta a la época de los mayas, aztecas e incas, y desde entonces se ha usado tanto para fines nutricionales como médicos.</p> 	<p>Enfermedades</p>  <p>← Monilia</p> <p>"Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum."</p> <p>Mazorca</p>  <p>"Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat."</p> <p>Mazorca</p>  <p>"Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat."</p> <p>Preguntas acerca de la enfermedad</p>
---	--

Sugerencias principal

[←](#)

Problema con los listados

Formulario de pregunta

[←](#)

Cacao



Nombre: pablito
Cargo: agricultor

Pregunta: ¿porque el cacao..

Descripcion de pregunta

Enviar Pregunta 

Tratamientos

[←](#)



Tratamiento

"Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum."

Control Cultural




"Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat."

Control Cultural




"Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat."

Nueva sugerencia

Pop up registro

Para entrar al foro tiene que iniciar sesión o registrarse

Iniciar sesión

cancelar

Listado de foros

[←](#)

Cacao	
Monilia	
Escoba de bruja	
Tomate	
Monilia	
Mora	
Monilia	
Monilia	

Pop up

¿Estas seguro de realizar esta acción?

No

Sí

Foro general de cada enfermedad

Detección en vivo

Diagnóstico

Cacao (Monilia)

Pregunta: ¿porque el cacao...

Responder

Pregunta: ¿porque el cacao...

Responder

Pregunta: ¿porque el cacao...

Responder

Hacer pregunta



Enviar para detección



95% de coincidencia con monilia

Tratamientos

Tratamiento para imagen

control Biológico

control cultural

control cultural

95% de coincidencia con monilia



Opciones de tratamiento
Segun el porcentaje de ___ te recomendamos los siguientes tratamientos

Control Biológico



Segun la etapa en la que se encuentra la enfermedad se recomienda usar el metodo biológico ya que los insectos como la ayuda a la reducion de las esporas de la minilisia

Pasos

Control cultural



Segun la etapa en la que se encuentra la enfermedad se recomienda usar el metodo cultural especificamente la poda porque la enfermedad ya esta muy abansada

Pasos

Control químico



Las prácticas culturales y de manejo del cultivo permiten la manipulación del medio ambiente para generar condiciones menos favorables al desarrollo de la enfermedad.

1. como primer paso debemos tomar la mazorza infectada y colocarla en la población de insectos

2. como segundo paso debemos tomar la mazorza infectada y colocarla en la población de insectos

1. como primer paso debemos tomar la mazorza infectada y colocarla en la población de insectos

2. como segundo paso debemos tomar la mazorza infectada y colocarla en la población de insectos

Log in foro



Cuenta

Contraseña

[¿No tienes una cuenta? Regístrate](#)

Formulario de registro foro



Usuario

Contraseña

Contraseña de nuevo

Cargo

Formulario de sugerencias



Titulo sugerencia

Label
This is a very long input that extends beyond the height of the text area which is masked at the top when

Android - 1

Cacao

Pregunta: ¿porque el cacao..

Descripcion Pregunta

Teresa Martines
Agronomia

Referente a tu pregunta te puedo aconsejar que

**Yo
Agricultor**

complementando la pregunta de
@Teresa creo que

detalles pregunta

Cacao (Monilia)

Pregunta: ¿porque el cacao..

Descripcion Pregunta

Teresa Martines
Agronomia

Referente a tu pregunta te puedo aconsejar que

**Yo
Agricultor**

complementando la pregunta de
@Teresa creo que

foro cacao

Cacao

Pregunta: ¿porque el cacao..

Responder

Pregunta: ¿porque el cacao..

Responder

Pregunta: ¿porque el cacao..

Responder

Hacer pregunta

Anexo F Pruebas

ID caso de prueba	01
Nombre caso de prueba	Ver listado de enfermedades
Descripción	Se probará la respuesta de la aplicación cuando el usuario de clic en el apartado de enfermedades
Precondiciones	El usuario deberá tener instalada la aplicación en su dispositivo móvil Android, y haber ingresado a ella
Relación de Caso de Uso	El caso de prueba está relacionado con el caso de uso “Ingresar al listado de enfermedades”
Pasos y condiciones de ejecución	El usuario ingresa a la aplicación dando clic en el botón de enfermedades y la aplicación debe redirigirse a una pantalla donde se muestran el listado de todas las enfermedades registradas desde el admin web y cargados en la base de datos
Resultado esperado	Se espera que se muestre el listado de cada una de las enfermedades registradas con una pequeña descripción de la misma
Estado de caso de prueba	Ejecutado
Resultado obtenido	Al llevar acabo la ejecución se vio que la aplicación realiza el caso de prueba de manera correcta ya que el usuario es dirigido a una pantalla donde se muestra el listado de las enfermedades registradas
Errores asociados	Ninguno
Responsable de diseño caso de prueba	Equipo de trabajo
Responsable de ejecución caso de prueba	Usuario
Comentarios	No

ID caso de prueba	02
Nombre caso de prueba	Seleccionar enfermedad
Descripción	Se probará que la aplicación permita que el usuario seleccione una de las enfermedades registradas
Precondiciones	El usuario deberá estar en la pantalla del listado de enfermedades
Relación de Caso de Uso	El caso de prueba está relacionado con el caso de uso “Seleccionar enfermedad”

Pasos y condiciones de ejecución	El listado de enfermedades deberá permitir ser cliqueable para poder seleccionar una enfermedad previamente registrada en el admin web
Resultado esperado	Se espera que la aplicación permita que una de las enfermedades registradas pueda ser seleccionadas con el fin de obtener más información de la misma
Estado de caso de prueba	Ejecutado
Resultado obtenido	Al llevar acabo la ejecución se vio que la aplicación realiza el caso de prueba de manera correcta ya que el usuario puede seleccionar de manera correcta la enfermedad
Errores asociados	Ninguno
Responsable de diseño caso de prueba	Equipo de trabajo
Responsable de ejecución caso de prueba	Usuario
Comentarios	No

ID caso de prueba	03
Nombre caso de prueba	Información de enfermedad
Descripción	Se probará la respuesta de la aplicación al cuando se de clic en el nombre de la enfermedad
Precondiciones	Estar en la pantalla de listado de enfermedades y haber seleccionado una de las enfermedades registradas
Relación de Caso de Uso	El caso de prueba está relacionado con el caso de uso “Mostrar información de la enfermedad”
Pasos y condiciones de ejecución	Se deberá dar clic a la enfermedad de la que se requiere mayor información y la aplicación deberá redirigir a una pantalla que contiene información cargada de la base de datos, con datos como partes de la planta que afecta esta enfermedad y una descripción general de lo que es la enfermedad y su sintomatología
Resultado esperado	Se espera que la aplicación redirija al usuario a una pantalla que contenga toda la información, imágenes ilustrativas de la enfermedad
Estado de caso de prueba	Ejecutado
Resultado obtenido	Al llevar acabo la ejecución se vio que la aplicación realiza el caso de prueba de manera correcta ya que el usuario es dirigido a una

	pantalla donde se carga de manera correcta la información de la enfermedad y las imágenes que describen la enfermedad
Errores asociados	Ninguno
Responsable de diseño caso de prueba	Equipo de trabajo
Responsable de ejecución caso de prueba	Usuario
Comentarios	No

ID caso de prueba	04
Nombre caso de prueba	Foro
Descripción	Se probará la respuesta de la aplicación al cuando el usuario desee ingresar al listado de discusiones
Precondiciones	Tener instalada la aplicación en su dispositivo Android, estar en la pantalla principal de la aplicación
Relación de Caso de Uso	El caso de prueba está relacionado con el caso de uso “Ingresar listado de discusiones”
Pasos y condiciones de ejecución	Los pasos a seguir para llevar acabo este caso de prueba el usuario debe ingresar al menú principal de la aplicación y seleccionar el botón de discusiones el cual lo redirige a un listado de foros que esta dividido por cultivos y enfermedades asociadas al mismo
Resultado esperado	Con la ejecución de este caso de prueba se quiere ver como se comporta la aplicación cuando el usuario haga la petición de ingresar al listado de discusiones dando clic en el botón el usuario deberá ser dirigido a una pantalla con todos los foros de cultivos registrados en el admin y traídos a la aplicación mediante el API
Estado de caso de prueba	Ejecutado
Resultado obtenido	Se evidencio el correcto funcionamiento de la aplicación cuando se hace la petición de mostrar el listado de discusiones ya que esta muestra los foros categorizados por cultivos y enfermedades del mismo
Errores asociados	Ninguno
Responsable de diseño caso de prueba	Equipo de trabajo
Responsable de ejecución caso de prueba	Usuario

Comentarios	No
--------------------	----

ID caso de prueba	05
Nombre caso de prueba	Seleccionar foro de cultivo
Descripción	Se probará la respuesta de la aplicación al cuando el usuario desee seleccionar el foro de un cultivo específico
Precondiciones	El usuario debe estar en el listado de cultivos de discusiones para poder seleccionar un tema en el que quiera profundizar
Relación de Caso de Uso	El caso de prueba está relacionado con el caso de uso “Seleccionar cultivo”
Pasos y condiciones de ejecución	Ingresar al foro de discusiones y buscar el cultivo del cual quiere ver lo que los demás usuarios comentan, estos datos son de cultivos que ya han sido registrados y se traen por medio del API y se muestran en modo de lista para que se puedan seleccionar.
Resultado esperado	Se desea cargar la lista de todos los cultivos registrados y seleccionar uno de ellos para posteriormente poder visualizar toda la discusión que han tenido otros usuarios y participar de la misma
Estado de caso de prueba	Ejecutado
Resultado obtenido	Al realizar esta prueba se vio que la aplicación trae de manera correcta los datos como el listado de los cultivos registrados y permite seleccionar uno de ellos para ser explorado con mayor profundidad en un foro y participar del mismo
Errores asociados	Ninguno
Responsable de diseño caso de prueba	Equipo de trabajo
Responsable de ejecución caso de prueba	Usuario
Comentarios	No

ID caso de prueba	06
Nombre caso de prueba	Registro
Descripción	Se quiere probar al momento que un usuario quiera hacer una pregunta o responder una de las que ya están ahí plasmadas se le informa y redirija a un formulario de registro

Precondiciones	Para llegar al registro se debe seleccionar un el botón de hacer pregunta que se encuentra en la pantalla de discusiones o si se desea responder alguna pregunta igualmente debe ser seleccionado el botón de responder en los dos casos el usuario será dirigido a la pantalla que contiene el formulario de registro
Relación de Caso de Uso	El caso de prueba está relacionado con el caso de uso “Registrarse”
Pasos y condiciones de ejecución	Se debe seleccionar el botón de hacer una pregunta o el de responder los cuales se encuentran en la pantalla de foro al seleccionarlos estos redirijeran a la pantalla de registro donde se encuentra un formulario en blanco donde se debe llenar información como nombre de usuario, contraseña, cargo o profesión,
Resultado esperado	Se desea cargar una pantalla con cada uno de los comentarios hechos por los usuarios y sus respuestas todo esto a modo de foro
Estado de caso de prueba	Ejecutado
Resultado obtenido	Al realizar esta prueba se vio que la aplicación trae de manera correcta los datos como las preguntas y respuestas de manera rápida.
Errores asociados	Ninguno
Responsable de diseño caso de prueba	Equipo de trabajo
Responsable de ejecución caso de prueba	Usuario
Comentarios	No

ID caso de prueba	07
Nombre caso de prueba	Realizar pregunta
Descripción	Se quiere probar que la aplicación permita al usuario hacer preguntas, las almacena y las muestra a los demás usuarios
Precondiciones	El usuario debe estar previamente registrado con nombre de usuario, contraseña y cargo y debe haber dado clic al botón de hacer pregunta
Relación de Caso de Uso	El caso de prueba está relacionado con el caso de uso “Realizar pregunta”
Pasos y condiciones de ejecución	Para probar este caso el usuario debe estar en el apartado de foro para hacer la pregunta se debe dar clic en el botón el cual lo redirigirá a una

	pantalla donde aparecen los datos de registro y una casilla donde se pide al usuario que ingrese el texto de la pregunta que desea hacer y posteriormente podrá dar clic en el botón de enviar, esa información se sube a la base de datos la cual posteriormente se comunica con el API y esta con la aplicación para mostrar la información en la pantalla de discusiones.
Resultado esperado	Se espera que la aplicación suba y muestre la información de forma correcta que la comunicación de la aplicación con el API y base de datos sea correcta que el usuario pueda hacer la pregunta y esta a su vez se muestre en el foro
Estado de caso de prueba	Ejecutado
Resultado obtenido	Al realizar esta prueba se vio que la aplicación trae de manera correcta los datos como las pregunta y la almacena en la base de datos.
Errores asociados	Ninguno
Responsable de diseño caso de prueba	Equipo de trabajo
Responsable de ejecución caso de prueba	Usuario
Comentarios	No

ID caso de prueba	08
Nombre caso de prueba	Hacer sugerencia
Descripción	Se quiere probar que la aplicación permita al usuario hacer sugerencias, permita almacenarla y que los demás usuarios la visualicen
Precondiciones	Para hacer una sugerencia se debe ingresar por el botón de sugerencias que se encuentra en el menú principal de la aplicación
Relación de Caso de Uso	El caso de prueba está relacionado con el caso de uso “Hacer sugerencia”
Pasos y condiciones de ejecución	Para hacer la sugerencia el usuario debe ingresar a la pantalla de sugerencias y dar clic en el botón flotante que parece en la misma, este botón lo redirigirá a una pantalla donde esta el formulario de nueva sugerencia el cual contiene una casilla donde se pide que proporcione un texto con la sugerencia respecto al funcionamiento de la aplicación

Resultado esperado	Se espera que la aplicación suba y muestre la información de forma correcta que la comunicación de la aplicación con el API y base de datos sea correcta que el usuario pueda hacer la sugerencia y está a su vez se muestre en el listado de sugerencias
Estado de caso de prueba	Ejecutado
Resultado obtenido	Al realizar esta prueba se vio que la aplicación trae de manera correcta los datos como las sugerencias y la almacena en la base de datos y la muestra en el listado
Errores asociados	Ninguno
Responsable de diseño caso de prueba	Equipo de trabajo
Responsable de ejecución caso de prueba	Usuario
Comentarios	No

ID caso de prueba	09
Nombre caso de prueba	Diagnostico
Descripción	Ingresar desde el menú principal de la aplicación por medio del botón detección en vivo que lleva a la pantalla para tomar una foto que se deseé analizar
Precondiciones	Para hacer una sugerencia se debe ingresar por el botón de sugerencias que se encuentra en el menú principal de la aplicación.
Relación de Caso de Uso	El caso de prueba está relacionado con el caso de uso “Ingresar a pantalla para obtener diagnostico”
Pasos y condiciones de ejecución	Para ingresar a la pantalla de obtener un diagnóstico, el usuario debe ingresar al menú principal de la aplicación y hacer clic en el botón de detección en vivo y ésta lo dirige a solicitar permiso para utilizar la cámara de su equipo.
Resultado esperado	Se espera que la aplicación abra una pestaña donde solicite permiso para hacer uso de la cámara de su dispositivo móvil para lograr desde ahí captar las imágenes que posteriormente serán analizadas.
Estado de caso de prueba	Ejecutado
Resultado obtenido	Al realizar la prueba se evidenció que la aplicación realiza de manera correcta el direccionamiento desde el menú principal hasta la pantalla que

	muestra la cámara del dispositivo y solicita permiso para utilizar la cámara el cual al ser otorgado por el usuario se puede hacer la captura de la foto
Errores asociados	Ninguno
Responsable de diseño caso de prueba	Equipo de trabajo
Responsable de ejecución caso de prueba	Usuario
Comentarios	No

ID caso de prueba	10
Nombre caso de prueba	Imagen diagnostico
Descripción	Se evaluará que la aplicación envié de manera correcta las imágenes captadas para un diagnóstico.
Precondiciones	Para enviar la imagen a diagnóstico el usuario debe haber hecho captura de la misma desde la aplicación.
Relación de Caso de Uso	El caso de prueba está relacionado con el caso de uso “Enviar foto para diagnostico”
Pasos y condiciones de ejecución	Después de que el usuario tenga imagen que desea analizar la aplicación muestra en la pantalla un botón para enviar la imagen a un análisis por medio de la IA para esto la aplicación debe cargar la imagen al API rest y está la envía a la IA por medio de una petición http.
Resultado esperado	Se espera que la aplicación conecte de manera correcta con el API y la IA para que la imagen sea subida y guardada dentro de una carpeta del proyecto llamada media que es donde se va almacenar todas las imágenes que los usuarios envían para análisis
Estado de caso de prueba	Ejecutado
Resultado obtenido	Al realizar la prueba se evidencio que la aplicación realiza de manera correcta ya que al hacer la petición a la API esta devuelve un código que nos dice de confirmación diciendo que la acción que solicito fue exitosa
Errores asociados	Ninguno
Responsable de diseño caso de prueba	Equipo de trabajo

Responsable de ejecución caso de prueba	Usuario
Comentarios	No

ID caso de prueba	11
Nombre caso de prueba	Detección de enfermedad
Descripción	Se evaluará que la aplicación muestre el resultado del diagnóstico a modo de porcentaje que devuelve la IA.
Precondiciones	Para que esta prueba se lleve a cabo de manera correcta la imagen se debe encontrar en el modulo de inteligencia artificial, en el cual la imagen es analizada y se devuelve un diagnóstico de dicha enfermedad.
Relación de Caso de Uso	El caso de prueba está relacionado con el caso de uso "Detección de enfermedad"
Pasos y condiciones de ejecución	Ya que la imagen se encuentre en el modulo de la IA, esta debe devolver un diagnostico a modo de porcentaje de coincidencia de dicha enfermedad y este diagnóstico será enviado al API y esta pasa la información a la aplicación.
Resultado esperado	Se espera que la aplicación conecte de manera correcta con el API y la IA para que la imagen sea analizada por la inteligencia artificial muestre el diagnostico en la aplicación
Estado de caso de prueba	Ejecutado
Resultado obtenido	Se evidencia la correcta comunicación en el sistema y con esto se puede visualizar el diagnostico de la enfermedad a modo de porcentaje
Errores asociados	Ninguno
Responsable de diseño caso de prueba	Equipo de trabajo
Responsable de ejecución caso de prueba	Usuario
Comentarios	No

Nº. De prueba	Descripción de prueba	Entrada para probar	Resultado esperado	Resultado
1-1	Navegación por cada una de las interfaces	Pantallas	Entrada a cada pantalla y salida de la misma, pasar entre pantallas	
1-2	Enviar la información de formulario de registro a foro	Flujo de información entre el API, web admin, Base de datos	Recibir la información ingresada por el usuario, almacenarla en la base de datos y verificarla cada vez que intente ingresar	
1-3	Realizar preguntas y responder preguntas de un foro general y de una enfermedad específica	Flujo de información entre API, web admin y base de datos	Enviar las preguntas y respuestas hechas por los usuarios en el foro de cultivo y de una enfermedad de cultivo específico	
1-4	Enviar una foto para analizar el porcentaje de presencia de enfermedad	Conexión API, IA, App	Tomar la foto y enviarla para recibir el porcentaje de coincidencia de la imagen con la enfermedad y recibir el tratamiento de acuerdo con la etapa en la que se encuentre la enfermedad	
1-5	Tiempos respuesta peticiones realizadas desde la aplicación		Recibir la información de cada una de las peticiones y acciones en un tiempo prudente y asegurar que la aplicación no sea haga lenta	
1-6	Enviar la información de formulario de registro a foro		Información de registro recibida y almacenada en la base de datos de manera correcta	
1-7	Enviar la información de formulario de registro a foro		Información de registro recibida y almacenada en la base de datos de manera correcta	

Nº. De prueba	Descripción de prueba	Entrada para probar	Resultado esperado	Resultado
----------------------	------------------------------	----------------------------	---------------------------	------------------

1-1	Navegación por cada una de las interfaces	Pantallas	Entrada a cada pantalla y salida de la misma, pasar entre pantallas	Se logra evidenciar que la aplicación permite entrar y salir de forma fácil de todas las interfaces que esta contiene.
1-2	Enviar la información de formulario de registro a foro	Flujo de información entre el API, web admin, Base de datos	Recibir la información ingresada por el usuario, almacenarla en la base de datos y verificarla cada vez que intente ingresar	La aplicación muestra que todas las partes del sistema se encuentran conectadas de forma correcta ya que nos permiten hacer peticiones y recibir respuestas de la misma.
1-3	Realizar preguntas y responder preguntas de un foro general y de una enfermedad específica	Flujo de información entre API, web admin y base de datos	Enviar las preguntas y respuestas hechas por los usuarios en el foro de cultivo y de una enfermedad de cultivo específico	Se observa que la aplicación carga toda la información registrada por los demás usuarios y permite subir nueva, esto nos muestra que la conexión API base de datos es la correcta
1-4	Enviar una foto para analizar el porcentaje de presencia de enfermedad	Conexión API, IA, App	Tomar la foto y enviarla para recibir el porcentaje de coincidencia de la imagen con la enfermedad y recibir el tratamiento de acuerdo con la etapa en la que se encuentre la enfermedad	Se puede ver que la aplicación puede acceder a la cámara del dispositivo móvil, y captar la foto, enviarla al modulo de inteligencia artificial para el análisis, resultado que se muestra como porcentaje.
1-5	Tiempos de respuesta a peticiones realizadas desde la aplicación		Recibir la información de cada una de las peticiones y acciones en un tiempo prudente y asegurar que la aplicación no sea haga lenta	Se pudo evidenciar que la aplicación tiene un tiempo prudente en cada una de las peticiones hechas.

caso de prueba	Resultado	Fortalezas	Dificultades
Ver listado de enfermedades	Al llevar acabo la ejecución se vio que la aplicación realiza el caso de prueba de manera correcta ya que el usuario es dirigido a una pantalla donde se muestra el listado de las enfermedades registradas	Con la ejecución de este caso de prueba se logro evidenciar que la aplicación conecta de manera rápida y fácil al usuario y le permite visualizar todos los cultivos registrados	el usuario no sabía que el listado era cliqueable, y por esta razón no podía ingresar a la pantalla de detalle de enfermedades
Seleccionar enfermedades	Al llevar acabo la ejecución se vio que la aplicación realiza el caso de prueba de manera correcta ya que el usuario puede seleccionar de manera correcta la enfermedad	Muestra la información de forma rápida	Ninguna
Información de enfermedad	Al llevar acabo la ejecución se vio que la aplicación realiza el caso de prueba de manera correcta ya que el usuario es dirigido a una pantalla donde se carga de manera correcta la información de la enfermedad y las imágenes que	Muestra de manera rápida la información y la interfaz que lo contiene es entendible para le usuario.	Las imágenes tardaron un poco en mostrarse en pantalla.

	describen la enfermedad		
Foro	Se evidencio el correcto funcionamiento de la aplicación cuando se hace la petición de mostrar el listado de discusiones ya que esta muestra los foros categorizados por cultivos y enfermedades del mismo.	La aplicación móvil muestra de manera rápida la pantalla del foro y el usuario se pudo identificar que se trataba de un foro de discusiones	ninguna
Registro	Al realizar esta prueba se vio que la aplicación trae de manera correcta los datos como el listado de los cultivos registrados y permite seleccionar uno de ellos para ser explorado con mayor y profundidad en un foro y participar del mismo	El formulario es entendible para el usuario y los datos son almacenados de forma correcta	Ninguna

Realizar pregunta	Al realizar esta prueba se vio que la aplicación trae de manera correcta los datos como las preguntas y respuestas de manera rápida, se pudo realizar la pregunta	Al ser un módulo filtrado por cultivos y enfermedades permite que el usuario pueda participar más activamente en los foros	Cuando se hacen un número de peticiones alto tiende a no mostrar de forma rápida las preguntas o respuesta hechas por los usuarios
Hacer sugerencia	Al realizar esta prueba se vio que la aplicación trae de manera correcta los datos como las preguntas y la almacena en la base de datos	Al tener un botón flotante para el usuario fue rápido poder hacer una nueva sugerencia	Ninguna
Diagnóstico	Al realizar la prueba se evidenció que la aplicación realiza de manera correcta el direccionamiento desde el menú principal hasta la pantalla que muestra la cámara del dispositivo y solicita permiso para utilizar la cámara el cual al ser otorgado por el usuario se puede hacer la captura de la foto	El botón de detección en vivo es muy ilustrativo y el usuario sabe que ahí deberá tomar una foto que será analizada	Ninguna

Imagen Diagnóstico	<p>La aplicación conecta de manera correcta con el API y la IA para que la imagen sea subida y guardada dentro de una carpeta del proyecto llamada media que es donde se va almacenar todas las imágenes que los usuarios envían para análisis</p>	<p>La imagen se sube puede subir apenas es tomada</p>	<p>Cuando el dispositivo móvil no se encuentra con conexión a internet el usuario no podrá obtener el diagnóstico.</p>
Detección de enfermedad	<p>Se evidencia la correcta comunicación en el sistema y con esto se puede visualizar el diagnóstico de la enfermedad a modo de porcentaje</p>	<p>El diagnóstico es entendible para el usuario ya que sabe que cuando el numero es un numero cercano a 100 es porque tiene una mazorca muy enferma y si es cercano a 0 tiene una mazorca sana</p>	<p>Al solo estar trabajando con la monilia algunas mazorcas que tengan un síntoma similar se puede llegar a confundir.</p>