

UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI



FACULTAD DE INDUSTRIAS AGROPECUARIAS Y CIENCIAS AMBIENTALES

CARRERA DE COMPUTACIÓN

Tema: "Visión por computadora para la detección de enfermedades en cultivo de la *Pisum Sativum.L* (Arveja)"

Trabajo de Integración Curricular previo a la obtención del título de Ingeniero en Ciencias de la Computación

AUTOR: Castro Pérez Washington Patricio

TUTOR: Ing. Guano Cárdenas Carlitos Alberto, MSc.

Tulcán, 2026.

CERTIFICADO DEL TUTOR

Certifico que el estudiante Castro Pérez Washington Patricio, con el número de cédula 04020122264 respectivamente ha desarrollado el Trabajo de Integración Curricular: “Visión por computadora para la detección de enfermedades en cultivo de la *Pisum Sativum.L* (Arveja)”

Este trabajo se sujeta a las normas y metodología dispuesta en el Reglamento de la Unidad de Integración Curricular, Titulación e Incorporación de la UPEC, por lo tanto, autorizo la presentación de la sustentación para la calificación respectiva

Ing. Guano Cardenas Carlitos Alberto, MSc.

TUTOR

Tulcán, enero de 2026

AUTORÍA DE TRABAJO

El presente Trabajo de Integración Curricular constituye un requisito previo para la obtención del título de Ingeniero en la Carrera de computación de la Facultad de Industrias Agropecuarias y Ciencias Ambientales

Yo, Castro Pérez Washington Patricio con cédula de identidad número 0402012264 respectivamente declaro que la investigación es absolutamente original, auténtica, personal y los resultados y conclusiones a los que he llegado son de mi absoluta responsabilidad.



Castro Pérez Washington Patricio

AUTOR

Tulcán, enero de 2026

ACTA DE CESIÓN DE DERECHOS DEL TRABAJO DE INTEGRACIÓN CURRICULAR

Yo Castro Pérez Washington Patricio declaro ser autor de los criterios emitidos en el Trabajo de Integración Curricular: "Visión por computadora para la detección de enfermedades en cultivo de la *Pisum Sativum*.L (Arveja)" y eximo expresamente a la Universidad Politécnica Estatal del Carchi y a sus representantes de posibles reclamos o acciones legales.



Castro Pérez Washington Patricio

AUTOR

Tulcán, enero de 2026

AGRADECIMIENTO

Agradezco de manera especial a la Universidad Politécnica Estatal del Carchi, por brindarme la oportunidad de formarme académica y profesionalmente, así como por los conocimientos y valores impartidos a lo largo de mi carrera.

Expreso mi sincero agradecimiento a mi tutor de tesis, al MSc. Guano Cárdenas Carlitos Alberto, por su orientación, apoyo constante, paciencia y valiosos conocimientos, los cuales fueron fundamentales para el desarrollo y culminación de este trabajo de investigación.

Agradezco profundamente a mis padres y hermanos, quienes, con su apoyo incondicional, consejos y motivación permanente, fueron un pilar fundamental durante todo este proceso académico.

De igual manera, extiendo mi agradecimiento a mis amigos Yon Montesdeoca y Stiven Bustos, por formar parte de esta etapa de mi vida, brindándome su apoyo y compañerismo. Así mismo, agradezco a Odalys Criollo por su ayuda y colaboración, las cuales contribuyeron significativamente al desarrollo de este trabajo.

Finalmente, agradezco a todas las personas que, de manera directa o indirecta, aportaron para que este objetivo académico se haga realidad.

DEDICATORIA

Dedico este trabajo de investigación, en primer lugar, a mi hijo Ian Matías, por ser la motivación principal en cada logro y etapa de mi vida, y por impulsarme a seguir adelante con esfuerzo y perseverancia.

A mi madre, Sonia Aíde Pérez Fuelagan, y a mi padre, Silvio Alberto Castro Castro, por su amor, sacrificio, apoyo constante y por creer siempre en mí.

A mis hermanos Armando, Verónica y Luis, por su respaldo, comprensión y palabras de aliento a lo largo de este camino académico.

De manera especial, dedico este logro a mi amigo Dennis Reina, por su apoyo desde el colegio, por confiar en mí y por estar presente en los momentos en que más lo necesité.

Gracias por todo.

ÍNDICE

| | |
|--|----|
| RESUMEN | 14 |
| ABSTRACT | 15 |
| INTRODUCCIÓN | 16 |
| I. EL PROBLEMA | 17 |
| 1.1. PLANTEAMIENTO DEL PROBLEMA | 17 |
| 1.2. FORMULACIÓN DEL PROBLEMA | 18 |
| 1.3. JUSTIFICACIÓN | 18 |
| 1.4. OBJETIVOS Y PREGUNTAS DE INVESTIGACIÓN | 18 |
| 1.4.1. Objetivo General..... | 18 |
| 1.4.2. Objetivos Específicos..... | 19 |
| 1.4.3. Preguntas de Investigación..... | 19 |
| II. FUNDAMENTACIÓN TEÓRICA | 20 |
| 2.1. ANTECEDENTES DE LA INVESTIGACIÓN | 20 |
| 2.2. MARCO TEÓRICO | 23 |
| 2.2.1. Generalidades de la arveja | 23 |
| 2.2.1.1. Definición de la arveja (<i>Pisum Sativum. L</i>) | 23 |
| 2.2.2.1. El oídio en la arveja | 24 |
| 2.2.2.2. Fusarium en la arveja | 25 |
| 2.2.2.3. Antracnosis en la arveja | 26 |
| 2.2.2. Generalidades de las TIC | 27 |
| 2.2.2.1. Definición de las TIC | 27 |
| 2.2.2.2. TIC en la producción agrícola | 28 |
| 2.2.3. Visión por computador | 29 |
| 2.2.4. Visión por computador en la agricultura | 30 |

| | |
|--|-----------|
| 2.2.5. Algoritmos | 30 |
| 2.2.5.1. Algoritmos de aprendizaje | 30 |
| 2.2.5.2. Algoritmo YOLO | 32 |
| 2.2.5.3. Algoritmo R-CNN..... | 33 |
| 2.2.5.4. Teachable Machine | 35 |
| 2.2.5.5. Fundamentos del proceso de entrenamiento de la red neuronal | 36 |
| 2.2.6. Tecnologías de entrenamiento..... | 38 |
| 2.2.6.1. Tecnologías en la nube Colab | 38 |
| 2.2.6.2. API'S | 39 |
| 2.2.6.3. Gemini | 40 |
| 2.2.7. Lenguajes de programación | 42 |
| 2.2.7.1. Python..... | 42 |
| 2.2.7.2. Java | 43 |
| 2.2.8. Tipos de App | 43 |
| 2.2.8.1. App móvil | 43 |
| 2.2.8.2. App web..... | 44 |
| 2.2.8.3. App de escritorio | 45 |
| 2.2.9. Entorno de desarrollo | 46 |
| 2.2.9.1. Visual Studio Code | 46 |
| 2.2.9.2. Flutter | 47 |
| 2.2.9.3. TensorFlow | 47 |
| 2.2.10. Gestores de bases de datos | 47 |
| 2.2.11. Base de datos relacional- no relacional | 48 |
| III. METODOLOGÍA | 50 |
| 3.1. ENFOQUE METODOLÓGICO | 50 |
| 3.1.1. Enfoque | 50 |
| 3.1.2. Tipo de Investigación | 50 |
| 3.1.2.1. Investigación descriptiva | 50 |

| | |
|---|-----------|
| 3.1.2.2. Investigación explicativa | 50 |
| 3.1.2.3. Investigación de campo | 51 |
| 3.2. IDEA A DEFENDER | 51 |
| 3.3. DEFINICIÓN Y OPERACIONALIZACIÓN DE LAS VARIABLES | 51 |
| 3.3.1. Definición de variables | 51 |
| 3.3.2. Operacionalización de variables | 52 |
| 3.4. MÉTODOS UTILIZADOS | 54 |
| 3.4.1. Observación directa | 54 |
| 3.4.2. Métodos transversales..... | 54 |
| 3.4.3. Técnicas | 54 |
| 3.5. ANÁLISIS ESTADÍSTICO | 54 |
| IV. RESULTADOS Y DISCUSIÓN..... | 55 |
| 4.1. RESULTADOS | 55 |
| 4.1.1. Análisis de la entrevista | 55 |
| 4.1.2. Análisis de la encuesta | 56 |
| 4.1.3. Análisis de resultados de la Aplicación "ARVEJA CHECK" | 61 |
| 4.1.4. Propuesta | 64 |
| 4.1.4.1. Arquitectura | 66 |
| 4.1.4.2. Fase de factibilidad..... | 67 |
| 4.1.4.3. Factibilidad económica | 68 |
| 4.1.4.4. Factibilidad operativa, situación actual | 68 |
| 4.1.4.5. Situación actual..... | 69 |
| 4.1.5. Metodología XP..... | 69 |
| 4.1.5.1. Fase de planificación..... | 69 |
| 4.1.5.2. Fase de diseño | 72 |
| 4.1.5.3. Fase de codificación | 75 |
| 4.1.5.4. Creación modelo de detección de enfermedades | 78 |
| 4.1.5.5. Codificación Visual Studio Code | 80 |

| | |
|--|------------|
| 4.1.5.6. Fase de prueba | 86 |
| 4.1.6. Análisis de la evaluación- Aplicación ARVEJA CHECK..... | 90 |
| 4.2. DISCUSIÓN | 91 |
| V. CONCLUSIONES Y RECOMENDACIONES | 92 |
| 5.1. CONCLUSIONES | 92 |
| 5.2. RECOMENDACIONES | 92 |
| VI. REFERENCIAS BIBLIOGRÁFICAS | 94 |
| VII. ANEXOS | 101 |

ÍNDICE DE TABLAS

| | |
|--|----|
| Tabla 1. Principales enfermedades en los cultivos de arveja..... | 24 |
| Tabla 2. Ejemplos de visión por computador..... | 30 |
| Tabla 3. Ejemplos de uso de Gemini en el campo de la computación. | 41 |
| Tabla 4. Tipos de apps móviles. | 44 |
| Tabla 5. Características de las App | 45 |
| Tabla 6. Aplicaciones de escritorio vs aplicaciones web. | 46 |
| Tabla 7. Gestores de base de datos | 48 |
| Tabla 8. Diferencias de las bases de datos relacionales y las no relacionales..... | 49 |
| Tabla 9. Operacionalización de variables | 53 |
| Tabla 10. Recursos de software | 67 |
| Tabla 11. Recursos de hardware | 68 |
| Tabla 12. Factibilidad económica | 68 |
| Tabla 13. Roles..... | 69 |
| Tabla 14. Tiempo | 69 |
| Tabla 15. H_usuario 1 | 70 |
| Tabla 16. H_usuario 2 | 70 |

| | |
|-----------------------------|----|
| Tabla 17. H_usuario 3 | 71 |
| Tabla 18. H_usuario 4 | 71 |
| Tabla 19. H_usuario 5 | 71 |
| Tabla 20. H_usuario 6 | 71 |
| Tabla 21. H_usuario 7 | 71 |
| Tabla 22. H_usuario 8 | 72 |

ÍNDICE DE FIGURAS

| | |
|---|----|
| Figura 1. Oídio en la arveja | 25 |
| Figura 2. Arveja afectada por Fusarium oxysporum..... | 26 |
| Figura 3. Vainas de arveja afectadas por antracnosis | 27 |
| Figura 4. Entrenamiento Teachable Machine..... | 36 |
| Figura 5. Épocas, tamaño de lote y tasa de aprendizaje | 38 |
| Figura 6. Género | 56 |
| Figura 7. Implementación de un prototipo | 57 |
| Figura 8. Tipo de método | 58 |
| Figura 9. Uso de tecnología | 58 |
| Figura 10. Factores que influyen en la enfermedad oídio | 59 |
| Figura 11. Desarrollo del oídio..... | 60 |
| Figura 12. Beneficios de la introducción de un prototipo..... | 60 |
| Figura 13. Características de la enfermedad oídio..... | 61 |
| Figura 14. Detección con luz natural | 62 |
| Figura 15. Detección con iluminación artificial..... | 63 |
| Figura 16. Flash del dispositivo móvil | 63 |
| Figura 17. Detección con baja luminosidad..... | 64 |

| | |
|---|----|
| Figura 18. Iluminación de la detección de enfermedades con "Arveja Check" | 64 |
| Figura 19. Errores de pantalla azul..... | 65 |
| Figura 20. Prototipo login..... | 72 |
| Figura 21. Prototipo menú de inicio | 73 |
| Figura 22. Prototipo de información..... | 75 |
| Figura 23. Visual Studio Code | 76 |
| Figura 24. Autenticación del usuario a través de FireBase | 77 |
| Figura 25. Realtime Database | 77 |
| Figura 26. API de Google Gemini (versión gemini-2.0-flash-lite)..... | 78 |
| Figura 27. Agregar imágenes de hojas de arveja de amarre con enfermedades..... | 78 |
| Figura 28. Marcación de imágenes para crear el modelo de detección | 79 |
| Figura 29. Etiquetado de imágenes con las enfermedades de la arveja | 79 |
| Figura 30. Conversión modelo TensorFlow lite..... | 80 |
| Figura 31. Dependencia en pubspec.yaml | 80 |
| Figura 32. creación interfaz login o inicio de sesión | 81 |
| Figura 33. creación interfaz de registro..... | 81 |
| Figura 34. Bienvenida después de logearse..... | 82 |
| Figura 35. Detección de enfermedades | 82 |
| Figura 36. Menú del contenido de la aplicación | 83 |
| Figura 37. Guía de usuario | 83 |
| Figura 38. información del creador de la aplicación | 84 |
| Figura 39. Asistente virtual especializado en arveja | 84 |
| Figura 40. Interfaz gráfica del clima..... | 85 |
| Figura 41. Calendario lunar..... | 85 |
| Figura 42. Historial de detecciones | 86 |
| Figura 43. Selección del dispositivo y aplicación a analizar..... | 86 |
| Figura 44. Test de pruebas | 87 |

| | |
|--|-----|
| Figura 45. Entorno de pruebas..... | 87 |
| Figura 46. Información de compatibilidad de la app con versiones Android..... | 88 |
| Figura 47. Resumen de prueba con resultado PASS (satisfactorio) | 88 |
| Figura 48. Resultado de los niveles de uso de RAM, CPU y red en el dispositivo | 88 |
| Figura 49. Prueba robo o stress realizado a la aplicación Arveja Cheek..... | 89 |
| Figura 50. Rendimiento CPU, gráficos y RAM | 89 |
| Figura 51. Resultados de rendimiento de la prueba en Test Lab | 90 |
| Figura 52. Aprobación por parte del docente | 90 |
| Figura 53. Comunidad de Chután Bajo" El Falso" | 103 |
| Figura 54. Sector "Cascada de Paluz"..... | 103 |
| Figura 55. Comunidad de Cumbaltar..... | 103 |
| Figura 56. Comunidad de Tangüis | 103 |
| Figura 57. Comunidad de Tangüis Alto..... | 103 |

ÍNDICE DE ANEXOS

| | |
|--|-----|
| Anexo 1. Acta de la sustentación de Predefensa del TIC | 101 |
| Anexo 2. Certificado del abstract por parte de idiomas..... | 102 |
| Anexo 3. Toma de muestras..... | 103 |
| Anexo 4. Preguntas de evaluación y aprobación | 104 |
| Anexo 5. Manual de usuario | 106 |

RESUMEN

El objetivo del presente trabajo fue desarrollar un sistema basado en visión por computadora e inteligencia artificial para la detección automática de enfermedades en el cultivo de *Pisum sativum* L. (arveja) en la provincia del Carchi, Ecuador. Mediante el uso de algoritmos de aprendizaje profundo aplicados al análisis de imágenes digitales, se logró identificar y clasificar enfermedades relevantes como antracnosis, fusarium, oídio y mildiu, las cuales inciden directamente en la productividad y el rendimiento de los cultivos. La metodología empleada correspondió a un enfoque mixto, de tipo cuantitativo y cualitativo, con una investigación descriptiva, explicativa y de campo. Para el desarrollo del sistema se aplicó la metodología ágil Extreme Programming (XP), caracterizada por procesos iterativos de planificación, diseño, codificación y pruebas continuas. La aplicación móvil, denominada ARVEJA CHECK, incorpora módulos complementarios como información climática, calendario lunar y un módulo de chatbot inteligente basado en la API de Gemini, orientado a brindar asistencia técnica y recomendaciones al agricultor; este módulo requiere conexión a internet para su funcionamiento. El entrenamiento del modelo de detección de imágenes se realizó mediante la herramienta web Teachable Machine de Google, utilizando una red neuronal convolucional, la cual fue exportada en formato TensorFlow Lite, permitiendo que la detección de enfermedades se ejecute directamente en el dispositivo móvil sin requerir conexión a internet. Los resultados obtenidos evidencian una precisión de detección del 100 % bajo condiciones óptimas de iluminación natural y una fiabilidad superior al 93,8 % en condiciones de baja luminosidad con apoyo de iluminación auxiliar. En conclusión, el sistema desarrollado constituye una herramienta tecnológica eficiente que contribuye a la detección temprana de enfermedades en el cultivo de arveja, promoviendo prácticas agrícolas más sostenibles y el fortalecimiento de la producción agrícola local.

Palabras clave: visión por computadora, inteligencia artificial, aprendizaje profundo, arveja, enfermedades.

ABSTRACT

The objective of this study was to develop a system based on computer-vision training algorithms for the automatic detection of diseases in *Pisum sativum* L. (pea) crops in the province of Carchi, Ecuador. By applying deep-learning techniques to the analysis of digital images, it was possible to identify and classify major diseases such as anthracnose, fusarium, powdery mildew, and downy mildew, all of which directly affect crop productivity and yield. The methodology followed a mixed approach, combining quantitative and qualitative components, and was grounded in descriptive, explanatory, and field research. For system development, the agile Extreme Programming (XP) methodology was applied, characterized by iterative processes of planning, design, coding, and continuous testing. The resulting mobile application, named ARVEJA CHECK, integrates complementary modules such as weather information, a lunar calendar, and an intelligent chatbot based on the Gemini API, designed to provide technical assistance and recommendations to farmers. This module requires an internet connection to operate. The image-detection model was trained using Google's Teachable Machine web tool, employing a convolutional neural network. The model was then exported in TensorFlow Lite format, enabling disease detection to run directly on the mobile device without requiring an internet connection. The results show a detection accuracy of 100% under optimal natural-light conditions and reliability above 93.8% in low-light environments with artificial illumination support. In conclusion, the system developed represents an efficient technological tool that supports early detection of diseases in pea crops, promotes more sustainable agricultural practices, and strengthens local agricultural production.

Keywords: computer vision, artificial intelligence, deep learning, peas, diseases.

INTRODUCCIÓN

La automatización de procesos agrícolas mediante tecnologías de información y comunicación se ha convertido en una necesidad para el sector rural ecuatoriano, especialmente en la provincia del Carchi, donde el cultivo de arveja (*Pisum Sativum L.*) constituye una fuente importante de ingresos para las familias agrícolas. Sin embargo, las enfermedades que afectan este cultivo generan pérdidas económicas significativas debido a la disminución de la calidad y cantidad de la producción, así como al aumento de los costos por manejo y recuperación de las plantas dañadas.

Ante este escenario, el objetivo principal de esta investigación es desarrollar un sistema tecnológico basado en visión por computadora e inteligencia artificial, que permita la detección automática y clasificación de enfermedades en plantas de arveja a través del análisis de imágenes digitales. La implementación de herramientas de aprendizaje profundo busca incrementar la precisión y rapidez en el diagnóstico fitosanitario, optimizando los recursos y favoreciendo la toma de decisiones por parte de los productores locales.

El estudio está estructurado en capítulos que abordan desde la identificación del problema y el análisis del contexto productivo local, hasta la fundamentación teórica sobre el uso de inteligencia artificial en la agricultura, para finalmente describir la metodología ágil utilizada para el desarrollo del sistema y mostrar los resultados obtenidos en pruebas de campo y validación de la herramienta propuesta.

Con ello se espera aportar una solución práctica para mitigar las pérdidas económicas causadas por enfermedades, contribuir a la seguridad alimentaria regional y fomentar prácticas agrícolas sostenibles a través de la adopción de tecnologías innovadoras.

I. EL PROBLEMA

1.1. PLANTEAMIENTO DEL PROBLEMA

La agricultura constituye un sector esencial para el consumo humano que debe ir de la mano con el progreso tecnológico, aunque se considera uno de los ámbitos menos desarrollados globalmente. Esta situación es evidente en regiones rurales de África como el sur de Asia, donde prevalece la pobreza y el hambre, los cuales requieren de nuevos métodos y sistemas para mejorar las condiciones de vida.

Las patologías que afectan a los cultivos de guisantes (*Pisum sativum*) reducen su productividad y calidad de la cosecha. Dichas enfermedades son provocadas por patógenos como hongos, bacterias o virus que afectan a la estructura de la planta, desde hojas, tallos y semillas. La principal problemática de estas enfermedades es que pueden causar tallos marchitos, cogollos deformados, manchas necróticas en las hojas, disminución de los cogollos e incluso la muerte de la planta. Además, generan consecuencias económicas para los agricultores, ya que deben enfrentar pérdidas de cosecha y deben invertir en medidas de control y tratamiento para controlar la enfermedad.

Según mencionan (Fuglie et al., 2020) el 80% de la población mundial en situación de pobreza reside en zonas rurales, y su principal fuente de ingresos proviene de la agricultura. Por lo que, aumentar la productividad agrícola mediante el uso de la tecnología contribuye a la disminución de pobreza.

La arveja es la segunda leguminosa más consumida en los hogares de Ecuador, sin importar los estratos socioeconómicos. La Provincia del Carchi lidera la producción nacional, representando el 47,46%, por lo que se ha dedicado en los últimos años a técnicas y manejo de los cultivos, reconociendo que las enfermedades que lo afectan provocan pérdidas económicas para los agricultores. Según (Angulo, 2019) las principales enfermedades de esta leguminosa son: oídio (47%), antracnosis (20%) y fusarium (33%).

1.2. FORMULACIÓN DEL PROBLEMA

¿En qué medida la visión por computadora ayuda en la detección de enfermedades en la *Pisum Sativum.L* (Arveja)?

1.3. JUSTIFICACIÓN

La arveja (*Pisum Sativum.L*) es la leguminosa de mayor consumo de los ecuatorianos, y la provincia del Carchi su principal productora. Por lo que, la implementación de la tecnología para incrementar el rendimiento de la productividad agrícola es fundamental para contribuir en la disminución de la pobreza (Angulo, 2019). Por lo tanto, para mejorar el rendimiento y evitar pérdidas económicas a causa de las enfermedades, es importante incorporar métodos de procesamiento de imágenes digitales que detecten a tiempo la presencia de plagas y patologías.

De este modo, la investigación es conveniente debido a que aborda un problema crítico y observable. La implementación de la visión por computadora sirve para dar una solución práctica y precisa para los agricultores, permitiéndoles identificar las enfermedades en sus cultivos de forma temprana y así prevenir pérdidas económicas. Además, el uso de la tecnología no solo tiene un impacto positivo en la economía y el bienestar, sino también garantiza la seguridad alimentaria, brindando un producto de buena calidad. Por otro lado, la información obtenida puede servir para desarrollar modelos futuros sobre teledetección en la agricultura (Piedad et al., 2020).

Esta investigación contribuye a varios objetivos tales como:

- ODS 3: Salud y bienestar. Menos plagas y enfermedades significan menos uso de agroquímicos, y esta reducción de pesticidas contribuye directamente a mejorar la salud de agricultores, familias y consumidores.
- ODS 12: Producción y consumo responsable. La detección temprana y precisa de enfermedades en los cultivos de arveja reduce el uso de pesticidas, fomentando prácticas sostenibles.

1.4. OBJETIVOS Y PREGUNTAS DE INVESTIGACIÓN

1.4.1. Objetivo General

Diseñar un sistema a través de visión de computadora para la detección de enfermedades en el cultivo de arveja (*Pisum Sativum.L*).

1.4.2. Objetivos Específicos

- Identificar los diferentes tipos de enfermedades y sus características distintivas que afectan el cultivo de arvejas.
- Analizar como la visión por computador permite el reconocimiento de patógenos en el cultivo de la arveja.
- Proponer una aplicación tecnológica para la detección de para la detección de enfermedades como antracnosis, fusarium, oídio y mildiu en el cultivo de la arveja.

1.4.3. Preguntas de Investigación

- ¿Cuáles son los diferentes tipos de enfermedades que afectan al cultivo de arveja y sus características distintivas?
- ¿Qué herramientas de visión por computadora están disponibles para el reconocimiento de patógenos en el cultivo de arvejas?
- ¿Qué tan eficiente es el reconocimiento de enfermedades predominantes en el cultivo de esta leguminosa que pueden ser utilizados en una aplicación de visión por computadora?

II. FUNDAMENTACIÓN TEÓRICA

2.1. ANTECEDENTES DE LA INVESTIGACIÓN

(Guerrero, 2021) en su estudio sobre reconocimiento de lesiones necróticas para la detección de la plaga thrips en los cultivos de guisantes, facilitó la evaluación de productos y la previsión de cosechas. Sin embargo, los cambios climáticos y condiciones atmosféricas del campo hacen que la detección de estos patógenos sea un reto permanente. Por lo que, propuso la creación de un sistema rápido y eficiente diseñado para reconocer lesiones y detectar la presencia de plagas en la soja. Para la implementación del sistema, empleo un smartphone y un dron DJI Mavic Mini para la recolección de imágenes en formato JPT directamente de las parcelas. Posteriormente, utilizo la plataforma Roboflow para el etiquetado, extracción y segmentación de las características esenciales del objeto de estudio de cada fotografía. La arquitectura de red neuronal convolucional (CNN) Yolov4-Tiny fue seleccionada para llevar a cabo la detección de la plaga thrips en los cultivos. El desarrollo del sistema se apoyó en herramientas de código abierto como Python y librerías clave de aprendizaje automático como TensorFlow y de visión artificial como Open CV. El sistema se sometió a pruebas, obteniendo como resultados de IoU (Intersección sobre la Unión) del 59,23 %, una mAP (Precisión media) del 87,8 % y una efectividad del sistema del 80 %.

(Martínez & Guerrero, 2022) en su investigación sobre reconocimiento y detección de la plaga de trips en guisantes, es importante para la agricultura de precisión, debido a que permite la estimación del rendimiento y la predicción de cosechas. Sin embargo, las condiciones atmosféricas y los factores climáticos hacen que la detección de plagas sea un desafío constante para los agricultores. Por lo que propusieron un sistema rápido y eficiente para detectar la infestación de trips en guisantes, utilizando un modelo profundo Yolov4-tiny, un dron DJI Mavic mini y un teléfono inteligente para recopilar imágenes del campo. A su vez, con la herramienta Roboflow extrajeron, etiquetaron y segmentaron las características más importantes del objeto de estudio. Comprobaron que los datos de alta calidad que la intersección

sobre la unión (IoU) fue 59,23 % y la precisión promedio (mAP) de 81,60 %. Además, las predicciones del sistema se validaron con un diagnóstico de un experto en producción agrícola, concluyendo una eficacia del sistema en detección de dicha plaga del 86%.

(Gómez et al., 2022) en su estudio sobre técnicas de procesamiento digital de imágenes para detección de plagas y enfermedades de cultivos, mencionan que la detección de patógenos en los cultivos es importante para aumentar el rendimiento de la productividad de manera sostenible. La agricultura 4.0 integra sistemas computacionales y tecnologías avanzadas para optimizar los procesos agrícolas, además las técnicas digitales de imágenes facilitan detectar las plagas y enfermedades de cultivos como cereales, frutales y tubérculos; ayudando a reducir las pérdidas económicas. Actualmente, el 40 % de los cultivos se pierden debido a patógenos, que suelen manifestarse con síntomas durante el crecimiento de las plantas. Por el contrario, el diagnóstico basado en la observación humana suele ser menos preciso y erróneo. Por ello, analizaron distintas técnicas de procesamiento digital de imágenes, abarcando la captura de imágenes, procesamiento, segmentación, extracción y clasificación de enfermedades o plagas.

(Morán & Quiranza, 2022) en su investigación sobre visión artificial para la detección temprana de afecciones en el cultivo de papa, utilizaron un enfoque mixto: el componente cualitativo se dedica a examinar a fondo las características y detalles de cada patología, mientras que el enfoque cuantitativo procesa los datos estadísticos importantes recopilados. Además, combinaron diferentes tipos de estudio: descriptivo, explicativo, exploratorio, documental y de campo. Por otro lado, aplicaron métodos como la entrevista para recabar datos sobre los procedimientos de detección actuales y las prácticas previas a la siembra. Se constató que los agricultores determinan la presencia de estas patologías basada en la experiencia. Para el desarrollo del sistema utilizaron metodología XP (Programación Extrema) favoreciendo la comunicación constante entre los usuarios y el equipo de desarrollo. Concluyendo que es un software ágil y eficaz apto para diagnosticar las enfermedades.

(Guerrero, 2021) en su estudio sobre detección de la plaga thrips en los cultivos de arveja. Siendo una actividad fundamental para evaluar el rendimiento de cada cosecha, debido a factores climáticos y enfermedades patológicas que afectan a los sembríos. Presento un sistema de reconocimiento rápido y eficiente de lesiones

necróticas para la detección temprana de plaga thrips en la arveja usando técnicas de Deep Learning. Para la recolección de imágenes JPG desde las parcelas, se empleó un smartphone y un dron DJI Mavic Mini. Durante la creación del sistema utilizó un conjunto de datos compuestos por 502 imágenes y se ejecutaron 6000 iteraciones de entrenamiento. Los resultados de la Intersección sobre la Unión (IoU) alcanzaron el 59.23%, la Precisión Media (mAP) del 81.60% con una eficiencia del 86%. Además, el modelo tuvo un rendimiento satisfactorio con un promedio del 96%.

(Campoverde, 2024) en su investigación sobre diseño de aplicación para detección de enfermedades en el cultivo de pimiento, presentó un prototipo de vanguardia para que los agricultores identifiquen las enfermedades y ataques de plagas en sus sembríos. La tecnología central se basó en la aplicación de visión artificial junto a redes neuronales convolucionales (CNN), lo que permitió un diagnóstico automático y altamente fiable de anomalías en las plantas. El objetivo es agilizar el proceso de detección de problemas, disminuyendo el lapso entre la manifestación inicial de los síntomas y garantizar intervenciones rápidas y eficientes para reducir las pérdidas de rendimiento. Este prototipo es fundamental en el campo de la agricultura de precisión, ya que ayuda a los expertos a tomar decisiones importantes para incrementar la sostenibilidad y productividad del cultivo de pimientos en Ecuador.

(Zambrano D. , 2024) en su estudio sobre detección de plagas en el maíz, creó una aplicación móvil "PlagaScan" para detectar las plagas en este cultivo y ofrecer asistencia en el diagnóstico a agricultores principiantes que tienen problemas para identificarlas. La aplicación constó con tres funciones: el diagnóstico de plagas en el momento, a través de fotos subidas desde la galería, y una sección de control para posibles estrategias de manejo para la plaga detectada. Los datos obtenidos se produjeron a través de un modelo informático de aprendizaje cimentado en la arquitectura de red neuronal convolucional MobileNet-v2, el cual fue entrenado y verificado usando 750 imágenes. Este modelo se programó para identificar 5 plagas: gusano trozador, gallina ciega, grillo topos, gusano alambre y gusano lepidóptero; para la creación de la aplicación eligió la metodología Mobile-D, diseñada especialmente para el desarrollo de apps móviles. Concluyendo que la aplicación tiene una eficiencia alta, logrando una precisión del 56.42 %, siendo un instrumento valioso para agricultores con poca experiencia en la identificación de plagas en el cultivo de maíz.

2.2. MARCO TEÓRICO

2.2.1. Generalidades de la arveja

2.2.1.1. Definición de la arveja (*Pisum Sativum. L*)

La arveja es un guisante, científicamente conocido como *Pisum Sativum. L*, cultivo tiene una larga historia, y se cree que se originó en Asia central, Asia menor, la cuenca del Mediterráneo o Etiopía. Hoy en día es cultivada a nivel mundial, debido a que su principal importancia es la obtención de cogollos y semillas comestibles.

Esta planta pertenece a la familia de las leguminosas y se caracteriza por sus tallos que pueden alcanzar 2 metros de altura, flores blancas y hojas compuestas terminadas en zarcillos. Su fruto es una vaina que contiene semillas esféricas y es la fuente de alimento del ser humano (Ramos, 2024). La arveja puede consumirse cruda o cocida y venderse en botellas o latas. Actualmente existe una nueva tendencia de productos congelados rica en proteínas, vitaminas y minerales.

2.2.2. Principales enfermedades en el cultivo de arveja

Las enfermedades en los cultivos de arvejas son alteraciones patológicas causadas por virus, hongos o bacterias y están influenciadas por las condiciones climáticas, es decir, lluvias frecuentes, rocío y alta humedad. Dichas enfermedades pueden afectar a hojas, tallos, raíces y semillas y se manifiestan a través de tizones, manchas, pudriciones y amarillez.

En la tabla 1 se indican las generalidades de las enfermedades en los cultivos de arvejas.

Tabla 1. Principales enfermedades en los cultivos de arveja

| Nombre de la enfermedad | Características y síntomas principales | Causas | Medidas de manejo y control |
|-------------------------------------|--|--|--|
| Complejo Ascochyta | Manchas circulares de color café en hojas, tallos y vainas. Además, puede causar pudrición de raíz. | Humedad y rocío generan hongos y facilitan su propagación mediante salpicaduras de agua. | Uso de semillas de buena calidad y aplicación de fungicidas como Oxiclورو de cobre y Mancozeb. |
| Oídio | Crecimiento de color blanco como ceniza en hojas, tallos y vainas. | Alta densidad de siembra. | Control de malezas y aplicación de fungicidas sistémicos. |
| Fusarium spp., Rhizoctonia, Pythium | Amarillez y desecación de la planta, decoloración intensa café-rojiza. | Exceso de humedad en el suelo | Semillas desinfectadas |
| Antracnosis | Manchas de color pardo en tallos, hojas y vainas. Pueden contaminar y ser transmitidas por la semilla. | Humedad y rocío prolongado. | Semillas certificadas y uso de fungicidas a base de Oxiclورو de cobre. |

Fuente: (Saavedra, 2019)

2.2.2.1. El oídio en la arveja

Según mencionan (Lago et al., 2021) el oídio es una afección que afecta a las legumbres a nivel mundial. El principal causante es el hongo *Erisiphe pisi DC*, común en la región sur de Santa Fe.

Esta enfermedad se desarrolla bajo las condiciones climáticas, principalmente en las épocas de primaveras secas, combinadas con noches frescas y húmedas entre 21 °C y 27 °C. La sintomatología se caracteriza por la aparición de manchas blancas en hojas y tallos. Tal y como se muestra en la figura 1, pudiendo extenderse a otras estructuras como ramas, semillas y vainas. Bajo condiciones óptimas, este patógeno puede completar su ciclo en siete días, lo que explica su alta tasa de propagación. El impacto del oídio en los cultivos es considerable, debido a que reduce la capacidad fotosintética y disminuye los rendimientos hasta en un 59%.



Figura 1. Oídio en la arveja
Fuente: (Valencia, 2018)

De este modo, para el manejo efectivo de este hongo se recomienda la implementación de técnicas, como el manejo adecuado de fecha de siembra, el uso de genotipos resistentes y control de químicos, que ayudan a controlar la propagación de esa enfermedad.

2.2.2.2. Fusarium en la arveja

El *Fusarium spp* es un tipo de hongo que perjudica principalmente a los sembríos de arveja y provoca la enfermedad denominada "chupadera fungosa", dañando las raíces y amarillando a toda la planta, afectando su productividad y rendimiento. Según mencionan (Franco & Tokin, 2020) este patógeno se propaga en temperaturas moderadas entre 18 °C - 20°C, a una humedad relativa superior al 90%. La arveja (*Pisum Sativum*) presenta vulnerabilidad a diferentes enfermedades, destacándose la marchitez por *Fusarium oxysporum*, como se indica en la figura 2, la cual produce daños en los cultivos de guisantes, provocando daños significativos en el rendimiento y calidad. Sin embargo, no existe información relevante de estrategias de control biológico para combatir esta patología.



Figura 2. Arveja afectada por *Fusarium oxysporum*

Fuente: (Pabón & Castaño, 2012)

Este hongo daña las raíces del cultivo, ya que actúa sobre el sistema radicular y progresa a través de este, provocando una obstrucción en los haces vasculares del flujo de agua y nutrientes. Como consecuencia, se produce el marchitamiento del cultivo, clorosis foliar y muerte de la planta. Además, las condiciones óptimas de crecimiento de este patógeno son suelos húmedos y temperaturas cálidas. A su vez, por semillas infectadas o residuos de cultivos procedentes de suelos contaminados.

Para controlar esta enfermedad se recomienda implementar estrategias preventivas y de control, entre ellas:

- Selección de variables con resistencia genética a *Fusarium*.
- Rotación de cultivos
- Eliminación de residuos vegetales infectados.
- Prácticas adecuadas de manejo de suelo.
- Control de insectos y plagas portadores de enfermedades.

2.2.2.3. Antracnosis en la arveja

La antracnosis es una enfermedad fúngica que causa daños a los cultivos de guisantes (*Pisum Sativum*), provocada por el hongo *Colletotrichum ssp* y se caracteriza por la creación de manchas necróticas en tallos y hojas e incluso en toda la planta. Las lesiones comienzan como áreas acuosas y oscuras que se propagan rápidamente, causando la muerte del tejido vegetal afectado.

Según mencionan (Lui et al., 2016) estas manchas presentan forma circular de color marrón y rodeadas por halos negros en hojas y tallos. Este patógeno se desarrolla en condiciones ideales como ambientes de alta humedad y temperaturas moderadas;

además, la presencia de residuos de cultivos y malezas contribuye significativamente a su propagación.

En la figura 3 se indica la enfermedad Antracnosis en las vainas de arveja.



Figura 3. Vainas de arveja afectadas por antracnosis

Fuente: (Valencia, 2018)

La transmisión de la antracnosis ocurre por diferentes medios:

- Residuos vegetales contaminados
- Semillas infectadas
- Salpicaduras de lluvia
- Ascosporas transportadas por el viento
- Contacto directo entre plantas

El hongo *Colletotrichum ssp* sobrevive en restos de cosecha y semillas, lo que facilita su presencia entre ciclos de cultivo.

2.2.2. Generalidades de las TIC

2.2.2.1. Definición de las TIC

Las TIC (tecnologías de la información y comunicaciones) contribuyen con un grupo de herramientas y equipos diseñados para procesar, administrar y distribuir información mediante elementos tecnológicos como televisores, ordenadores, teléfonos, entre otros (Sánchez E. , 2007). En la actualidad, el uso de estos recursos ha ido creciendo y ofreciendo servicios como correo electrónico, descarga de materiales y comercio electrónicos, búsqueda y filtrado de información.

Las TIC cumplen dos funciones primordiales:

- Proporciona acceso rápido y sencillo a la información en diferentes formatos; esto implica la digitalización de datos para su acceso o almacenamiento, incluso si la información reside en un dispositivo remoto.
- Facilita la comunicación, permitiendo compartir información de manera instantánea. Según (Alvarado, 2022) su principal función es posibilitar la comunicación bidireccional entre la población, esto se aplica a mensajería, foros y videoconferencias.

2.2.2.2. TIC en la producción agrícola

La agricultura de precisión (AP) representa una integración estratégica de diferentes tecnologías aplicadas a la producción agrícola. Se caracteriza por la toma de decisiones importantes e implementación en el campo, la incorporación de las TIC en el campo agrícola ha transformado las prácticas tradicionales, optimizando la eficiencia operativa, disminuyendo costos de producción y fortaleciendo el proceso decisorio.

Según (Rambauth, 2022) la adopción de TIC basada en big data, plataformas digitales, internet de las cosas (IoT), sistemas analíticos avanzados, entre otros. Esta optimización tecnológica resulta en un incremento de la productividad y, en consecuencia, en una mayor rentabilidad de los productos.

Las tecnologías y aplicaciones digitales cumplen un papel fundamental en la producción agrícola, ya que ayudan a los agricultores a mejorar la productividad y sostenibilidad de sus cosechas. De este modo, los agrarios utilizan sensores y dispositivos conectados para monitorear y recopilar información del suelo, la humedad, el clima y otros factores que perjudiquen a los sembríos (Angulo, 2019). Estos datos son procesados mediante técnicas de aprendizaje y algoritmos avanzados que generan alto valor sobre el estado sanitario de las plantaciones, fertilización y detección precoz de enfermedades y plagas. También, mejoran la gestión de la cadena de suministro agrícola, por medio de plataformas digitales que rastrean y gestionan el almacenamiento de sus productos. De tal forma, ayudan a optimizar procesos logísticos, mejoran la trazabilidad de los productos y reducen el desperdicio de alimentos.

Así mismo, las TIC son aplicadas en la agricultura de precisión, utilizando teledetección, SIG (sistemas de información geográfica) y drones (Ríos, 2020). Los agricultores pueden recopilar datos e imágenes sobre los campos que van a estudiar,

con la finalidad de tomar decisiones sobre el riego, la siembra, la aplicación de fertilizantes y protección de los cultivos, logrando una optimización de recursos, reducción de costos e impacto ambiental (Guerrero, 2021).

En conclusión, las TIC están transformando el sector agrícola al proporcionar herramientas que aumentan la productividad, sostenibilidad y la eficiencia. El análisis de datos y las aplicaciones permiten que los agricultores tomen decisiones basadas en información precisa y real.

2.2.3. Visión por computador

La visión por computador es un conjunto de herramientas y tecnologías que permiten captar imágenes del mundo real. Es decir, la inteligencia artificial, permitiendo que el software reconozca personas, entornos y objetivos por medio de algoritmos matemáticos (Ponce, 2012).

Según (Mirmehdi, 2012) la visión por computador se ha utilizado en la operación de los sistemas industriales modernos debido a su eficiencia.

- **Equipos electrónicos:** la selección de pieza y soldadura automática por medio de la visión artificial.
- **Industria alimentaria:** equipos con visión por computador, capaces de detectar problemas en las líneas de preparación y empaquetado, evitando pérdidas de producción y garantizando la calidad alimentaria.
- **Automoción:** Incorporación de robots para el proceso de soldadura y montaje (reconocer piezas, comprobar pintura, inspeccionar soldadura).
- **Logística:** áreas de empaquetados, despacho, alimentación de piezas, inspección de productos terminados y clasificación.
- **Seguridad:** procesar imágenes que ayudan a controlar al personal o a detectar intrusos. A su vez, detectan potenciales accidentes y detención de trabajos.

En la tabla 2 se detallan algunos ejemplos de visión por computador.

Tabla 2. Ejemplos de visión por computador

| Aplicación | Tarea principal | Características | Tecnología (IA) | Información/Salida extraída |
|-----------------------------------|-------------------------------|---|---|--|
| Vehículos automáticos | Navegación y conducción | Robustez ante condiciones climáticas | Seguimiento de objetos, fusión de sensores | Ubicaciones de peatones, estimación de distancia, límites de carril. |
| Reconocimiento facial | Identificación y verificación | Reconoce características biométricas, iluminación y expresiones. | CNN, localización de puntos clave, clasificación de imágenes. | Verificación de acceso o de emociones e identidad de la persona. |
| Inspección de calidad en fábricas | Detección de defectos | Alta precisión de procesamiento, capacidad de identificar fallas sutiles. | Clasificación de imágenes, detección de anomalías. | Piezas defectuosas, control de inventario y conteo de productos. |
| Realidad aumentada | Localización y mapeo | Mapea el entorno para superponer objetos virtuales. | Reconocimiento de marcadores | Orientación del mundo real, puntos de referencia. |

Fuente: (Mirmehdi, 2012)

2.2.4. Visión por computador en la agricultura

La agricultura de precisión se ha fundamentado en el contenido empírico. Sin embargo, actualmente los avances tecnológicos han introducido nuevos métodos que ayudan a aumentar el conocimiento sobre el desarrollo de los cultivos. Aunque esta tecnología cubre extensas áreas terrestres, su uso se limita principalmente al ámbito civil, ya que emplean plataformas como drones y satélites, que son sistemas costosos que no proporcionan información en tiempo real (Integra, 2020).

Por otra parte, el IoT (Internet de las Cosas) está cambiando este paradigma al permitir el despliegue de múltiples nodos en los campos agrícolas. Estos nodos están equipados con sensores que pueden monitorear parámetros ambientales, edáficos y vegetales. Esta estrategia ofrece dos ventajas principales: primero, obtienen datos en tiempo real sobre el cultivo, cruciales para responder ante situaciones críticas, y segundo, representa una alternativa más económica. Mediante la integración del IoT, la agricultura de precisión se puede transformar en agricultura inteligente.

2.2.5. Algoritmos

2.2.5.1. Algoritmos de aprendizaje

El aprendizaje automático son secuencias de instrucciones limitadas y desarrolladas que una computadora puede ejecutar para alcanzar una meta. Estos algoritmos

permiten que los sistemas informáticos adquieran conocimientos de manera autónoma a partir de datos, sin la necesidad de recibir una programación específica para cada actividad. Para (Díaz, 2021) es la ciencia que permite a los sistemas informáticos adquirir conocimientos y mejorar su rendimiento con la experiencia, en lugar de seguir instrucciones estrictas predefinidas.

Los algoritmos de visión por computadora utilizan el reconocimiento de patrones, este proceso implica alimentar sistemas computacionales con extensos conjuntos de datos visuales, ayudando a procesar imágenes, identificar y etiquetar los objetos que contienen y encuentran patrones. Según (González, 2021) los métodos y técnicas de aprendizaje han transformado el campo de la visión por computadora hasta convertirse en el estándar metodológico para numerosas tareas. Por otro lado, el aprendizaje automático se establece como una rama de la inteligencia artificial, ya que radica en el uso de algoritmos que identifican patrones en datos históricos para generar predicciones al aplicarlos a información nueva (Díaz, 2021).

Este aprendizaje puede ser de diferentes tipos:

- **Aprendizaje reforzado:** Es un grupo de algoritmos dentro del aprendizaje automático que operan mediante un sistema de ensayo y error. Se premia la conducta correcta o la predicción esperada y se omite la recompensa ante un resultado no deseado. Ha producido algoritmos capaces de superar a la capacidad humana en complejos juegos de estrategia como el ajedrez, el gol y otras disciplinas.
- **Aprendizaje automático supervisado:** Están diseñados para descubrir y memorizar patrones ocultos dentro de grandes conjuntos de datos previamente registrados, aplicando luego ese conocimiento para generar predicciones sobre una información (Bergmann, 2025). En este modelo, al algoritmo se le suministran los datos de un problema que ya tiene las variables de entrada como las respuestas de salida correctas que se desea que el sistema aprenda a pronosticar. Por ejemplo, los correos spam.
- **Aprendizaje automático no supervisado:** Estos algoritmos no están diseñados para mapear entradas de salidas específicas, sino para identificar estructuras y patrones inherentes en los datos sin necesidad de tener resultados correctos (Díaz, 2021).

Particularmente, las redes neuronales convolucionales (CNN) han alcanzado resultados que superan el rendimiento de última generación implementando métodos tradicionales de visión por computadora.

2.2.5.2. Algoritmo YOLO

YOLO (You Only Look Once) es un algoritmo de detección de objetos para imágenes y videos en el campo de la visión por computadora (Data Science Team, 2020). Fue desarrollado por Joseph Redmon, Santosh Divwala, Ross Girshik y Ali Farhadi en el 2016, mediante un artículo de investigación titulado "Miras una vez: detección unificada de objetos en tiempo real" (Menegaz, 2018). Esta red de detección utiliza redes neuronales que procesan las imágenes o videos en una sola pasada. Por el contrario, otros algoritmos de detección de objetos requieren de múltiples análisis para examinar la imagen completa. Gracias a este enfoque, YOLO ha logrado realizar detecciones en tiempo real con gran eficiencia (Guan, 2023).

YOLO reside en su Red Neural Convolucional (CNN) que es una técnica que permite analizar celdas de cuadrícula de manera simultánea, a través de "compresores" que facilitan la identificación de objetos en diferentes áreas de la imagen. La implementación estándar de YOLO utiliza Darknet como su CNN base, aunque se puede entrenar con otras redes convolucionales. Sin embargo, esta red mantiene neuronas convolucionales hasta el final del proceso, eliminando la necesidad de transformarla en una red tradicional. Por otro lado, este algoritmo de detección de objetos es popular debido a la velocidad, precisión de detección, buena generalización y código abierto (Keita, 2024).

Según (Kili, 2023) menciona que el mecanismo de YOLO se desarrolla a través de una secuencia de etapas:

- La imagen inicial se somete a un CNN con el propósito de extraer un mapa de características visuales.
- Luego, estas características pasan por una serie de capas densamente conectadas, las cuales generan las predicciones de las probabilidades de clase y los parámetros del cuadro delimitador.
- La imagen se estructura en una malla de celdas, donde cada una genera la tarea de predecir varios cuadros y sus respectivas probabilidades de clase.
- El producto final de la red es una colección de cajas delimitadoras y las probabilidades de clase agrupadas en cada celda de la cuadrícula.

- Posteriormente, se aplica un procedimiento de postprocesamiento conocido como supresión para refinar las cajas eliminando las supuestas (Kili, 2023).
- Una ventaja clave de YOLO es su capacidad para analizar la totalidad de la imagen en una única iteración, lo que confiere una velocidad y eficiencia superiores en comparación con arquitecturas de detección en dos fases como: R-CNN y sus sucesores.

Yolo se ha desarrollado en diversas versiones, como YOLOv1, YOLOv2, YOLOv3, YOLOv4, YOLOv5, YOLOv6 y YOLOv7. Cada versión se ha desarrollado sobre la versión anterior con mayor precisión y características mejoradas.

- **YOLOv2:** Conocido como YOLO 9000, la CNN predice las coordenadas del cuadro y las cajas de anclaje.
- **YOLOv3:** Se emplea una técnica llamada red piramidal de características y está compuesta por múltiples capas de filtro.
- **YOLOv4:** Utiliza la técnica de procesamiento de pirámides espaciales para extraer imágenes a diferentes resoluciones y escalas.
- **YOLOv5:** Se basa en la arquitectura EfficientNet que alcanza gran rendimiento y logra una alta precisión (Kili, 2023).
- **YOLOv6:** Ejecuta más rápido con menos recursos computacionales.
- **YOLOv7:** Implemento cajas de anclaje mejoradas, utilizadas para identificar objetos en diferentes formas.

2.2.5.3. Algoritmo R-CNN

El algoritmo de red neuronal convolucional basada en regiones (R-CNN) es aquel modelo utilizado en el aprendizaje automático para detectar objetos en imágenes. Para (Petru, 2023) es un método de arquitectura de aprendizaje profundo que se usa en visión artificial para la detección de objetos. Fue un modelo innovador que impulsó el desarrollo de la detección de objetos e integra de manera efectiva el poder de las redes neuronales convolucionales con estrategias que se enfocan en regiones específicas de una imagen. Además, para la detección de objetos requiere de tres pasos principales, como: generar propuestas de regiones, extraer características y clasificar objetos (Vina & Vina, 2024).

Se fundamenta en dividir la tarea de detección en dos pasos principales:

- **Generación de propuesta de región:** Utiliza algoritmos de recomendación de regiones, similar a la búsqueda selectiva, para crear un conjunto de zonas en

la imagen donde podrían encontrarse objetos. Estas zonas o regiones son extraídas y posteriormente transformadas para que todas tengan las mismas dimensiones antes de ser procesadas por una red convolucional.

- **Clasificación de región de propuesta:** La región propuesta se clasifica empleando una red convolucional entrenada. Este sistema extrae características específicas de cada zona y calcula la probabilidad de que contengan un objeto de una determinada clase. Adicionalmente, se refinan las coordenadas del área para lograr una delimitación exacta de los objetos.

Según menciona (Petru, 2023) algunas de las ventajas que tiene el método R-CNN es que permite una detección de objetos muy exacta al emplear características extraídas de regiones mediante redes convolucionales. Además, exhiben una notable solidez ante las diferentes variaciones de las imágenes, como los cambios en su orientación, tamaño y escala. Su flexibilidad inherente convierte a R-CNN en una arquitectura adaptable para diferentes aplicaciones de detección, tales como: el seguimiento y la segmentación. Sin embargo, presenta diferentes limitaciones, como exigir una elevada capacidad de cómputo debido a su proceso multifásico y la generación de propuestas de regiones superpuestas introduce cálculos redundantes, lo cual perjudica el rendimiento general de la detección. Finalmente, no opera de forma integral, ya que depende de módulos separados para proponer regiones y clasificarlas (Petru, 2023).

Por otro lado, (Data Science Team, 2020) menciona que, para evitar el problema de seleccionar muchas áreas, Ross Girshick y colaboradores propusieron extraer solo 2000 regiones de la tabla y utilizar una búsqueda especial llamada sugerencia de ubicación.

Por consiguiente, R-CNN ha evolucionado a través de sus versiones como Fast R-CNN y Faster R-CNN.

- **Fast R-CNN:** Se centra en procesar la imagen completa en una única pasada empleando la red convolucional para extraer todas sus características, lo que optimiza la eficiencia comparada con el R-CNN.
- **Faster R-CNN:** Incorpora el logaritmo de región dentro de la propia red convolucional, lo que permite una detección más rápida y precisa.

2.2.5.4. Teachable Machine

Esta aplicación web de Google tiene como objetivo generar modelos de aprendizaje automático de forma fácil, rápida y accesible sin necesidad de programar. Esta plataforma está diseñada para que cualquier persona entrene a su computadora y reconozca sus propias imágenes, sonidos y posturas (INTEF, 2019). Según (Armendariz et al., 2023) es una herramienta fácil de usar, ya que permite entrenar modelos de aprendizaje automático para identificar imágenes, sonidos o poses. A su vez, (Kurz & Jayasuriya, 2024) añade que con ella se puede experimentar la inteligencia artificial sin necesidad de programar o escribir códigos.

El proceso consta de tres pasos principales:

- **Recopilar:** La computadora aprende a clasificar las imágenes, sonidos o poses en clases o categorías.
- **Entrenar:** Ocurre directamente en el navegador utilizando los ejemplos de las clases para aprender patrones.
- **Exportar:** Una vez que el modelo esté entrenado y aprobado, se procede a exportarlo para usarlo en tus sitios webs, aplicaciones o proyectos.

Teachable Machine es una herramienta adaptable que acepta tanto archivos como ejemplos capturados en tiempo real. Lo más importante es que prioriza la privacidad: puedes usarla totalmente sin conexión, asegurando que los datos de la webcam o el micrófono permanezcan en el ordenador.

- **Imágenes:** Entrena un modelo para la clasificación de imágenes, utilizando archivos existentes como capturas de la webcam. El proceso comienza con la recopilación de un conjunto de datos que servirá para definir las categorías, como perros y gatos. Luego se ajustan parámetros clave como ciclos de entrenamiento, números de muestras y velocidad de corrección.
- **Sonidos:** Graba clips de audio para enseñar al modelo a categorizar distintos tipos de sonido. El proceso inicial necesita la recolección de un conjunto de datos, es decir cargar muestras de audio que puedan incluir música, silbidos o aplausos, permitiendo al modelo diferenciar y etiquetar cada tipo de entrada auditiva. Una vez que el modelo ha sido entrenado, la exportación permite exportarlo en formatos de TensorFlow.js o TensorFlow Lite (Kurz & Jayasuriya, 2024).

- **Posturas:** Enseña al modelo a clasificar movimientos y posturas usando archivos o adoptando poses corporales específicas que capte la webcam.

Esta herramienta de Google se destaca por hacer el aprendizaje automático y accesible para todos, ya que su interfaz intuitiva elimina la necesidad de escribir código. Las capacidades de Teachable Machine 2.0 abarcan la clasificación de distintos tipos de datos (auditivos, visuales y de movimiento). Una vez terminados, los modelos pueden ser implementados fácilmente en entornos variados gracias a opciones de exportación para sitios web y apps (Armendariz et al., 2023).

Por el contrario, Teachable Machine presenta limitaciones como; reside en que el rendimiento del modelo está directamente ligado a la calidad y cantidad de los datos de entrenamiento. Si los datos son insuficientes para la diversidad real de los objetos a clasificar, el modelo mostrará dificultades al procesar nuevas entradas. Además, la herramienta está restringida a clasificar únicamente sonidos, imágenes y poses.

Teachable Machine es una herramienta de inteligencia artificial flexible y práctica para el ámbito educativo. Permite a los estudiantes no solo diseñar y entrenar un modelo de IA, sino también evaluarlo y aplicarlo directamente en la resolución de problemas. Es aquel proceso en el cual se enseña a un modelo de aprendizaje automático a clasificar nuevos ejemplos basándose en las muestras que proporciona y agrupa en categorías.

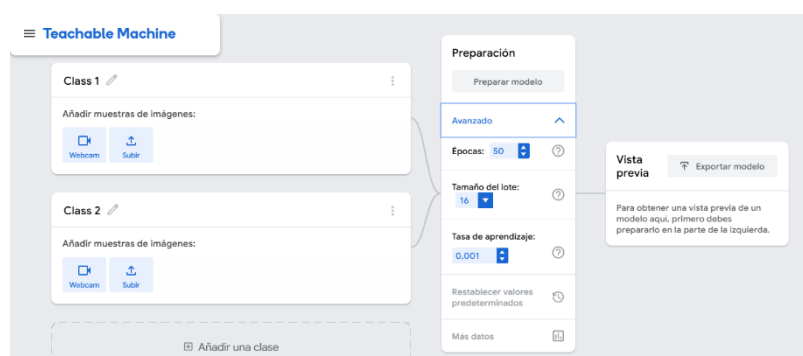


Figura 4. Entrenamiento Teachable Machine

2.2.5.5. Fundamentos del proceso de entrenamiento de la red neuronal

Épocas (Epochs): Las épocas representan el número de veces que el modelo procesa la totalidad del conjunto de datos de entrenamiento. En cada época, el algoritmo

analiza todas las imágenes disponibles con el fin de ajustar progresivamente sus parámetros internos y mejorar la capacidad de reconocimiento de patrones.

El uso de un número adecuado de épocas permite que el modelo aprenda de manera eficiente las características visuales asociadas a las diferentes enfermedades del cultivo de arveja, tales como antracnosis, fusarium, oídio y mildiu. No obstante, un número excesivo de épocas puede provocar sobreajuste, mientras que un número reducido puede resultar en un aprendizaje insuficiente.

Tamaño de lote (Batch Size): El tamaño de lote hace referencia a la cantidad de imágenes que el modelo procesa simultáneamente antes de actualizar los parámetros de aprendizaje. Este parámetro influye en la estabilidad y eficiencia del entrenamiento.

La selección de un tamaño de lote adecuado permite un equilibrio entre el consumo de recursos computacionales y la calidad del aprendizaje, especialmente en conjuntos de datos con clases desbalanceadas. En este estudio, un tamaño de lote reducido favorece un aprendizaje más estable y preciso, particularmente para aquellas clases con menor número de muestras.

Tasa de aprendizaje (Learning Rate): La tasa de aprendizaje determina la magnitud de los ajustes realizados por el modelo durante el proceso de entrenamiento cada vez que se detecta un error en la predicción. Este parámetro controla la velocidad con la que el modelo converge hacia una solución óptima.

Una tasa de aprendizaje elevada puede generar inestabilidad en el entrenamiento, mientras que una tasa demasiado baja puede ralentizar el proceso de aprendizaje. Por ello, se seleccionó una tasa de aprendizaje intermedia que garantiza un entrenamiento estable y eficiente, permitiendo al modelo mejorar progresivamente su capacidad de clasificación.

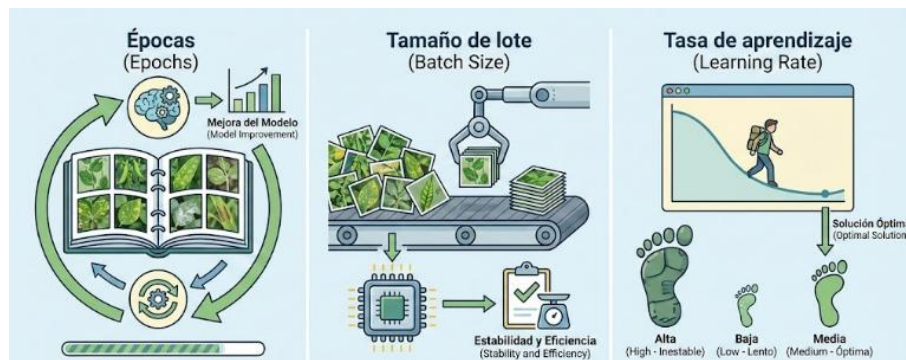


Figura 5. Épocas, tamaño de lote y tasa de aprendizaje

Pasos clave del entrenamiento

Definición de clases: Creas categorías para que el modelo pueda reconocer "puño cerrado" y "mano abierta".

Recolección de datos: Es importante proporcionar una cantidad suficiente y variada de ángulos, fondos, iluminación, etc; para que el modelo aprenda y evite el overfit.

Ajuste de parámetro: El número de veces que el modelo revisará y aprenderá del conjunto completo de datos de entrenamiento, las muestras que se procesan antes de actualizar el modelo interno y qué tan rápido el modelo responde a previos errores.

Inicio y monitoreo: Al presionar "Entrenar modelo" el modelo comienza. Una vez finalizado, se puede probar la precisión del modelo en la sección de "Vista previa".

2.2.6. Tecnologías de entrenamiento

2.2.6.1. Tecnologías en la nube Colab

Actualmente, el desarrollo de disciplinas relacionadas con el aprendizaje automático e inteligencia artificial está relacionado con el análisis de datos y el desarrollo de aplicaciones basados en IA, los cuales necesitan de grandes datasets para su ejecución y entrenamiento. Según (Fernández , 2022) Google Colab es una herramienta que facilita acceso universal y gratuito a recursos computacionales avanzados como GPU y TPU. Esta herramienta está exclusivamente especializada para desarrollar aplicaciones de inteligencia artificial, con análisis de datos. Puede considerarse como una evolución sofisticada de Jupyter Notebook (Fernández , 2022).

Una de las ventajas de Colab es que no se necesita configurar ni instalar ningún software, ya que todo se ejecuta en la nube, permitiendo así a los usuarios trabajar sin ningún inconveniente, optimizando tanto el acceso como la participación en proyectos compartidos. Además, permite una interfaz de cuaderno interactiva compatible con lenguajes como Python y R, permitiendo la integración de códigos ejecutables, contenido explicativo y elementos visuales en un solo documento (Bodero et al., 2020). Así mismo, Colab ofrece integración con otros servicios de Google, especialmente Google Drive, potenciando el almacenamiento eficiente y la colaboración en proyectos.

Las Ventajas de Google Colab son:

- **Colaboración sencilla:** Facilita el trabajo en equipo y el intercambio de documentos. Lo que genera una colaboración más fluida.
- **Integración versátil:** Se conecta con diversas herramientas y servicios.
- **Entorno preconfigurado:** Permite trabajar de inmediato acceso sin necesidad de configurar iniciales complejas, ya que viene con las bibliotecas instaladas de *data science*.
- **Accesos flexible a datos:** Permite manejar datos de múltiples orígenes y ofrece la posibilidad de cargar datasets desde fuentes externas o equipos locales.

Según menciona (Fernández , 2022) algunas restricciones de Google Colab son:

- **Limitación de almacenamiento:** restricción en el espacio de disco disponible, lo que puede ser un obstáculo significativo al trabajar con conjuntos de datos muy grandes.
- **Instalaciones no permanentes:** los paquetes y bibliotecas que se instalen son para la sesión de Colab. Es necesario reinstalar los paquetes en cualquier equipo nuevo para retomar el trabajo.
- **Foco en Python:** diseñado únicamente para Python.

2.2.6.2. API'S

La API son mecanismos que otorgan a dos componentes de software comunicarse por medio de definiciones y protocolos de forma segura y eficiente (Goodwin, 2024). Es decir, la API ayuda a definir qué información se puede solicitar, con que métodos-formatos y como debe ser la respuesta.

Según mencionan (Zambrano et al., 2025) para acceder a esta programación, el cliente debe enviar una solicitud a la API a una URL específica, lo que incluye definición del tipo de acción que se desea realizar (GET, POST, PUT/PATCH, DELETE) y detalles específicos para la solicitud, luego la API recibe la solicitud y se comunica con el servidor para ejecutar la acción solicitada. Finalmente, el servidor envía la información de vuelta a la API que la formatea, por lo general en JSON o XML y se la entrega a la aplicación cliente.

Ejemplos comunes de uso:

- **Mapas integrados:** ubicación exacta de un restaurante o un lugar que tenga una página web.
- **Iniciar sesión con Google/Facebook:** Un sitio web utiliza la API de estas redes sociales para autenticar el usuario en lugar de crear una nueva cuenta.
- **Clima:** A través de la API una aplicación móvil obtiene datos meteorológicos actualizados.

2.2.6.3. Gemini

Gemini es un nuevo modelo de inteligencia artificial (IA) generativa desarrollada por Google. Cabe mencionar que no es solo un chatbot, sino un conjunto de modelos multimodales, es decir que pueden operar, combinar y comprender sin problemas cualquier tipo de información como códigos, imágenes, texto, audios y videos (Diane & Stryker, 2024). Este asistente está diseñado para ayudar con tareas diarias, desde la productividad hasta la creatividad.

Gemini está construido sobre una arquitectura avanzada que combina algoritmos de aprendizaje y modelos de lenguaje de gran escala. Según (Tyagi, 2025) menciona que se basa en los siguientes principios:

- **Multimodalidad:** Gemini recibe, procesa y genera información utilizando textos, audios, imágenes y videos en conjunto. Por ejemplo, puedes mostrarle una foto y pedirle que te mencione una receta basada en los ingredientes que ve.
- **Entrenamiento y razonamiento:** Fue entrenado con datos de texto y código. Lo que le permite generar contenido coherente, realizar razonamiento lógico complejo y responder a preguntas de forma informativa.

- Integración: Puede interactuar con Gmail, Google Drive, Google Maps, Google Fotos y Google Workspace (Tyagi, 2025). Lo que permite realizar tareas cruzadas o acceder a tu información siempre con el permiso del usuario.
- Aprendizaje y adaptación: Aprende de cada conversación y se adapta a las demandas del usuario.

Algunos ejemplos de uso de Gemini en el campo de la computación se detallan en la tabla 3.

Tabla 3. Ejemplos de uso de Gemini en el campo de la computación.

| Área de la computación | Tarea específica | Beneficio principal |
|--------------------------|--|---|
| Desarrollo de software | Generación de código | Acelera la codificación de tareas complejas, acelerando su productividad del desarrollador. |
| Depuración | Identificación y corrección de errores | Reduce el tiempo en código extenso. |
| Análisis de código | Refactorización y optimización | Mejora el rendimiento y calidad del software. |
| Documentación | Generación automática de documentación | Ahorra tiempo al mantener la documentación actualizada y sincronizada con el código. |
| Seguridad informática | Análisis de vulnerabilidades | Ayuda a los desarrolladores a identificar y corregir fallos de seguridad. |
| Ingeniería de requisitos | Análisis de especificaciones | Facilita el proceso de aseguramiento de calidad del software. |

Fuente: (Diane & Stryker, 2024)

Gemini ofrece muchos beneficios para aumentar la productividad y potenciar la creatividad:

Automatización de tareas: genera resumen de documentos extensos hasta 1.500 páginas en versión Pro/Ultra, crea borradores de correos electrónicos o códigos de programación en segundos.

Organización: ayuda a planificar viajes combinando Google Maps y vuelos, estructurar información o resumir reuniones de Google Meet.

Integración de Apps: Accede a la información de tu Drive, Gmail y Calendar para realizar tareas sin tener que cambiar entre aplicaciones.

Generación de contenido: Crea guiones, idea de marketing, textos creativos y genera imágenes de alta calidad a partir de simples descripciones.

Asistencia en codificación: Proporciona ayuda con lenguajes de programación, análisis de repositorios extensos y sugerencias de código.

Edición multimedia: Generación de videos de alta calidad como clips de 8 segundos o restauración de fotografías.

Deep research: Puede navegar y analizar cientos de sitios web en tiempo real para crear informes.

Explicaciones complejas: permite hacer preguntas de seguimiento hasta que el concepto este claro y desglosa temas académicos o técnicos.

Análisis de datos: analiza grandes volúmenes de datos en documentos o archivos.

2.2.7. Lenguajes de programación

2.2.7.1. Python

Python es un lenguaje de programación que se centra en la legibilidad, lo que facilita su uso a quien posee conocimientos básicos de programación, tiene una serie de características y ventajas que lo hacen único y accesible. Según mencionan (Amaguaña et al., 2023). Para (Worsley, 2024) es una programación potente y versátil que impulsa una gran diversidad de soluciones tecnológicos; es clave en la innovación, desde el desarrollo de aplicaciones web y juegos hasta software especializado como el de animación y motores de búsqueda. Al ser un lenguaje gratuito y de código abierto, Python elimina la necesidad de adquirir licencias para su utilización. Además, cuenta con apoyo de una comunidad, lo que facilita el desarrollo de nuevas bibliotecas y aplicaciones.

Una de las características más destacables de Python es su naturaleza multiparadigma, que integra diferentes paradigmas de programación, lo que lo hace extremadamente fácil de aprender. Su aplicación no se limita a ámbitos específicos, sino a campos variados como el desarrollo web o la inteligencia artificial. Así mismo, se adapta a todas las plataformas, pudiendo ejecutarse en diferentes sistemas operativos como Windows o Linux (Universidad Europea, 2023).

La implementación estándar de Python, denominada CPython opera sin convertir el código fuente del programa directamente al lenguaje de máquina, que es el formato

que el hardware de la computadora puede procesar y ejecutar. Se desarrolla en tres fases principales:

1. El intérprete lee la instrucción o código Python, donde valida la sintaxis de cada línea. Si se destaca un error de sintaxis, el proceso se interrumpe de inmediato y se notifica en fallo.
2. Si la sintaxis es correcta, el intérprete traduce el código Python a un lenguaje intermedio llamado Bytecode. Una vez completado ese paso sin errores, el código original queda totalmente convertido a Bytecode (Worsley, 2024).
3. Finalmente, se transfiere a la máquina virtual Python, que se encarga de ejecutarlo. Si durante la ejecución se encuentra un error, el proceso se detiene y se emite un mensaje de error.

2.2.7.2. Java

Java es un lenguaje de programación basado en el paradigma de objetos con independencia de plataforma, que funciona también como plataforma independiente y está asociado con el desarrollo de aplicaciones o software. Estas nuevas tecnologías facilitan las tareas cotidianas, mejoran las comunicaciones y proporcionan entretenimiento. Según (Moreno, 2016) menciona que los programas creados con esta programación pueden realizarse en cualquier dispositivo electrónico para el desarrollo de sitios web, sistemas operativos, videojuegos o cualquier tipo de software.

2.2.8. Tipos de App

2.2.8.1. App móvil

Una aplicación móvil es un tipo de software diseñado para ejecutarse en dispositivos portátiles como tabletas o smartwatches que permiten al usuario realizar tareas concretas de cualquier tipo, facilitando las gestiones a desarrollar. Además, estas aplicaciones se clasifican en gratuitas y pagadas, donde el 20-30% del costo de la aplicación se destina al distribuidor y el resto al desarrollador (Silva et al., 2022).

Algunas características son:

- **Integración con el hardware:** Puede acceder y utilizar la cámara, el micrófono, el GPS, los sensores de movimientos, el lector de huellas dactilares y el sistema de notificaciones push del dispositivo (Palma et al., 2020).

- **Interfaz de usuario:** Está diseñada para ser usada con una pantalla táctil y con gestos sencillos.
- **Descarga e instalación:** Se distribuye a través de tiendas oficiales como donde el usuario la descarga e instala en la memoria del dispositivo.
- **Funcionamiento offline:** Muchas aplicaciones pueden funcionar total o parcialmente sin necesidad de una conexión a internet.

Las apps móviles se clasifican en tres tipos, tal y como se mencionan en la tabla 4.

Tabla 4. Tipos de apps móviles.

| Tipo de app | Descripción | Ventajas principales |
|-------------|--|---|
| Nativas | Desarrolladas para un sistema operativo (Kotlin/Swift para Android y iOS). | Mejor rendimiento y acceso completo a todas las funcionalidades del hardware. Menor costo y tiempo de desarrollo debido a que es una multiplataforma con un solo código base. |
| Híbridas | Construidas con tecnología web (CSS, HTML, JavaScript). | Son fáciles de actualizar y no requieren instalación ni aprobación de tiendas. |
| Web Apps | Sitios web que se ejecutan en el navegador móvil. | |

Fuente: (Palma et al., 2020).

2.2.8.2. App web

De acuerdo con (Mina, 2021) una aplicación web es un software accesible mediante un navegador como Safari, Chrome o Firefox y se ejecuta en un servidor externo en lugar de un dispositivo. Este tipo de aplicación depende de la conexión a internet para acceder a dichos servidores y obtener datos. Cabe mencionar que la información se mantiene de forma permanente gracias al almacenamiento en la nube.

Características de la app web

- **Acceso Universal:** Es necesario un navegador conectado a internet para acceder desde cualquier dispositivo.
- **Actualizaciones centralizadas:** Las actualizaciones se ejecutan y descargan automáticamente en el servidor.
- **Tecnologías estándar:** Utilizan HTML (escritura), JavaScript (funcionalidad) y CSS (estilos).
- **Almacenamiento en servidor:** Los datos del usuario se gestionan y almacenan en el servidor, no en el dispositivo.
- **Interacción Cliente-Servidor:** La parte que se mira y se usa interactúa constantemente con el servidor.

Algunos ejemplos de aplicaciones web son servicios de correo electrónicos, plataformas de redes sociales, tiendas en línea, banca en línea, herramientas de productividad y almacenamiento en la nube.

Una aplicación web es importante distinguirla de las aplicaciones nativas, ya que estas son programadas para un solo sistema operativo o plataforma, los cuales se deben descargar e instalar desde una tienda de aplicaciones como App Store o Google Play. Tal y como se indica en la tabla 5.

Tabla 5. Características de las App

| Característica | Aplicación web | Aplicación nativa |
|-----------------------|--------------------------------------|-----------------------------------|
| Acceso | Navegador web | Descarga e instalación |
| Actualizaciones | Automáticas | El usuario debe actualizarse |
| Uso de Hardware | Limitado | Acceso total de cámara. GPS, etc. |
| Plataforma | Independiente del sistema operativo. | Específica para Android y iOS. |

Fuente: (Llenera & Lombardo, 2021)

2.2.8.3. App de escritorio

Una aplicación de escritorio es un software diseñado para ser instalado y ejecutado directamente en el sistema operativo de una PC personal como Windows, Linux o macOS (Fuentes et al., 2016). Es decir, se instala en el sistema a través de un instalador especial y utiliza recursos para funcionar.

Algunas características son:

- **Instalación local:** El software se instala en el disco duro de la computadora (Fuentes et al., 2016).
- **Ejecución directa:** Se ejecuta utilizando los recursos del propio dispositivo en el que está instalado.
- **Independencia:** Cualquier aplicación de escritorio puede funcionar sin una conexión de internet.
- **Acceso a recursos del sistema:** Accede y gestiona archivos, carpetas, impresoras y hardware conectado.
- **Interfaces ricas:** Ofrece interfaces con mejor rendimiento que las aplicaciones web.

Según mencionan (Sampedro et al., 2016) las app de escritorio abarcan una amplia gama de software como productividad (Adobe Photoshop, Microsoft Word, Microsoft Excel), navegadores web (Google Chrome, Safari, Mozilla Firefox), comunicación (los

clientes de escritorio), juegos y utilidades del sistema (herramientas de compresión y antivirus). Por otro lado, es necesario diferenciarlas de las apps web, que son programas que ejecutan dentro de un navegador y dependen de un servidor y de internet para funcionar. De este modo, en la tabla 6 se detallan algunas características de las aplicaciones de escritorio y las aplicaciones web.

Tabla 6. Aplicaciones de escritorio vs aplicaciones web.

| Característica | Aplicación de escritorio | Aplicación web |
|-----------------------|---|---|
| Instalación | (Sí) En el dispositivo local. | (No) requiere del uso de un navegador. |
| Internet | Depende del programa. | Esencial para funcionar. |
| Recursos | Utiliza los recursos del dispositivo local. | Utiliza los recursos del servidor. |
| Actualización | El sistema o el usuario deben instalar parches. | Es automática, ya que el desarrollador las gestiona en el servidor. |

Fuente: (Fuentes et al., 2016)

2.2.9. Entorno de desarrollo

2.2.9.1. Visual Studio Code

Según (Sánchez J. , 2023) Visual Studio Code es un editor de códigos desarrollado por Microsoft, disponible para Linux, Windows y macOS. Se caracterizan por su eficiencia y ligereza. Permitiendo visualizar, editar, ejecutar y depurar el código fuente de aplicaciones con gran eficiencia, particularmente reconocido entre desarrolladores front-end.

Este editor gratuito y de código abierto ha ganado popularidad en la comunidad de programación debido a sus numerosas ventajas y características, una de ellas es su alto grado de personalización, donde los usuarios pueden instalar diferentes temas para adaptar su entorno de desarrollo según sus necesidades y preferencias. Esto incluye lenguaje de programación, herramientas de depuración integradas, control de versiones y compatibilidad con herramientas de desarrollo (Cimas, 2022).

Por otro lado, VS Code ofrece una interfaz de usuarios por su diseño intuitivo y accesible, permitiendo trabajar con múltiples archivos y carpetas simultáneamente en una única ventana, facilitando la organización y navegación entre proyectos. Además, ofrece herramientas y servicios populares tales como: Docker, Git y Azure, permitiendo a los desarrolladores que trabajen con mayor eficiencia entre miembros del equipo.

Mientras que, en términos de rendimiento, Visual Studio Code fue diseñado para mantener su velocidad y eficiencia incluso para proyectos grandes.

2.2.9.2. Flutter

Flutter es el kit de desarrollo de software (SDK) de Google destinado a construir aplicaciones compatibles tanto para Android como para iOS (Apple). Fue desarrollado únicamente como herramienta de uso interno para la compañía. Sin embargo, Google reconoció su potencial y decidió transformarlo en un proyecto de código abierto accesible para la comunidad de programadores.

Según manifiestan (Toapanta et al., 2022) en la actualidad, Flutter se ha posicionado como una de las plataformas de aplicaciones móviles con mayor crecimiento y aceptación del mercado, debido a que permite la creación de aplicaciones web y de escritorio para sistemas Windows y Mac.

2.2.9.3. TensorFlow

Es una plataforma de software de código abierto desarrollada por Google que ha revolucionado los campos del aprendizaje automático y la inteligencia artificial. Esta herramienta facilita la creación y el entrenamiento de modelos de aprendizaje gracias a un conjunto completo de recursos. La característica más importante de TensorFlow es la capacidad para trabajar con redes neuronales y realizar cálculos paralelos mediante unidades de procesamiento gráfico (GPU) y unidades de procesamiento tensorial (TPU). También, ofrece diferentes funcionalidades, como optimización de modelos autónomos, gráficos computacionales, gestión de datos y visualización de resultados (Innovación, 2024).

Así mismo, ofrece (API) que permite a los desarrolladores operar desde capas básicas hasta estructuras de alto nivel. En conclusión, esta herramienta representa una solución para el desarrollo, debido a que proporciona recursos y herramientas para crear modelos eficientes, flexibles y de gran funcionabilidad.

2.2.10. Gestores de bases de datos

Un Sistema de Gestión de Base de Datos es un conjunto integrado de datos interrelacionados, organizados mediante programas informáticos que permiten acceder a ellos de manera estructurada. Es decir, contribuyen conjuntos de software especializados que hacen posible la administración y gestión eficiente de los datos almacenados. Según (Darias, 2021) los usuarios pueden generar y editar datos,

acceder a las operaciones basadas en el conocimiento de la organización, generar informes y realizar búsquedas.

Por otro lado, la función de un SGBD es permitir el acceso a la base de datos tanto a los usuarios finales como a las aplicaciones que requieran interactuar con la información almacenada.

En la tabla 7 se identifican las ventajas, características e inconvenientes de los gestores de base de datos.

Tabla 7. Gestores de base de datos

| | MYSQL | POSTGRESQL | ORACLE | SQL SERVER/EXPRESS | FIREBASE |
|-----------------------------|--|--|---|--|---|
| Ventajas | Amplia adopción y popularidad | Soporte de tipos de datos avanzados | Amplia funcionalidad y características avanzadas. | Integración con el ecosistema de Microsoft | Desarrollo rápido y al instante. |
| Características principales | Ampliamente utilizado y de código abierto. | Base de datos avanzada y de clase empresarial | Soporte de datos avanzados y rendimiento optimizado | Herramientas de desarrollo y administración robustas | Base de datos NoSQL, accesible vía cloud, orientada al soporte de aplicaciones web y móviles. |
| Inconvenientes | Limitaciones en la gestión de transacciones y concurrencia | Menor adopción y popularidad en comparación con otros sistemas | Costoso y mayor complejidad en la administración. | Limitaciones en la versión Express, como restricciones de tamaño y recursos. | Dependencia del ecosistema de Google. |

Fuente: (Darias, 2021)

2.2.11. Base de datos relacional- no relacional

Las bases de datos relacional y las no relacional son modelos fundamentales distintos para almacenar y gestionar datos (Rendón, 2019).

Base de datos relacional: Agrupa los datos en tablas que consisten en filas y columnas, su característica es el uso de claves para definir las relaciones entre las tablas, lo que permite reducir la redundancia (Chingo & López, 2021). Ejemplos como PostgreSQL, Oracle, MySQL y Microsoft SQL Server.

Base de datos no relacional: Las bases de datos relacionales presentan dificultad para gestionar datos no estructurados, una limitación notable de aplicaciones web actuales y en contextos de Big Data.

Tabla 8. Diferencias de las bases de datos relacionales y las no relacionales.

| Características | Base de datos relacional | Base de datos no relacional |
|------------------------|---|---|
| Modelo de datos | Tabular tablas con filas y columnas. | Documentos, grafos y columna ancha. |
| Relaciones | Basadas en claves primarias y foráneas. | No existe un mecanismo de unión estricto. |
| Lenguaje de consulta | SQL | Varía según el modelo y el sistema. |
| Consistencia | Fuerte (ACID) | Eventual (BASE) |
| Escalabilidad | Vertical | Horizontal |

Fuente: (Rendón, 2019)

III. METODOLOGÍA

3.1. ENFOQUE METODOLÓGICO

3.1.1. Enfoque

El enfoque mixto se caracteriza por la integración de datos tanto cualitativos como cuantitativos dentro de una misma investigación, buscando dar respuesta al problema planteado (Sampieri, 2014). Este enfoque reconoce la sinergia y la complementariedad entre ambos tipos, lo que permite un entendimiento profundo de lo que se está investigando.

El estudio empleó un enfoque mixto; cuantitativo al obtener información estadística relacionada con la diversidad de enfermedades en el cultivo de arveja y cualitativo mediante recopilación de datos por medio de encuestas.

3.1.2. Tipo de Investigación

3.1.2.1. Investigación descriptiva

Según mencionan (Guevara et al., 2020) los estudios descriptivos se caracterizan por identificar atributos, cualidades y perfiles específicos de colectivos, individuos, procesos, objetos o cualquier entidad sometida a investigación. Además, se destaca por la recopilación de información y por el reconocimiento de variables.

Este proyecto se enmarca con una investigación descriptiva, define las propiedades de la visión artificial y su uso en la industria agrícola. Así mismo, establece criterios edafoclimáticos que propician la detección de enfermedades en el cultivo de guisantes. Además, implica la recopilación de datos, resultado de un extenso análisis y comprensión de las zonas de estudio del sector agrícola y los métodos empleados en el cultivo de arveja.

3.1.2.2. Investigación explicativa

Según (Guevara et al., 2020) investiga un fenómeno específico que no ha sido indagado a fondo y así proporcionar conocimiento sobre ellos.

Esta investigación adopta un enfoque explicativo, ya que examina e interpreta los fenómenos presentes en los cultivos de arveja. Se detallan las condiciones para la revelación de las enfermedades más comunes en este cultivo. Así mismo, se destaca el papel de la visión artificial como herramienta tecnológica para el ámbito agrícola.

3.1.2.3. Investigación de campo

De acuerdo con (Rus, 2020) la investigación de campo se utiliza para reunir datos directamente desde el área de estudio, lo que facilita obtener información precisa sobre un problema específico. Esta metodología puede combinarse con otros enfoques investigativos, tales como los estudios correlacionales, exploratorios o mixtos.

Por lo tanto, el presente estudio utiliza este enfoque, ya que permite tener contacto directo con los agricultores y la comunidad para obtener datos e información clara. Esto se logra mediante el uso de entrevistas, encuestas y análisis de datos obtenidos durante el proceso.

3.2. IDEA A DEFENDER

El uso de visión por computador permitirá ejecutar acciones correctivas y preventivas con el objetivo de eliminar la aparición de enfermedades en el cultivo de la arveja.

3.3. DEFINICIÓN Y OPERACIONALIZACIÓN DE LAS VARIABLES

3.3.1. Definición de variables

Variable independiente: Visión por computador

La visión por computadora se ha establecido como una herramienta eficaz en el sector agrícola, por lo que se están desarrollando indicadores para evaluar su rendimiento, incluyendo aquellos relacionados con el entrenamiento algorítmico, que se mide a través del tiempo empleado para aprender, el volumen de información y la calidad de la información obtenida. Así mismo, el tipo de algoritmo determina el coste de ejecución, precisión, eficiencia e interpretación de resultados.

De este modo, el componente fundamental de estos sistemas es el procesamiento de imágenes, evaluando meticulosamente cada paso de las figuras capturadas.

Variable dependiente: Detección de enfermedades en el cultivo de arveja

La detección de patologías en el cultivo de arveja se fundamenta en tres partes principales: la caracterización de la enfermedad, la descripción de los tubérculos y los factores climáticos que propician el desarrollo de la enfermedad.

3.3.2. Operacionalización de variables

En la tabla 9 se indican la definición y la operacionalización de variables, tomando en cuenta las variables independientes y dependientes. Además, se mencionan dimensiones, indicadores, técnicas e instrumentos empleados durante el proceso.

Tabla 9. Operacionalización de variables

| Variable independiente | Dimensión | Indicadores | Técnica | Instrumento |
|---|-------------------------------------|--|------------------------|--------------------|
| | Aprendizaje del algoritmo | Tiempo de estudio Cantidad de información Calidad de la información | | |
| Visión por computador | Algoritmo por estudiar | Costo de implementación La precisión del algoritmo Eficiencia de respuesta | Entrevista Encuesta | Guía de preguntas |
| | Proceso de imágenes | Interpretación de resultados Etapas de procesamiento | | |
| Variable dependiente | Dimensión | Indicador | Técnica | Instrumento |
| | Características de las enfermedades | Porte color Cantidad de manchas en hojas y tubérculos. Ciclo de la enfermedad | | |
| Detección de enfermedades en el cultivo de arveja | Características del tubérculo | Porte cifra Ciclo de cosecha Temperatura del suelo Temperatura del ambiente Humedad del piso | Entrevista Encuesta | Guía de preguntas |
| | Condiciones agroclimáticas | | | |

3.4. MÉTODOS UTILIZADOS

3.4.1. Observación directa

Este componente es fundamental en cualquier investigación, ya que facilita la captura máxima de datos sobre los procedimientos internos de una organización y permite determinar la tecnología informática necesaria para construir la aplicación. Los resultados se obtienen a partir de un análisis basado en la observación directa del entorno que se está estudiando.

3.4.2. Métodos transversales

(Huairé, 2019) define el método transversal como aquel que recopila datos en un único punto temporal. Su finalidad es describir las variables y examinar como influyen entre sí en ese momento específico.

Esta investigación utilizó este método, ya que permitió analizar las dos variables en un solo tiempo, como también obtener resultados sobre la implementación del sistema de detección de enfermedades en el cultivo de arveja.

3.4.3. Técnicas

Las técnicas empleadas en este proyecto fueron la encuesta y la entrevista, debido a que ayudaron a obtener información relevante de la detección y las diferentes patologías dentro del proceso del cultivo de arveja (*Pisum Sativum.L*).

La estrategia empleada consistió en aplicar una encuesta que fue dirigida a expertos en el cultivo de arveja de las distintas comunidades de San Gabriel. Buscando obtener información a profundidad sobre las posibles enfermedades que surgen durante el proceso de siembra. Por otro lado, la entrevista con el propósito de adquirir un entendimiento de los procesos aplicados en el cultivo de arveja.

3.5. ANÁLISIS ESTADÍSTICO

El método estadístico empleado fue el análisis descriptivo, con la finalidad de detallar un conjunto de datos, para obtener información clara y concisa. Así mismo, permite conocer a detalle la información de la población por medio de la recopilación de datos.

IV. RESULTADOS Y DISCUSIÓN

4.1. RESULTADOS

4.1.1. Análisis de la entrevista

1. ¿Está familiarizado con el significado de "TIC"? ¿Estaría dispuesto a utilizarlo en su cultivo?

Respuesta: Sí, como ingeniero agrónomo, estoy familiarizado con el significado de las TIC (Tecnologías de la Información y Comunicaciones) y estoy completamente dispuesto a utilizarlas en mi cultivo de arvejas. Las TIC pueden proporcionar herramientas y soluciones innovadoras para mejorar la gestión y el rendimiento de los cultivos.

2. ¿Tiene conocimiento sobre algún modelo experimental que pueda detectar tempranamente enfermedades en el cultivo de guisantes?

Respuesta: Tengo conocimiento sobre diversos modelos experimentales que pueden detectar tempranamente enfermedades en el cultivo de guisantes. Estos modelos suelen basarse en técnicas de monitoreo continuo, análisis de datos y detección temprana de patógenos para prevenir y controlar enfermedades de manera eficaz.

3. ¿Cuáles son las condiciones que, en su opinión, favorecen el desarrollo de la enfermedad oídio en el cultivo de guisantes?

Respuesta: En mi opinión como ingeniero agrónomo, los factores que propician la enfermedad del Oídio en el cultivo de guisantes incluyen la alta humedad ambiental, temperaturas moderadas o cálidas, densidad de siembra elevada y falta de ventilación adecuada que propicie la propagación del hongo causante del oídio.

4. ¿Considera que la implementación de las TIC es crucial para tomar medidas correctivas dentro del cultivo?

Respuesta: Considero que la implementación de las TIC es crucial para tomar medidas correctivas dentro del cultivo de arvejas. Las TIC permiten una gestión más eficiente, monitoreo en tiempo real, análisis de datos precisos y toma de decisiones informadas para controlar enfermedades y optimizar la producción.

5. ¿Opina que la integración de las TIC en el cultivo de arveja podría mejorar la calidad y el valor nutricional de dicha legumbre?

Respuesta: En mi opinión, la integración de las TIC en el cultivo de arveja tiene el potencial de mejorar significativamente la calidad y el valor nutricional de la legumbre. Las TIC pueden contribuir a una gestión más precisa, detección temprana de enfermedades, optimización de recursos y seguimiento de prácticas agronómicas.

6. ¿Cree que el uso de la tecnología permite ahorrar recursos humanos?

Respuesta: Sí, creo firmemente que el uso de la tecnología en la agricultura permite ahorrar recursos humanos al automatizar procesos, optimizar la producción y mejorar la eficiencia en la gestión de los cultivos de arvejas. La tecnología puede liberar tiempo y recursos para actividades más estratégicas y de alto valor.

7. ¿Aprueba la introducción de un modelo experimental que detecte tempranamente la enfermedad oídio en el cultivo de guisantes?

Respuesta: Apruebo totalmente la introducción de un modelo experimental que pueda detectar tempranamente la enfermedad del oídio en el cultivo de guisantes. La detección temprana es clave para la prevención y control de enfermedades, y la implementación de modelos experimentales puede mejorar la salud de los cultivos y la productividad agrícola.

4.1.2. Análisis de la encuesta

Gracias a las 7 encuestas realizadas a los agricultores, a continuación, se presenta el análisis respectivo.

1. Género

- Hombres: 6
- Mujeres: 1

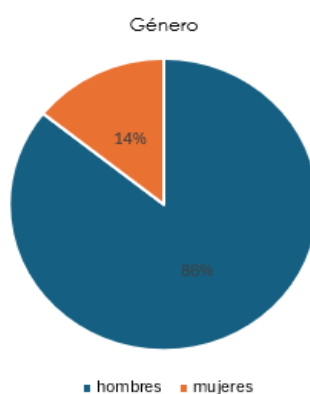


Figura 6. Género

Análisis: Se encuestó a siete personas, 6 hombres y 1 mujer. Las personas que más siembran arveja son de género masculino. Tal y como lo representa la figura, un 86% hombres y 14% mujeres.

2. ¿Considera que la implementación de un prototipo sería útil para detectar el oídio (manchas blancas en hojas, tallos, ramas, vainas y semillas) a tiempo y preservar la calidad y el valor nutricional de la leguminosa?

- De acuerdo: 7
- Neutral: 0
- En desacuerdo: 0

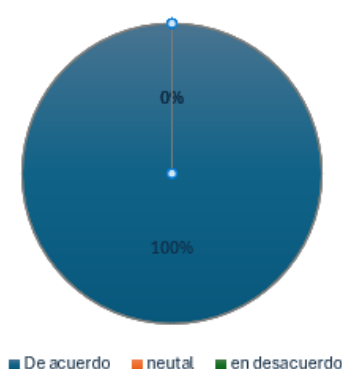


Figura 7. Implementación de un prototipo

Análisis: El 100% de los participantes están “de acuerdo” con la implementación de un prototipo para detectar el oídio en las plantas de leguminosas, ya que podría ser muy útil para identificar esta enfermedad a tiempo. Esto permitirá administrar mejor la afección y salvaguardar las siembras de leguminosas.

3. ¿Qué tipo de método de detección de esta enfermedad (oídio) utiliza en su cultivo de guisantes?

- Automatizado: 0
- Manual: 6
- Ambos: 1

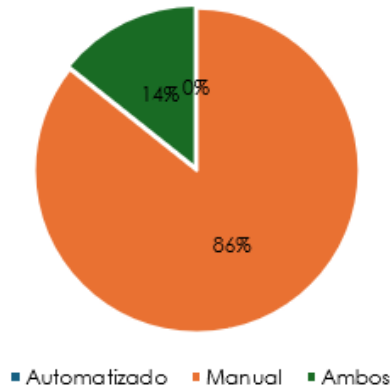


Figura 8. Tipo de método

Análisis: El 86 % de los encuestados se basa únicamente en la inspección visual o manual para detectar el oídio. Además, el 14 % utiliza una combinación de métodos, por otro lado, ningún agricultor emplea un sistema automatizado. Es importante contar con métodos efectivos de detección para tomar medidas preventivas o de control a tiempo, para proteger la calidad y el rendimiento de la arveja.

4. ¿Está de acuerdo en que el uso de la tecnología ahorra recursos humanos, especialmente en el tiempo dedicado al cuidado del cultivo de guisantes?
- De acuerdo: 7
 - Neutral: 0
 - En desacuerdo: 0

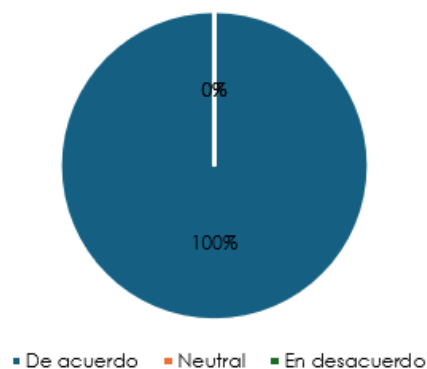


Figura 9. Uso de tecnología

Análisis: El 100 % de los participantes está “de acuerdo” con la implementación de esta tecnología eficaz para optimizar y automatizar ciertas tareas relacionadas con el cultivo, lo que podría liberar tiempo y recursos humanos para otras actividades. El uso de tecnología en la agricultura puede aumentar la productividad, eficiencia y

facilitar el manejo de los sembríos, lo que influye significativamente en un impacto positivo en la gestión y cuidado de los sembríos de guisantes.

5. ¿Cuáles considera que son los factores que influyen en el desarrollo de la enfermedad oídio en el cultivo de guisantes?

- Temperatura: 1
- Fertilizantes: 1
- Luz difusa: 0
- Agua: 0
- Humedad: 1
- Resistencia a fungicidas: 3
- Ninguna: 1

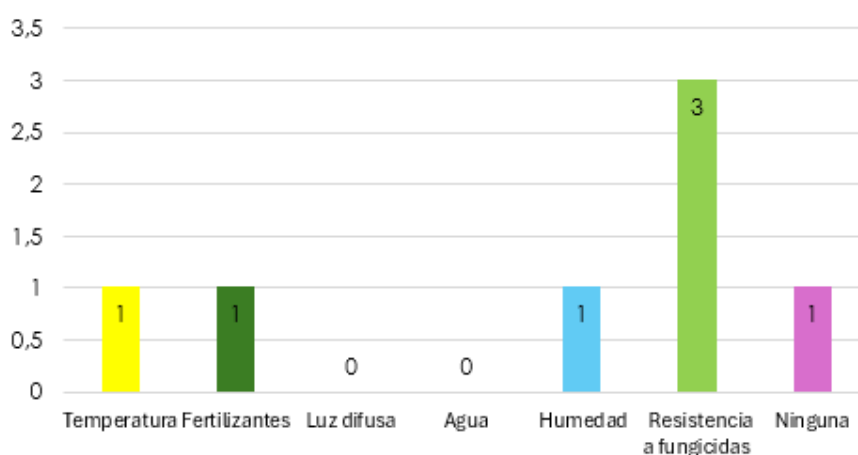


Figura 10. Factores que influyen en la enfermedad oídio

Análisis: El factor de más respuesta como influencia en el desarrollo del oídio en el cultivo de guisantes fue la "resistencia a fungicidas", con 3 respuestas. Lo que se considera un factor importante en la prevención o control de las patologías de los cultivos de guisantes.

6. ¿En qué medida estima que se desarrolla el oídio?

- Del 0 % al 25 %: 1
- Del 25 % al 50 %: 6
- Del 50 % al 75 %: 0
- Del 75 % al 100 %: 0

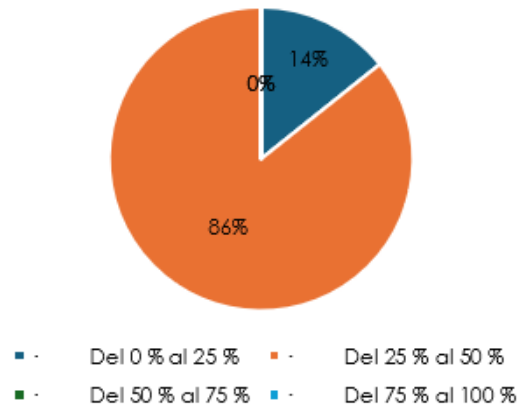


Figura 11. Desarrollo del oídio

Análisis: Según la estimación de los participantes, reportaron que el desarrollo de este hongo en el cultivo de guisantes se encuentra entre el 25% y el 50%. Lo cual sugiere un bajo grado de desarrollo de la enfermedad en los cultivos de esta leguminosa.

7. ¿Qué beneficios cree que aportaría la introducción de un prototipo que detecte tempranamente la enfermedad oídio en el cultivo de guisantes?

- Calidad del tubérculo: 2
- Valor nutricional: 1
- Ahorro de tiempo: 2
- Ahorro de dinero: 2

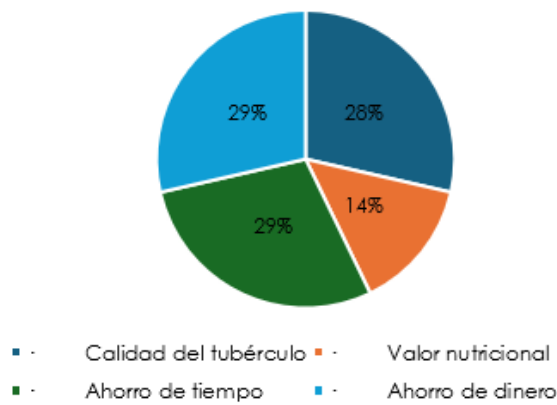


Figura 12. Beneficios de la introducción de un prototipo

Análisis: Se destaca que los beneficios más mencionados que se esperarían al introducir un prototipo para detectar tempranamente la enfermedad del oídio en el cultivo de guisantes son: Calidad del tubérculo (29 %), ahorro de tiempo (29 %) y valor

nutricional (14 %). Dichas ventajas pueden contribuir en el manejo eficiente de esta patología y el cuidado de los cultivos de esta leguminosa.

8. ¿Cuáles considera que son las características de la enfermedad oídio presente en el cultivo de guisantes?

- Putrefacción: 1
- Formación de manchas de color marrón: 6
- Formación de borde amarillo: 0
- Generación de plagas: 0

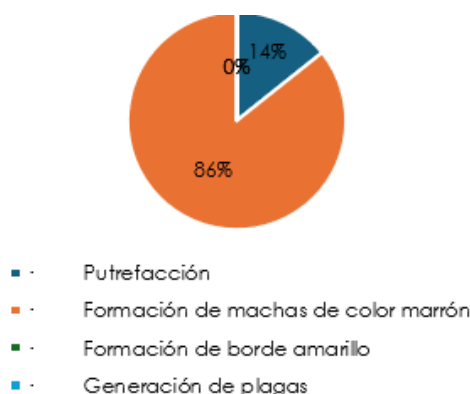


Figura 13. Características de la enfermedad oídio

Análisis: La característica más mencionada de la enfermedad del Oídio presente en el cultivo de guisantes es la "formación de manchas de color marrón" con el 86%. Lo que se puede interpretar es que la putrefacción es común en los sembríos de los encuestados.

Análisis general

La encuesta afirma que existe un alto interés por las soluciones tecnológicas que ayuden a detectar el oídio en los cultivos de arveja de manera temprana. Los agricultores valoran estas herramientas para salvaguardar la calidad y valor nutricional del producto. La resistencia a fungicidas es el factor más influyente en el manejo de esta enfermedad. De este modo, la implementación de nuevas estrategias como la detección de prototipos se ha convertido en una necesidad para los agricultores.

4.1.3. Análisis de resultados de la Aplicación "ARVEJA CHECK"

Para validar la efectividad del sistema **ARVEJA CHECK** en la identificación de patologías del cultivo de *Pisum Sativum L.* (Antracnosis, Fusarium, Oídio y Mildiu), se

llevaron a cabo pruebas de detección sobre una muestra positiva para **Antracnosis**. El objetivo principal fue determinar cómo las condiciones de iluminación y el horario de captura influyen en el porcentaje de confianza del algoritmo.

Los resultados obtenidos demuestran una variación significativa en la precisión dependiendo de la fuente de luz:

1. **Detección con luz natural (Condición Óptima):** El rendimiento máximo de la aplicación. Bajo condiciones de luz natural diurna, el sistema logró una **confianza del 100.0%**. Esto confirma que la iluminación solar difusa es el escenario ideal para que el modelo extraiga correctamente las características de la enfermedad.



Figura 14. Detección con luz natural

2. **Detección con iluminación artificial:** Ante la ausencia de luz natural, el sistema mantiene una alta fiabilidad si se utilizan fuentes externas:
 - El uso de **luz RGB blanca** permitió una confianza del **98.3%**.



Figura 15. Detección con iluminación artificial

- El uso del **flash del dispositivo móvil** resultó en una confianza del **96.5%**.



Figura 16. Flash del dispositivo móvil

3. **Detección con baja luminosidad:** La prueba realizada al atardecer (18:31 horas) sin apoyo lumínico presentó el índice más bajo, con un **93.8%** de confianza. Aunque el diagnóstico sigue siendo correcto, la falta de luz reduce la certeza del modelo.



Figura 17. Detección con baja luminosidad

Conclusión del Análisis: Se determina que la aplicación **ARVEJA CHECK** ofrece sus resultados más precisos (**100%**) cuando se utiliza durante el día (horario vespertino con luz natural). En caso de realizar monitoreos nocturnos o en condiciones de baja luz, es imperativo el uso de iluminación auxiliar (flash o luz blanca) para garantizar una confianza superior al 95%.



Figura 18. Iluminación de la detección de enfermedades con "Arveja Check"

4.1.4. Propuesta

Los errores de pantalla azul o BSOD suceden cuando el Windows está en un error crítico, lo que indica que ha ocurrido un fallo en el sistema y no puede recuperarse

de forma segura. Sin embargo, puede ser causada por diferentes problemas: cuando el controlador de hardware está desactualizado, virus que pueden corromper archivos importantes del sistema o archivos de Windows que se han dañado.

A su vez, es causada por la falta de memoria RAM y disco mecánico, ocasionando que el servidor se haga lento y no pueda ejecutar las actividades. De este modo, el algoritmo YOLO debe ser instalado en una computadora de alta calidad para no ocasionar problemas en el servidor.

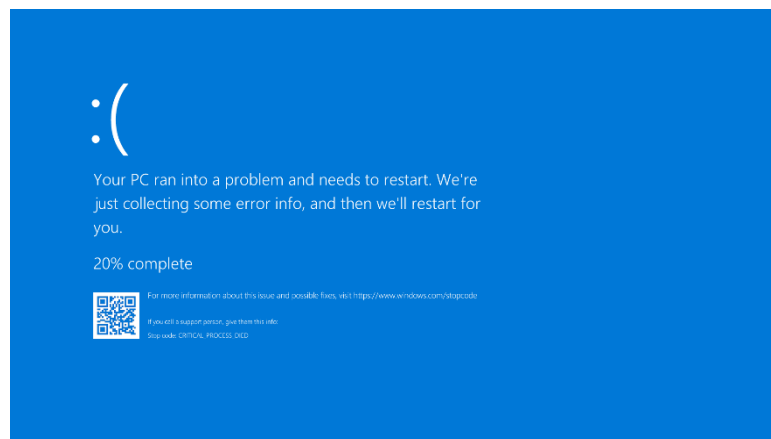


Figura 19. Errores de pantalla azul
Fuente: (Castro, 2025)

Debido al enfoque interactivo orientado al usuario, se utilizaron métodos Agile Extreme Programming (XP) para guiar el desarrollo del sistema, permitiendo una rápida adaptación a las necesidades y una colaboración con los agricultores. Además, se emplearon métodos de análisis de imágenes y Machine Learning, por medio de este sistema se analizó las imágenes de los cultivos de arveja y se detectaron patrones indicativos de las enfermedades comunes. Permitiendo que los agricultores tomen las medidas necesarias ante esta situación.

El enfoque XP garantizó un desarrollo eficiente del sistema, por lo que fue adecuada la unión del equipo de desarrollo y los expertos en el manejo de la arveja. Por medio, de un ciclo interactivo de planificación, codificación, pruebas y retroalimentación, se aseguró que el sistema cumpla con las necesidades de los usuarios. Además, este sistema permitió incorporar eficazmente comentarios y sugerencias de los agricultores, asegurando que el prototipo final se convierta en una solución práctica

y eficaz para la detección temprana de enfermedades en el cultivo de *Pisum Sativum* L.

4.1.4.1. Arquitectura

Plan de proyecto

- Historias de usuarios
- Tareas iterativas
- Planificación de versiones

Diseño

- Diseño simple: Hacer que el diseño del sistema sea lo más simple posible, centrándose únicamente en las funciones necesarias.
- Diseño incremental: Desarrollar iterativamente el sistema, añadiendo nuevas características en cada iteración.
- Metáfora del sistema: Emplear analogías o metáforas que faciliten la comprensión de la estructura y funcionamiento de los sistemas de visión por computadora.

Codificación

- Programación en pareja: Asigne dos programadores para que ejecuten cada tarea juntos.
- Propiedad colectiva del código: Permite que todo el equipo modifique cualquier segmento del código, lo cual promueve el trabajo en equipo.
- Estándares de codificación: Establecer y seguir pautas de codificación para mantener la calidad y legibilidad del código.

Pruebas

- Pruebas unitarias: Realizar pruebas automatizadas de cada componente individual del sistema de visión por computadora.
- Pruebas de aceptación: Examinar que el sistema cumpla con los requisitos de los agricultores y expertos en cultivo de arveja.

Implementación

- Entrega frecuente: Entregar periódicamente versiones funcionales del sistema de visión por computadora a los usuarios.

- 5. Clientes in situ: Involucrar a los agricultores y expertos en cultivo de guisantes en el proceso de desarrollo.
- Integración continua: Integrar de forma automática y frecuente los cambios realizados por el equipo.

Seguimiento

- Velocidad del proyecto: Medir la productividad del equipo para planificar y ajustar la velocidad del desarrollo.
- Revisión de la iteración: Reflexionar sobre los éxitos, los desafíos y las áreas de mejora después de cada iteración.
- Mejora continua: Implementar las lecciones aprendidas de la revisión para mejorar el proceso de desarrollo e iteraciones posteriores.

4.1.4.2. Fase de factibilidad

El modelo de detección se desarrolló utilizando tecnologías TensorFlow, Django y Python, ya que disponen de licencias de uso libre, lo que puede resultar valioso en la investigación actual. Por otro lado, MySQL se utilizó como sistema de base de datos, ya que permite acceder y recuperar datos en cualquier momento sin comprometer la integridad de los datos.

Recursos de software

En la tabla 10, se describen los diferentes recursos de software existentes, además de la cantidad de cada uno de estos.

Tabla 10. Recursos de software

| Tipo de recurso | Nombre del recurso | Descripción | Cantidad |
|-----------------|--------------------|---------------------------------|----------|
| Software | TensorFlow | Sistema de aprendizaje autónomo | 1 |
| | Flutter | SDK de desarrollo | 1 |
| | Pytho | Lenguaje de programación | 1 |
| | N Dart | Lenguaje de programación | 1 |
| | GitHub | Sistema de control de versiones | 1 |

El progreso de la investigación demanda la presencia de varios equipos necesarios para su ejecución, como una conexión a internet para la interacción entre los equipos de trabajo y acceso a los recursos de internet.

Recursos de hardware

En la tabla 11, se detallan los diferentes recursos de hardware existentes, además de la cantidad de cada uno de estos.

Tabla 11. Recursos de hardware

| Tipo de recurso | Nombre del recurso | Descripción | Cantidad |
|-----------------|-----------------------|--|----------|
| Hardware | Equipo de computación | Dell Laptop Inspiron 7570, FHD táctil de 15.6 pulgadas, Intel Core i7-8550U, 8 GB DDR4, 1 TB SSHD, cámara IR, Nvidia GeForce 940MX | 1 |
| | Dispositivo móvil | Dispositivo móvil Android | 1 |

4.1.4.3. Factibilidad económica

El diseño del proyecto contempló diversos recursos, detallados en la tabla 12, incluyendo equipamiento físico (hardware), programas informáticos (software) y personal cualificado (talento humano) y equipamiento de oficina.

Tabla 12. Factibilidad económica

| Descripción | Cantidad | Costo unitario | Costo total |
|-------------------------------------|----------|-------------------|------------------|
| Costos de hardware | | | |
| Laptop | 1 | \$1200,00 | \$1200,00 |
| Dispositivo móvil | 1 | \$300,00 | \$300,00 |
| Total, de hardware | | | 1500,00 |
| Costos de software | | | |
| Python | | 00,00 | 00,00 |
| TensorFlow | | 00,00 | 00,00 |
| Flutter | | 00,00 | 00,00 |
| GitHub | | 00,00 | 00,00 |
| Dart | | 00,00 | 00,00 |
| Django | | 00,00 | 00,00 |
| Servidor | | \$30,00 x 3 meses | 00,00 |
| Total, Software | | \$30,00 | \$90,00 |
| Talento Humano | | | |
| Programador | | 2500,00 | 5000,00 |
| Total, Talento Humano | | | \$5000,00 |
| Material de Oficina | | | |
| Internet | 1 | \$30 x 6 meses | 180,00 |
| Material de oficina | 1 | \$100 | 100,00 |
| Otros | 1 | 00,00 | 100,00 |
| Total, materiales de oficina | | | \$380,00 |
| Servicios | | | |
| Luz | | \$20 x 6 meses | \$120,00 |
| Transporte | | \$10 x 6 meses | \$60,00 |
| Total, Servicios | | | \$180,00 |
| Total | | | \$7150,00 |

4.1.4.4. Factibilidad operativa, situación actual

Los agricultores de las diferentes comunidades de San Gabriel detectan enfermedades en sus cultivos de arvejas utilizando conocimientos tradicionales o confiando en el asesoramiento técnico de expertos. A lo que conlleva, una

detección incorrecta de las enfermedades más habituales o un retraso en la toma de decisiones por dependencia de factores externos.

4.1.4.5. Situación actual

Agricultores de las comunidades de San Gabriel identifican enfermedades durante el cultivo. La detección de prototipos de patologías en la arveja mediante visión por computadora puede ayudar a los agricultores y asesores técnicos a identificar de manera confiable enfermedades comunes en el sembrío. El aspecto técnico se apoya en expertos que ofrecen la orientación y los datos necesarios para asegurar que el prototipo cumpla su función.

4.1.5. Metodología XP

4.1.5.1. Fase de planificación

Durante esta fase, se establecen las funciones específicas de cada desarrollador y se determinan los cronogramas para la aplicación de cada herramienta metodológica.

a. Roles del proyecto

Tabla 13. Roles

| Nombre | Descripción | Rol XP |
|-----------------------|---|-------------|
| MSc. Carlitos Guano | Tutor | Consultor |
| Washington Castro | Investigador | Programador |
| Ing. Jonathan Amolotó | Representante técnico "Comercial de Agrisa" | Consultor |
| Ing. Galo Sánchez | Representante técnico "Comercial Casa de los Bonos" | Consultor |

b. Estimación del tiempo

La valoración del tiempo implica el valor de las horas y días de la semana laboral. Esta evaluación será fundamental para determinar los parámetros en las etapas siguientes de la metodología.

Tabla 14. Tiempo

| Semanas | Días | Horas |
|---------|------|-------|
| 0,2 | 1 | 4 |
| 0,4 | 2 | 8 |
| 1 | 5 | 20 |
| 1,6 | 8 | 32 |
| 2 | 10 | 40 |
| 2,4 | 12 | 48 |
| 2,6 | 13 | 52 |
| 3 | 15 | 60 |
| 4 | 20 | 80 |

c. Módulos de aplicación

- Modelo de visión artificial
- Entrenamiento del modelo
- Pruebas del modelo
- Implementación del modelo
- API REST
- Conexión con el modelo de visión artificial
- Envío y recepción de datos
- Aplicación móvil
- Conexión con la API REST
- Captura de datos
- Captura de imágenes
- Envío de imágenes a la API REST
- Consulta de datos
- Recepción de datos de la API REST
- Presentación de los resultados

d. Historia de usuario

Módulo 1: Inicio de sesión

Tabla 15. H_usuario 1

| Historia de usuario | |
|---|-----------------------------------|
| Nº1 | Usuario: Administrador |
| Nombre de historia: Entrenamiento del modelo de visión por computador. | |
| Prioridad: Alta | Riesgo de desarrollo: Alto |
| Estimación de tiempo: 3 semanas | Interacción: 1 |
| Responsable: Washington Castro | |
| Descripción: El administrador será quien actualice y entrene el modelo por visión por computador | |
| Detalle: Puede acceder directamente al script de entrenamiento del modelo y cargar los datos de la siguiente manera: | |
| <ul style="list-style-type: none">• Etiqueta• Imagen | |

Tabla 16. H_usuario 2

| Historia de usuario | |
|--|-----------------------------------|
| Nº 2 | Usuario: Administrador |
| Nombre de historia: Pruebas del modelo | |
| Prioridad: Alta | Riesgo de desarrollo: Alto |
| Estimación de tiempo: 3 semanas | Interacción: 2 |
| Responsable: Washington Castro | |
| Descripción: El administrador será quien podrá probar el modelo de visión por computador. | |
| Detalle: El acceso a modelos capacitados se puede probar antes de comenzar desde el programa. | |

Tabla 17. H_usuario 3

| Historia de usuario | |
|--|-----------------------------------|
| Nº 3 | Usuario: Administrador |
| Nombre de historia: Implementación del modelo | |
| Prioridad: Alta | Riesgo de desarrollo: Alto |
| Estimación de tiempo: 3 semanas | Iteración: 1 |
| Responsable: Washington Castro | |
| Descripción: El administrador será quien vincule el nuevo modelo de visión por computadora y ubicarlo en la aplicación. | |
| Detalle: Puede acceder a modelos entrenados y probarlos antes de implementarlos en su aplicación. | |

Módulo 2: API REST

Tabla 18. H_usuario 4

| Historia de usuario | |
|---|-----------------------------------|
| Nº 4 | Usuario: Administrador |
| Nombre de historia: Conexión con el modelo de visión por computador | |
| Prioridad: Alta | Riesgo de desarrollo: Alto |
| Estimación de tiempo: 5 semanas | Iteración: 1 |
| Responsable: Washington Castro | |
| Descripción: Conexión de datos API REST | |
| Detalle: API REST podrá conectarse con la información y con el modelo de visión por computador | |

Tabla 19. H_usuario 5

| Historia de usuario | |
|--|-----------------------------------|
| Nº 5 | Usuario: Administrador |
| Nombre de historia: Conexión con la aplicación móvil | |
| Prioridad: Alta | Riesgo de desarrollo: Alto |
| Estimación de tiempo: 3 semanas | Iteración: 2 |
| Responsable: Washington Castro | |
| Descripción: API REST de conexión con los datos de la aplicación móvil. | |
| Detalle: API REST será capaz de conectarse con la aplicación móvil. | |

Módulo 3: Captura de datos

Tabla 20. H_usuario 6

| Historia de usuario | |
|--|-----------------------------------|
| Nº 6 | Usuario: Administrador |
| Nombre de historia: Captura de las imágenes | |
| Prioridad: Alta | Riesgo de desarrollo: Bajo |
| Estimación de tiempo: 2 semanas | Iteración: 2 |
| Responsable: Washington Castro | |
| Descripción: La captura de fotografías o seleccionar desde la galería en la app móvil. | |
| Detalle: La aplicación móvil tendrá la función de capturar las fotografías o subir las imágenes desde la galería para su análisis con la visión por computador. | |

Tabla 21. H_usuario 7

| Historia de usuario | |
|--|------------------------------------|
| Nº 7 | Usuario: Administrador |
| Nombre de historia: Envío de las imágenes a la API REST | |
| Prioridad: Alta | Riesgo de desarrollo: Medio |

Estimación de tiempo: 1 semana

Iteración: 2

Responsable: Washington Castro

Descripción: La captura de fotografías o seleccionar desde la galería en la app móvil.

Detalle: La aplicación móvil tendrá la función de capturar las fotografías o subir las imágenes desde galería para su análisis con la visión por computador.

Módulo 4: Consulta de datos

Tabla 22. H_usuario 8

| Historia de usuario | |
|--|------------------------------------|
| Nº 8 | Usuario: Agricultor |
| Nombre de historia: Aceptación de los datos de API REST | |
| Prioridad: Alta | Riesgo de desarrollo: Medio |
| Estimación de tiempo: 1 semana | Iteración: 2 |
| Responsable: Washington Castro | |
| Descripción: El modelo recibe la imagen. | |
| Detalle: El modelo de visión por computador recibirá la imagen, la cual será procesada. | |

4.1.5.2. Fase de diseño

El prototipo emplea la arquitectura cliente-servidor con el objetivo de lograr resultados óptimos.

Interfaz 1: Login

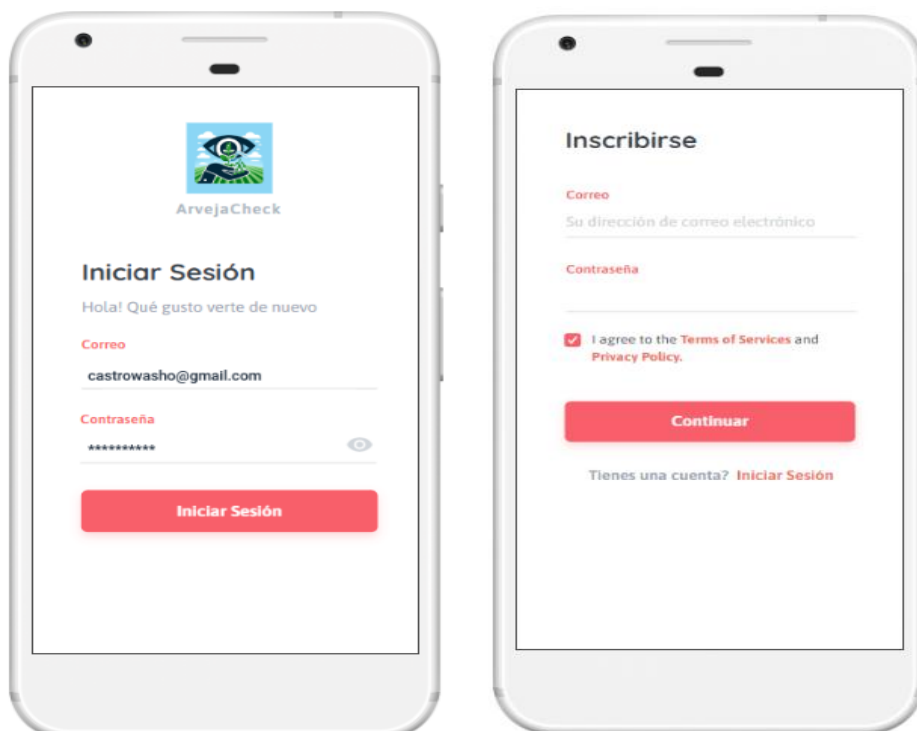


Figura 20. Prototipo login

Interfaz 2: Menú de inicio

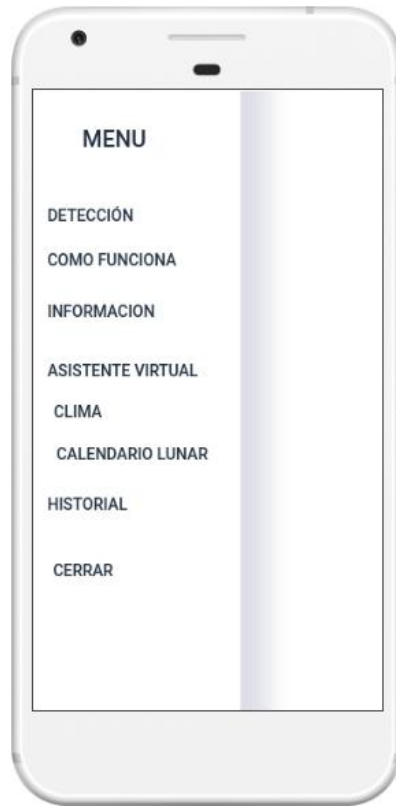
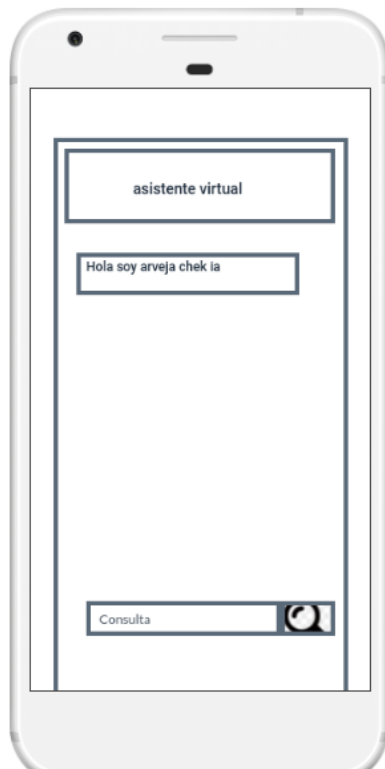
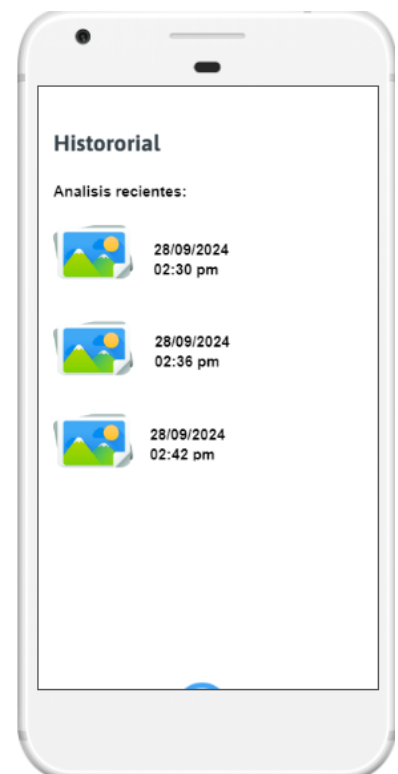


Figura 21. Prototipo menú de inicio

Interfaz 3: Manejo agrónomo



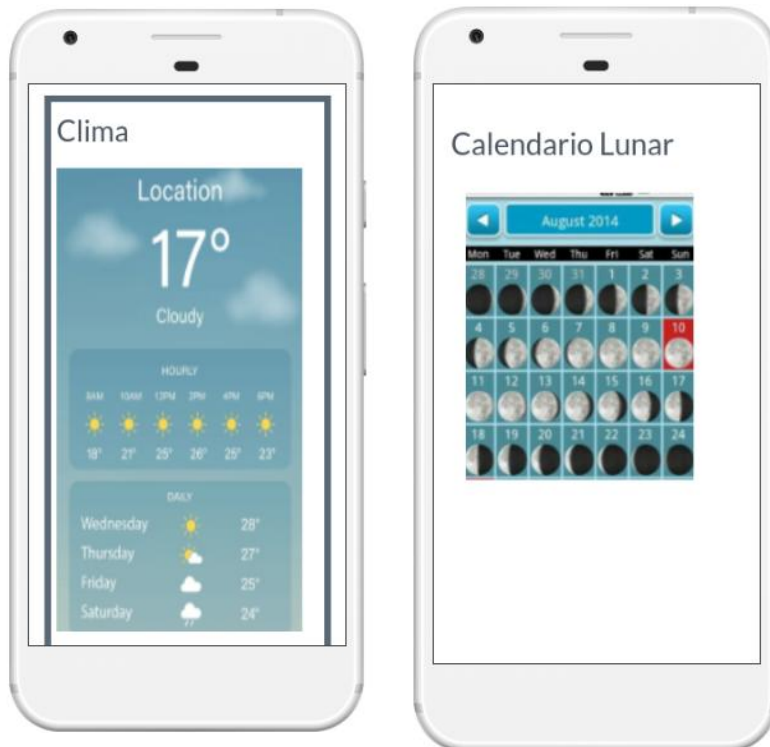


Figura 22. Prototipo de información

4.1.5.3. Fase de codificación

Entorno y herramienta de desarrollo

Para la implementación del sistema, se usó Dart como lenguaje de programación principal. El mismo que fue combinado con el framework Flutter, el cual ofrece un conjunto completo de herramientas para construir interfaces de usuario multiplataforma. Esta combinación permite construir aplicaciones nativas para iOS y Android utilizando un único código fuente.

Para el desarrollo, se utilizó Visual Studio Code como entorno integrado (IDE). Esta herramienta ofrece la funcionalidad necesaria para escribir y depurar el código fuente, proporcionando extensiones especializadas para el desarrollo con Dart y Flutter agilizando el flujo de trabajo del desarrollador.

Para la gestión de datos y autenticación, se implementó Firebase como plataforma de BaaS (Backend-as-a-Service), utilizando específicamente:

- **Firestore Authentication:** para la gestión y autenticación segura de usuarios.

- **Firestore Database:** para el almacenamiento de datos estructurados, incluyendo imágenes capturadas y en tiempo real.

Cabe mencionar que los servicios de Firebase requieren conectividad a internet para la sincronización y persistencia de datos en la nube.

Para la integración de inteligencia artificial, se utilizó la API de Google Gemini (versión Gemini-2.0-flash-lite) para el análisis avanzado de datos y el procesamiento de lenguaje natural. Es importante destacar que se requiere conexión de internet para enviar las consultas al modelo.

En cuanto al manejo de dependencias, se utilizaron las librerías oficiales de Flutter y Dart disponibles en el repositorio Pub.dev, las cuales añaden funcionalidades necesarias para el desarrollo de la aplicación. Finalmente, para la clasificación de imágenes mediante aprendizaje automático, se implementó un modelo de red neuronal convolucional empleando Teachable Machine de Google, exportando posteriormente como modelo TensorFlow Lite.

Este modelo facilita la clasificación de patologías a través del análisis de imágenes directamente en el dispositivo, una vez descargado, funciona sin conexión a internet, asegurando la funcionalidad de diagnóstico en todo momento.

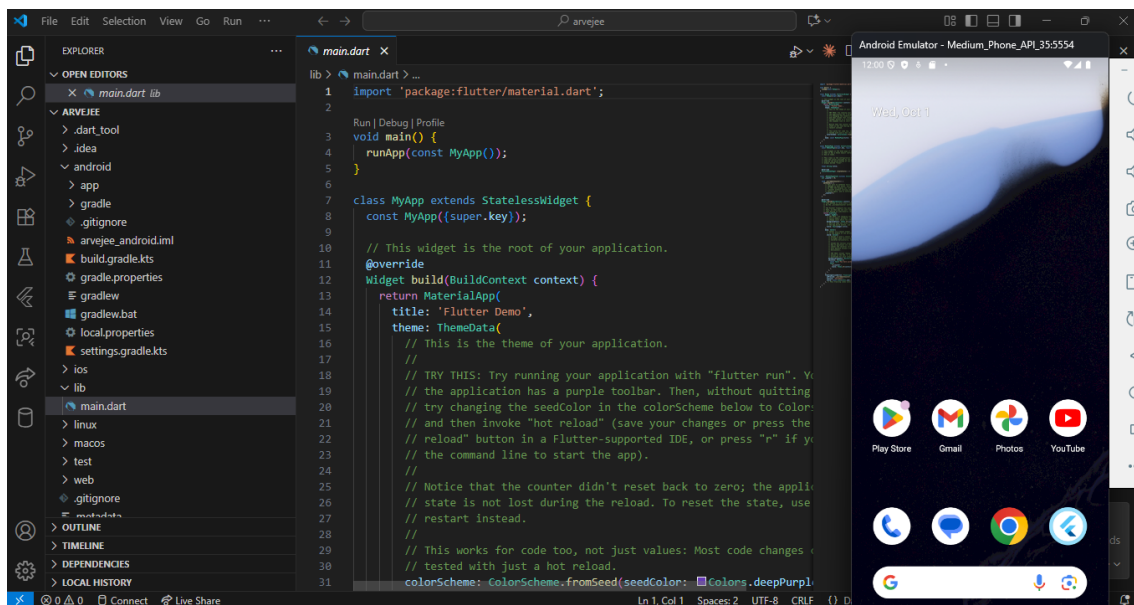


Figura 23. Visual Studio Code

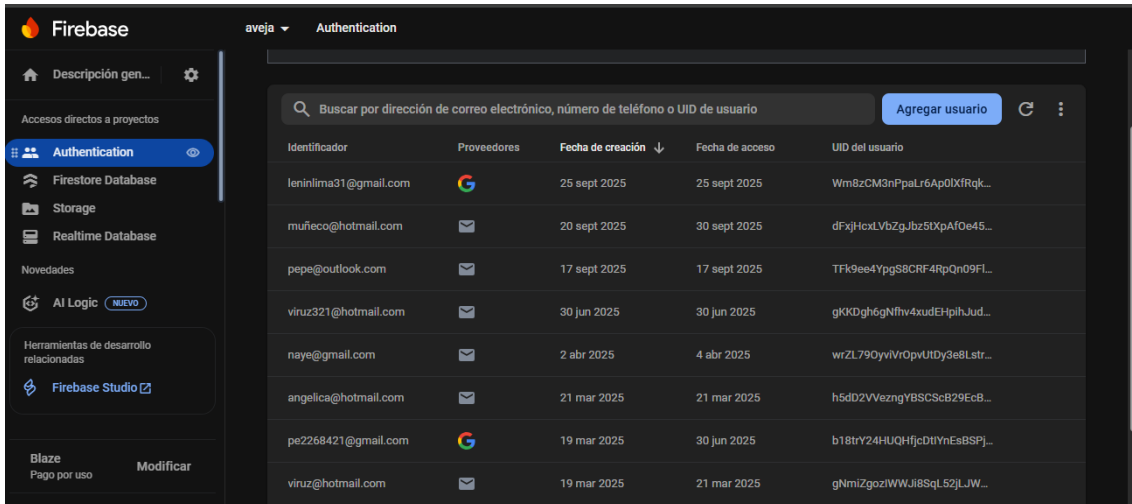


Figura 24. Autenticación del usuario a través de FireBase

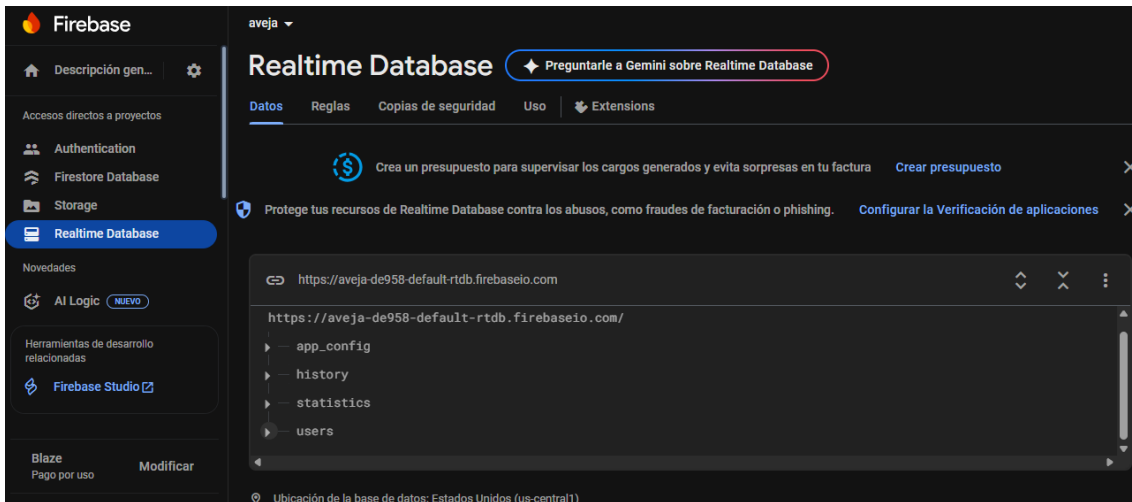


Figura 25. Realtime Database

Firestore Database utilizado para el almacenamiento de datos estructurados, incluyendo las imágenes capturadas por la aplicación y en tiempo real.

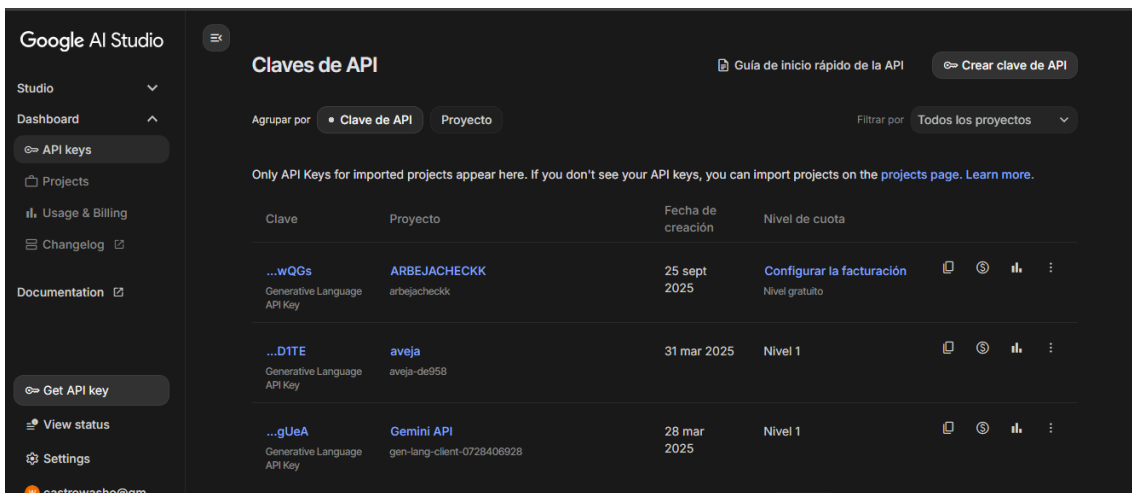


Figura 26. API de Google Gemini (versión gemini-2.0-flash-lite)

4.1.5.4. Creación modelo de detección de enfermedades

Agregar fotografías para empezar a entrenar el modelo de detección

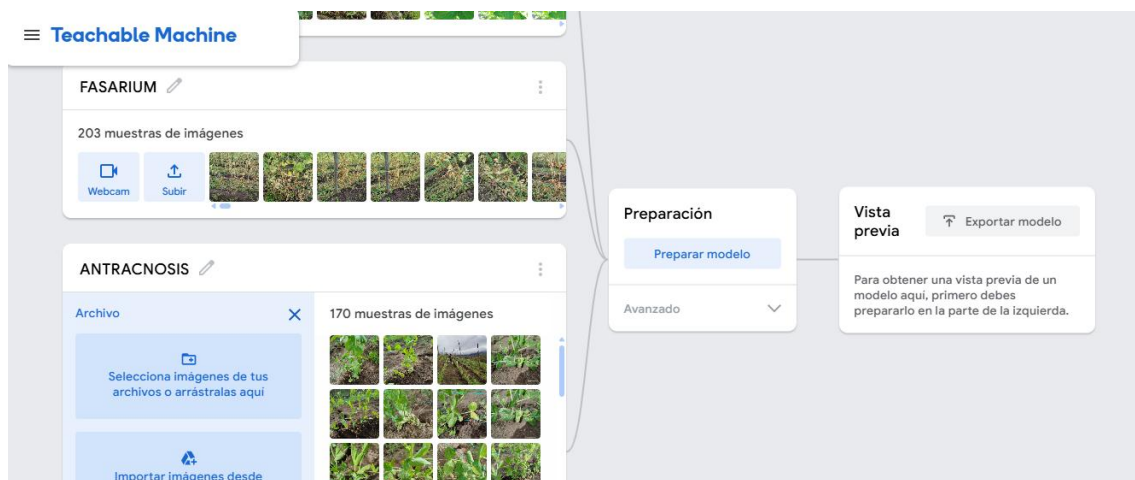


Figura 27. Agregar imágenes de hojas de arveja de amarre con enfermedades

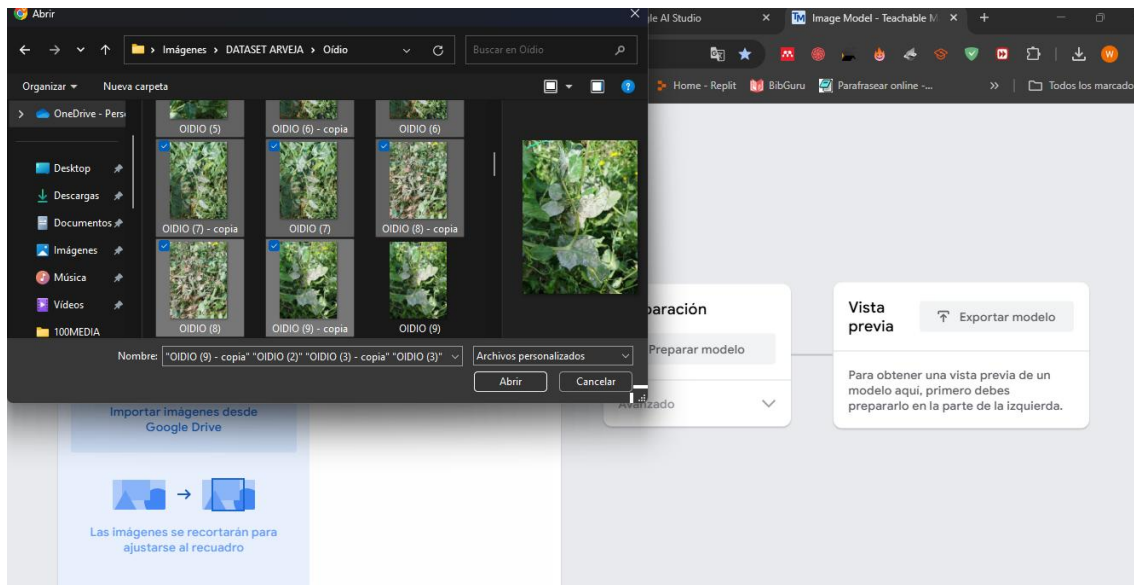


Figura 28. Marcación de imágenes para crear el modelo de detección

Una vez cargadas las imágenes en la aplicación, se hizo el etiquetado según la enfermedad representada. Posteriormente, se ejecutó el proceso de entrenamiento automático del modelo, siguiendo los parámetros establecidos por el sistema.

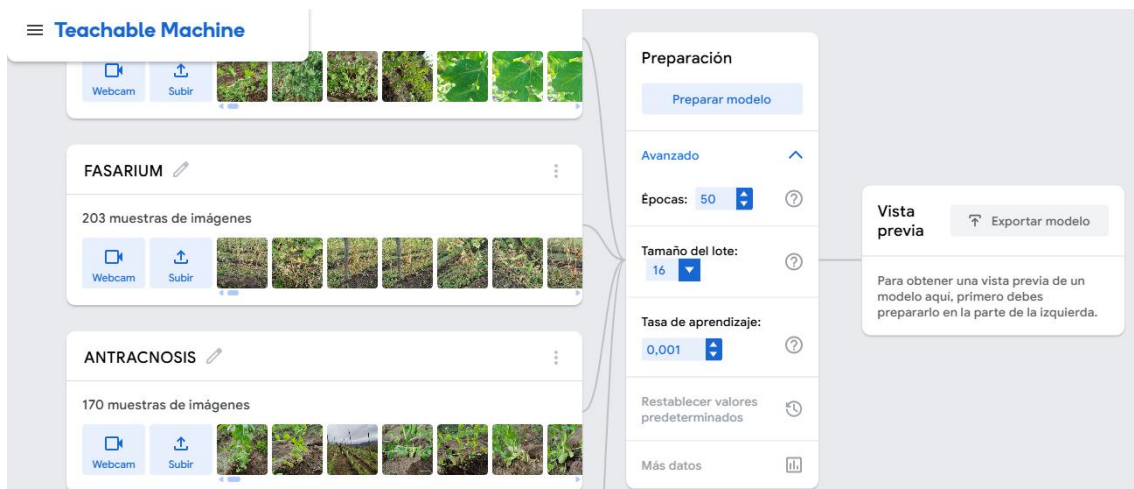


Figura 29. Etiquetado de imágenes con las enfermedades de la arveja

Tras el entrenamiento, el modelo se exportó en formato TensorFlow Lite (model.tflite). Este archivo, junto con su archivo de etiquetas asociado (labels.txt), permitió la integración directa en el módulo de detección de patologías de la aplicación.

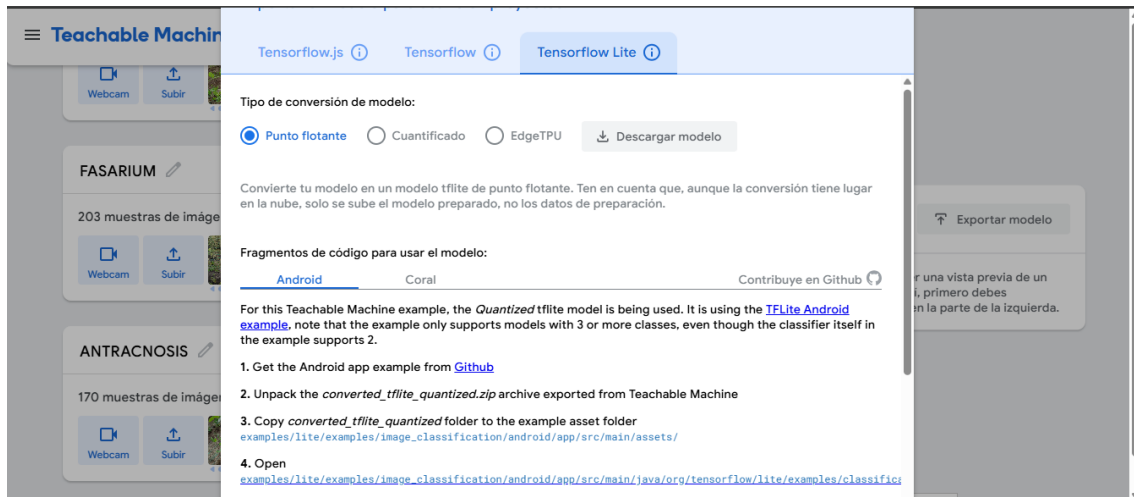


Figura 30. Conversión modelo TensorFlow lite

4.1.5.5. Codificación Visual Studio Code

El desarrollo de los módulos del sistema requirió integrar librerías externas obtenidas del repositorio oficial Pub.dev. Esta estrategia permitió añadir las funcionalidades para construir las interfaces y cumplir con los requerimientos funcionales establecidos.

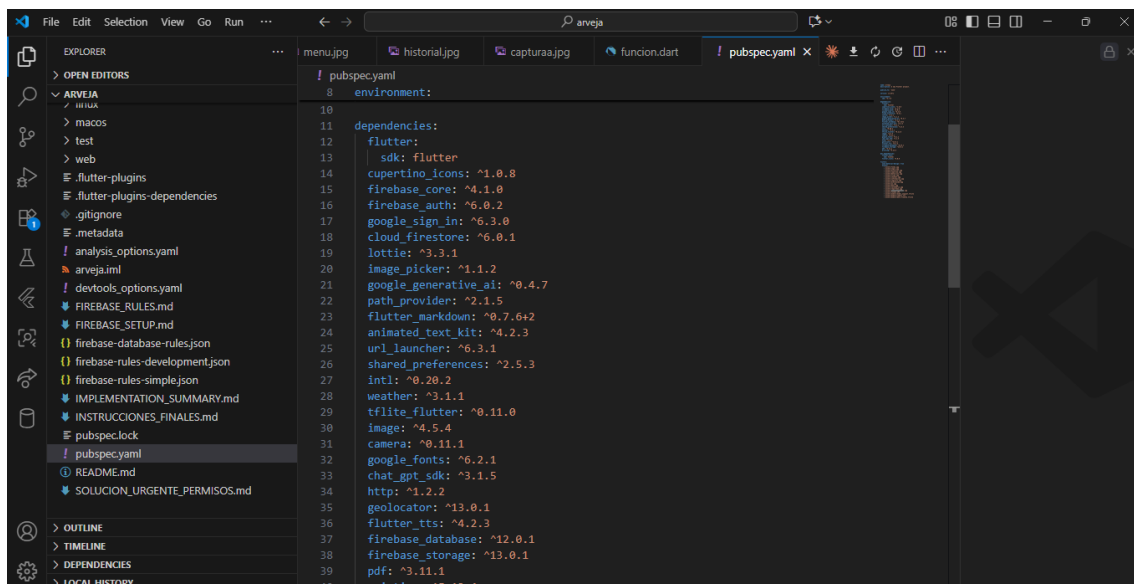


Figura 31. Dependencia en pubspec.yaml

En el desarrollo de las interfaces gráficas, se desarrollaron componentes visuales para capturar y gestionar datos. El módulo de autenticación incluye campos para credenciales (correo y contraseña del usuario) junto a opciones de acceso con Google (OAuth), recuperación de contraseña y un sistema de registro. Además, se

incorporaron validaciones en tiempo real que alertan sobre cualquier dato inconsistente.

Los módulos funcionales del sistema (incluyen captura de datos, generación de reportes, detección de patologías, etc.) se desarrollaron usando widgets personalizados. Estos no solo incluyen validaciones de datos y navegación interna, sino también rutas de transición entre pantallas. Lo que permite al usuario realizar operaciones CRUD (leer, crear, actualizar y eliminar) y acceder a las herramientas especializadas para el sector agrícola.

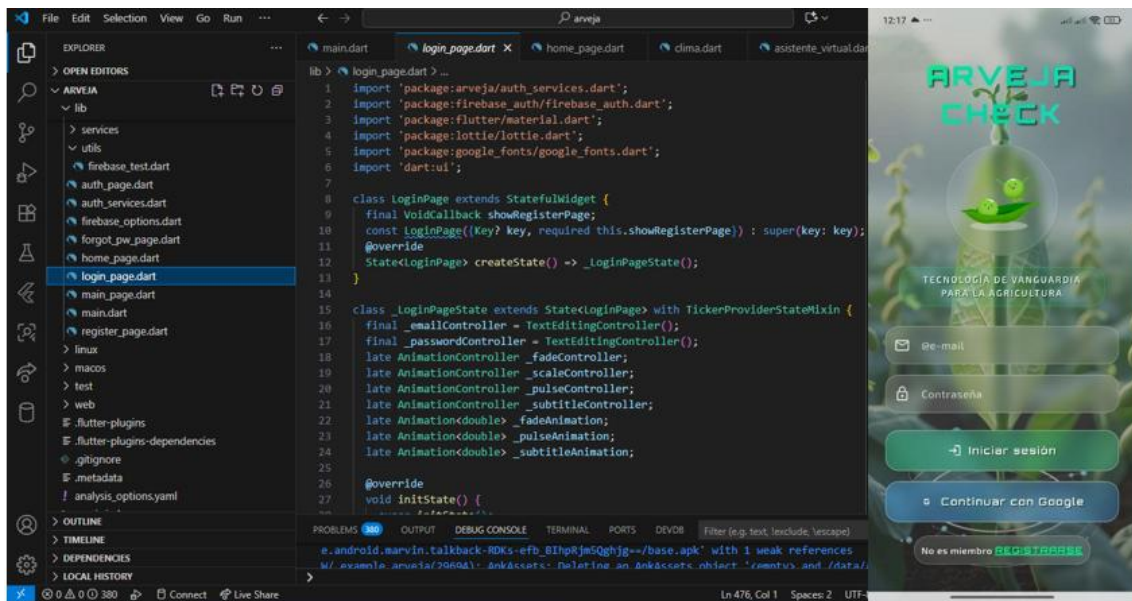


Figura 32. creación interfaz login o inicio de sesión

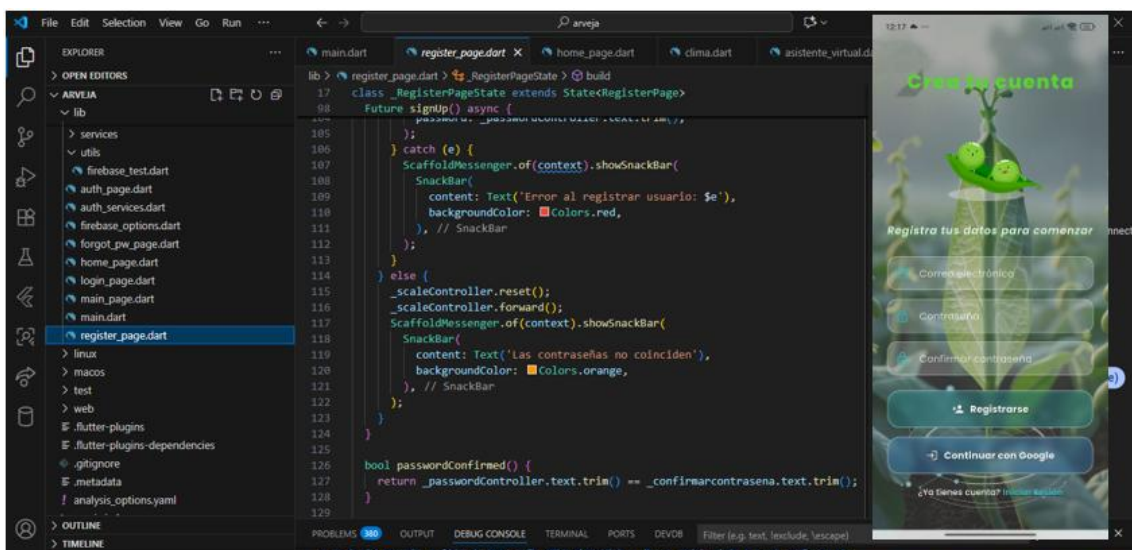


Figura 33. creación interfaz de registro

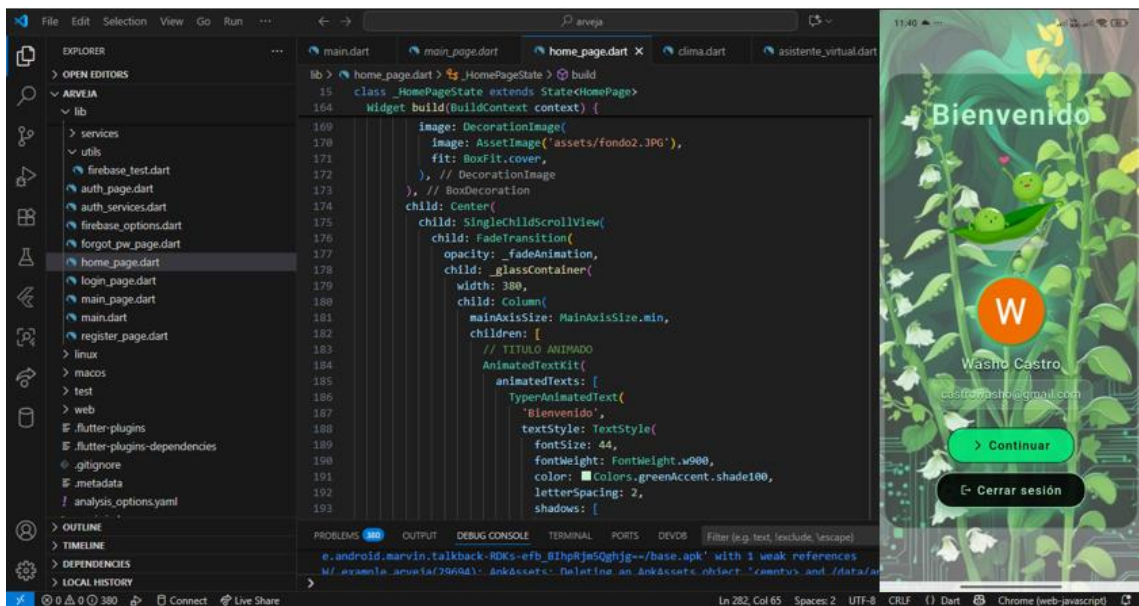


Figura 34. Bienvenida después de logearse

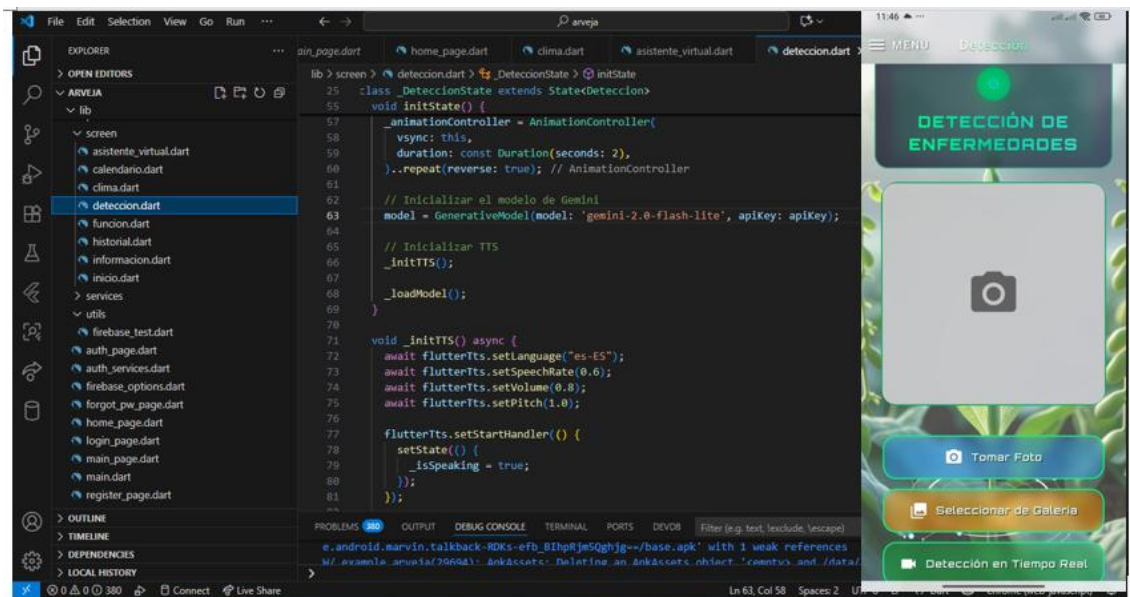


Figura 35. Detección de enfermedades

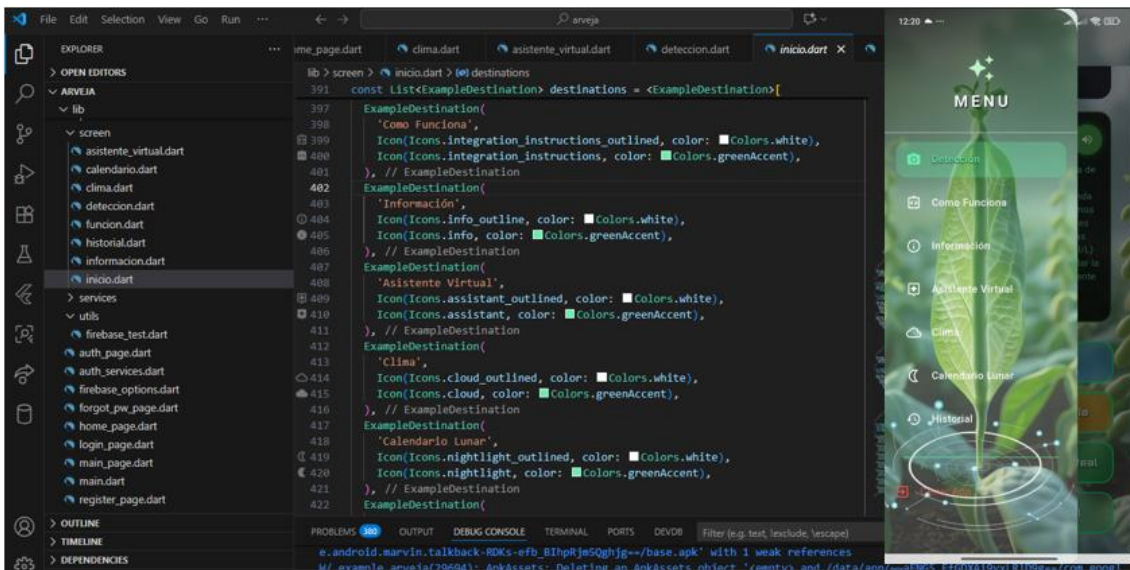


Figura 36. Menú del contenido de la aplicación

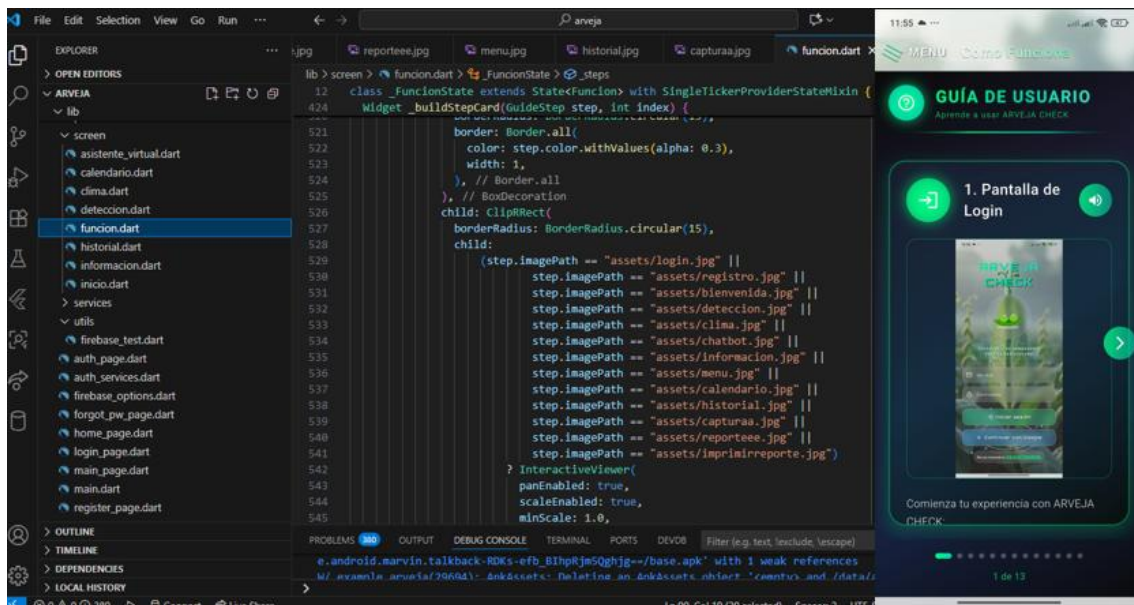


Figura 37. Guía de usuario

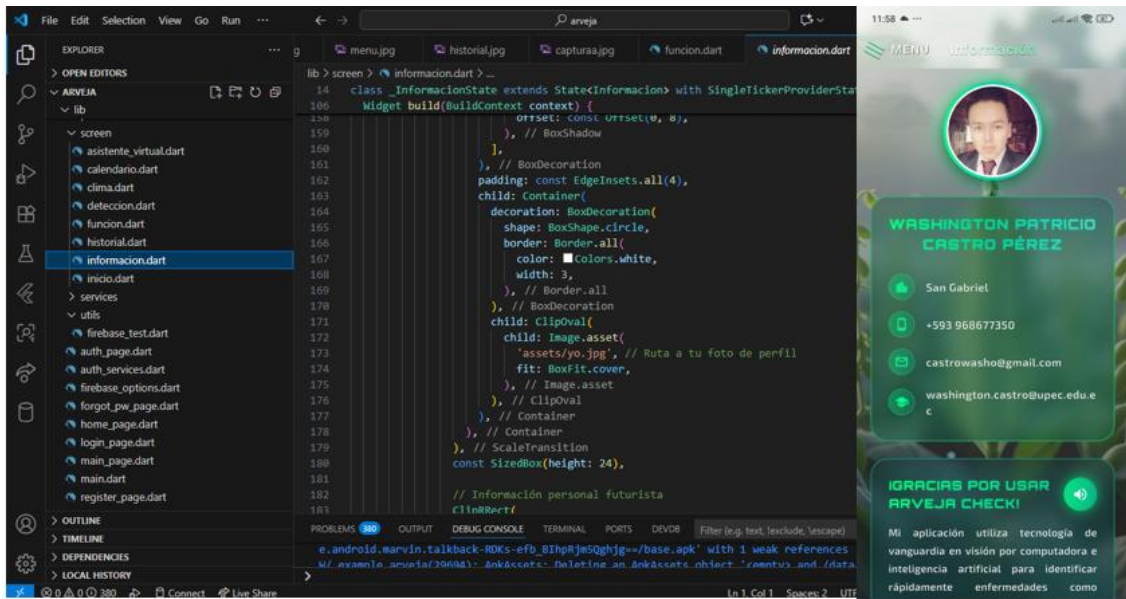


Figura 38. información del creador de la aplicación

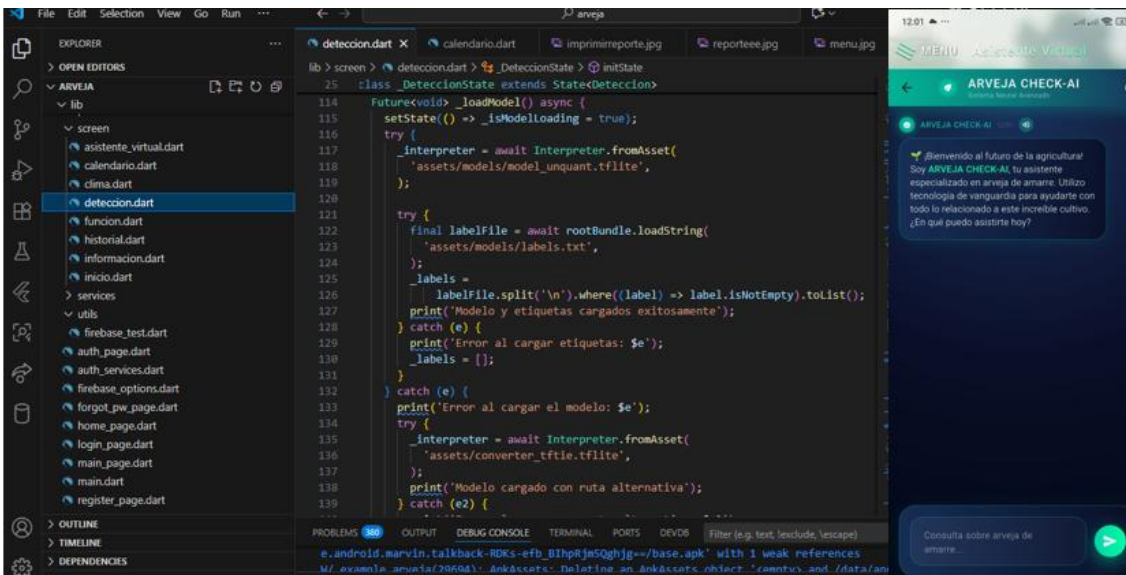


Figura 39. Asistente virtual especializado en arveja

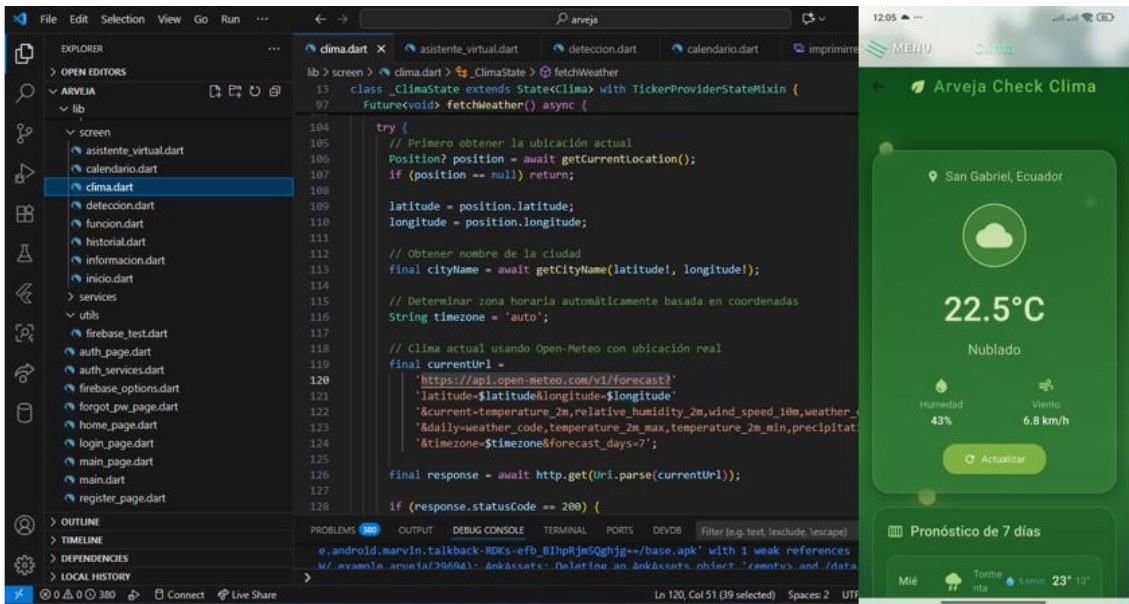


Figura 40. Interfaz gráfica del clima

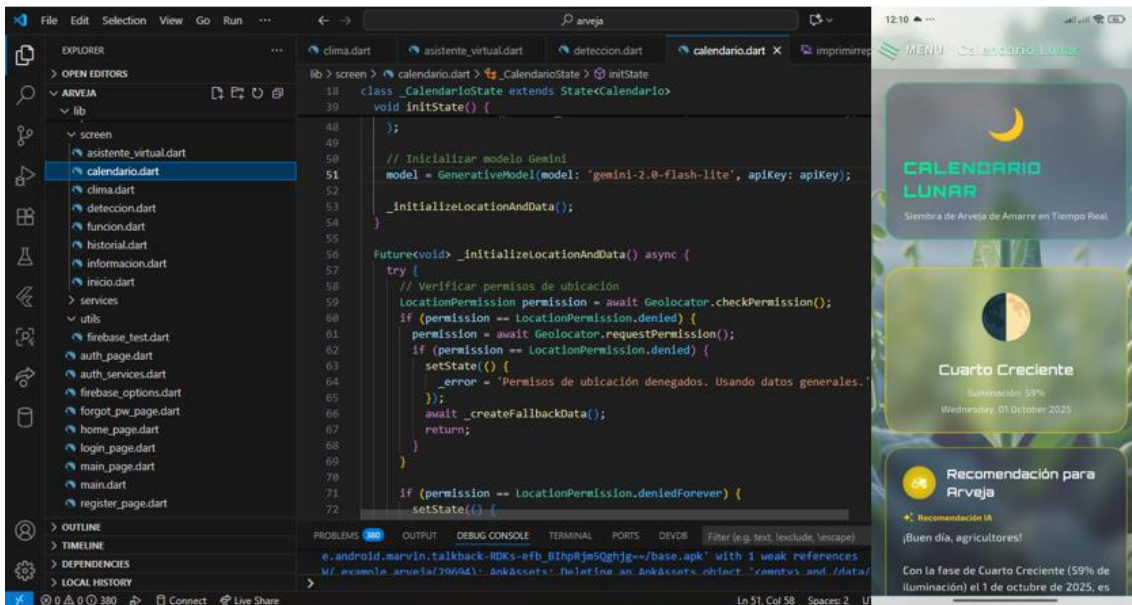


Figura 41. Calendario lunar

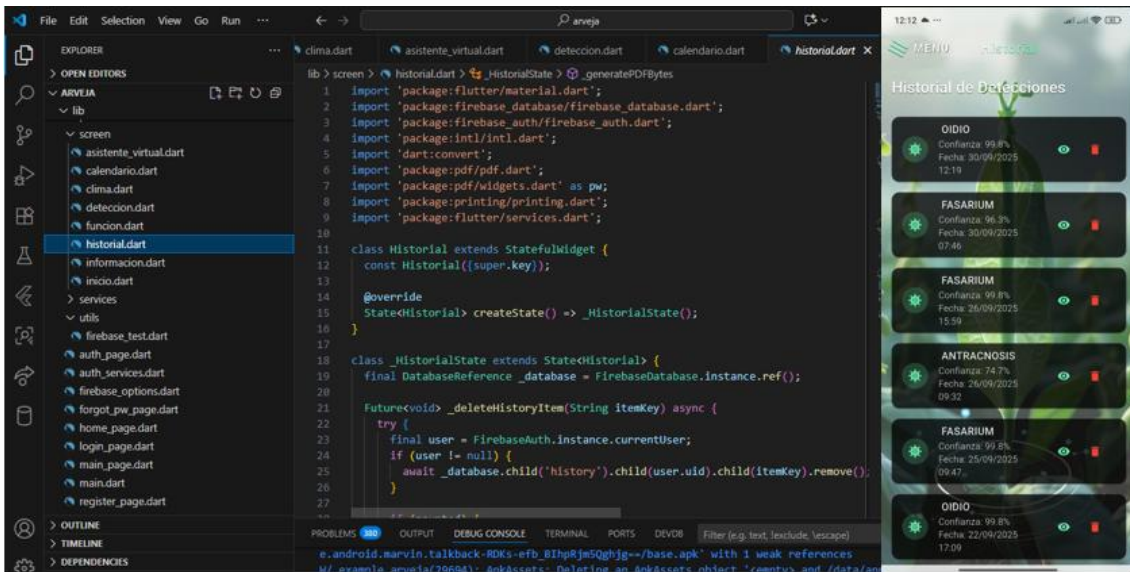


Figura 42. Historial de detecciones

4.1.5.6. Fase de prueba

Apptim es la plataforma utilizada para evaluar la usabilidad y analizar el rendimiento de la aplicación en dispositivos reales. Esto ayuda a detectar y solucionar problemas de performance durante la fase de desarrollo, antes de que el software se lance a producción.

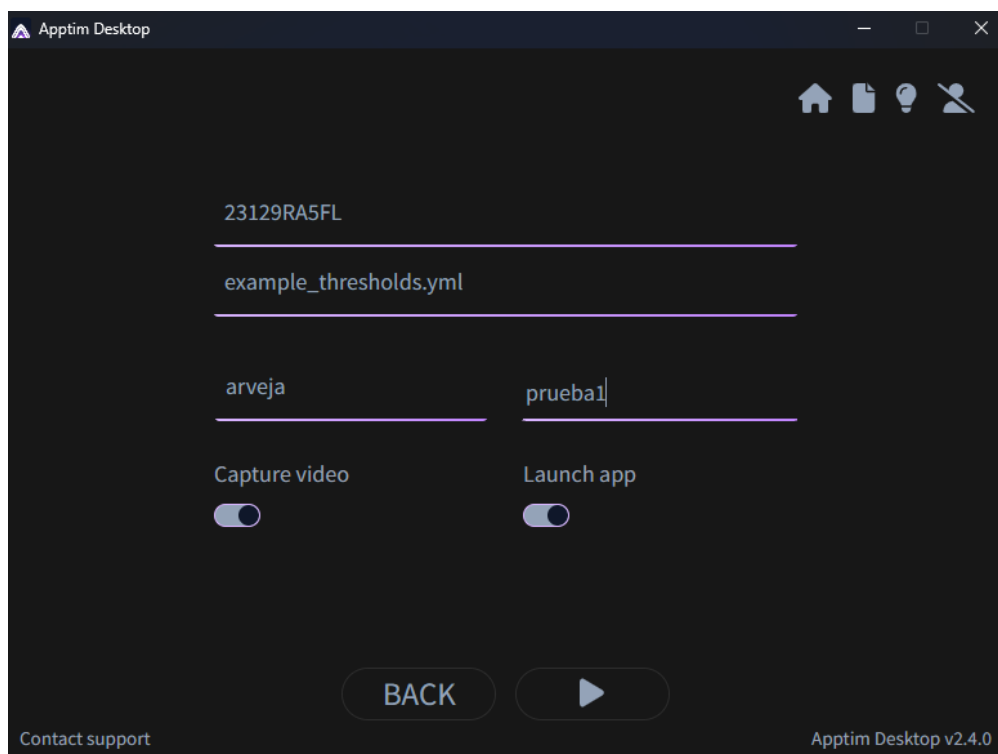


Figura 43. Selección del dispositivo y aplicación a analizar

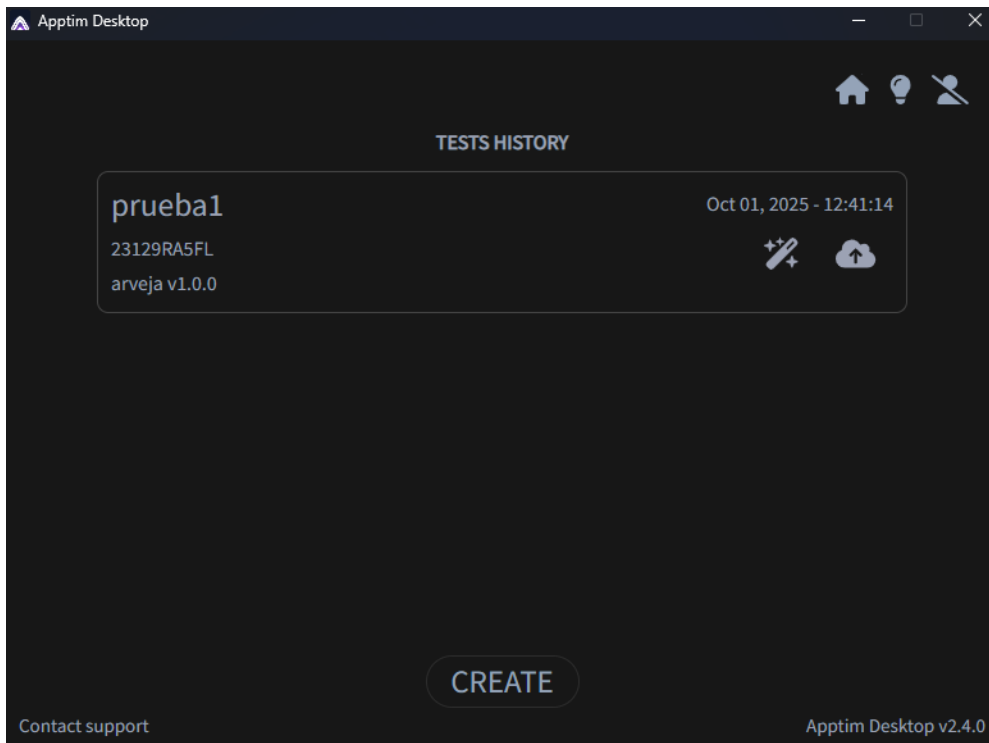


Figura 44. Test de pruebas

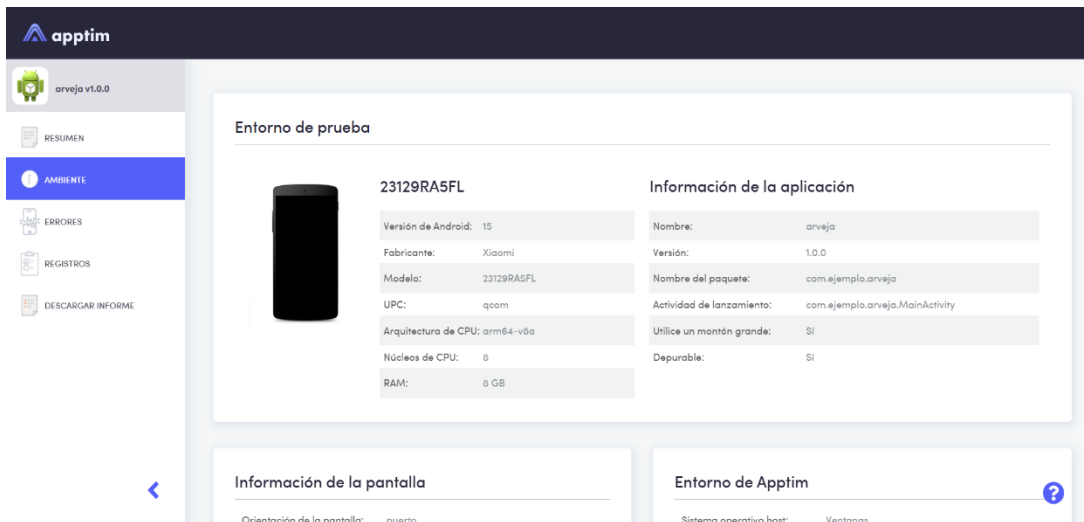


Figura 45. Entorno de pruebas

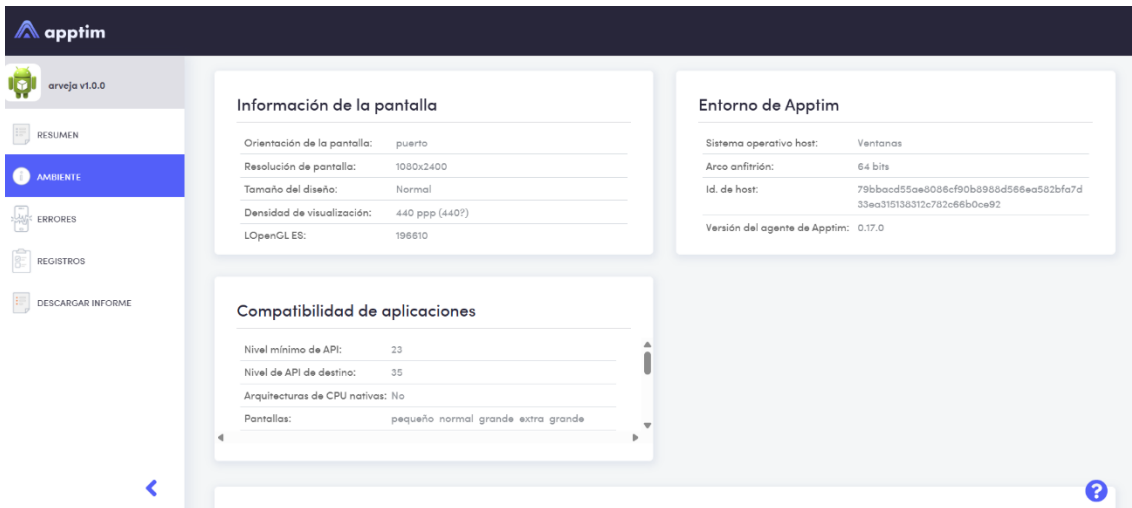


Figura 46. Información de compatibilidad de la app con versiones Android

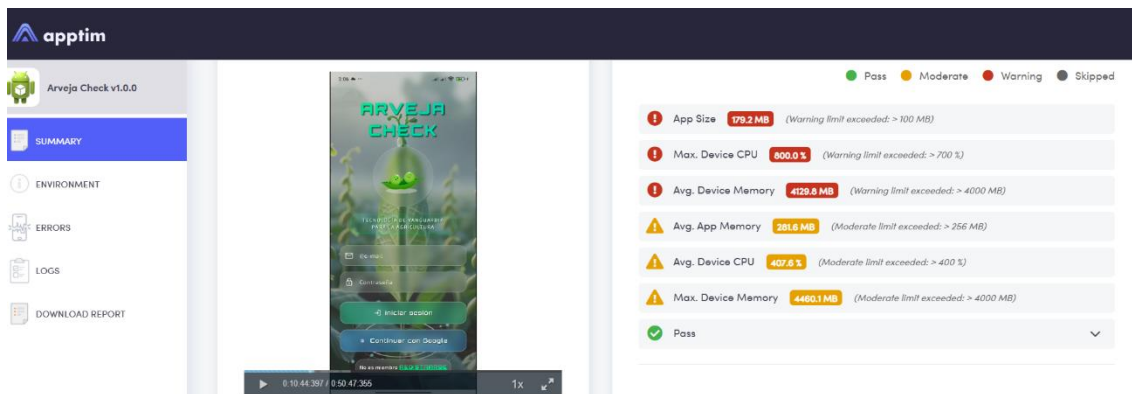


Figura 47. Resumen de prueba con resultado PASS (satisfactorio)



Figura 48. Resultado de los niveles de uso de RAM, CPU y red en el dispositivo

Firebase Test Lab es una plataforma que ofrece un entorno de pruebas totalmente autorizado y basado en la nube. Su función principal es verificar la integridad del sistema analizando la estructura de la interfaz y ejecutando pruebas simuladas en una variedad de dispositivos virtuales que pueden usar sin cargos extras. La herramienta documenta cada prueba entregando capturas de pantalla, informes detallados y grabaciones de video.

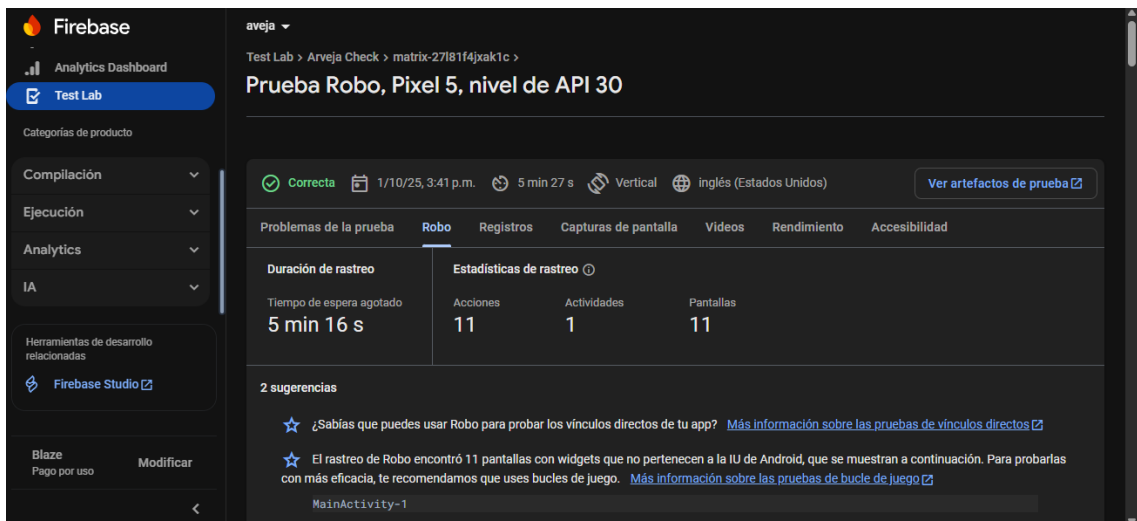


Figura 49. Prueba robo o stress realizado a la aplicación Arveja Cheek

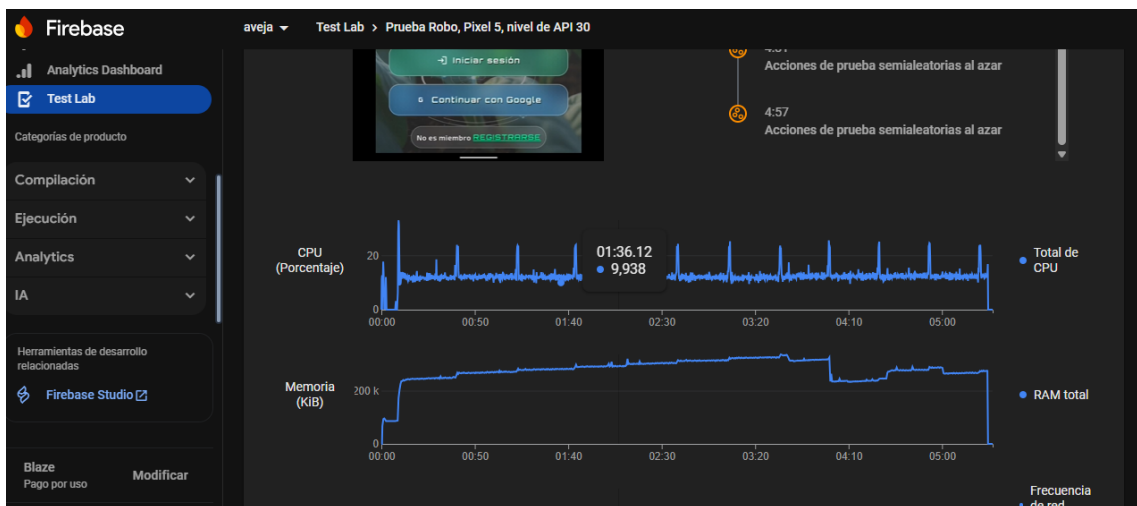


Figura 50. Rendimiento CPU, gráficos y RAM

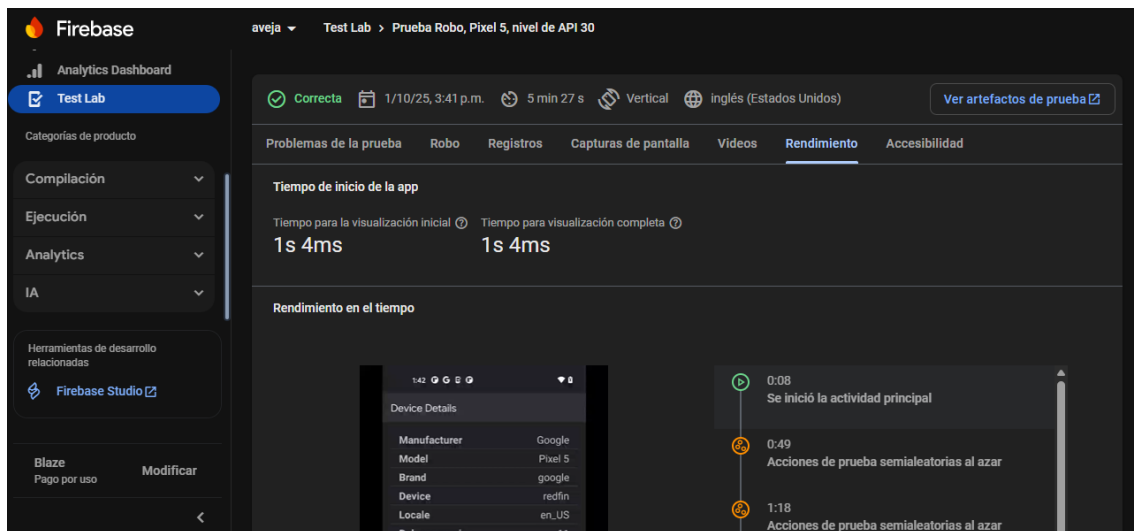


Figura 51. Resultados de rendimiento de la prueba en Test Lab

4.1.6. Análisis de la evaluación- Aplicación ARVEJA CHECK

La aplicación ARVEJA CHECK fue evaluada por el Msc David Herrera docente de la carrera de Agropecuaria quien aprobó las 10 preguntas referentes a la app, tal y como se muestra en la figura 46. La evaluación cubrió desde la seguridad y acceso (pregunta 1), las funcionalidades centrales basadas en IA (preguntas 2, 3, 5), utilidad práctica como el clima, calendario lunar, historial (preguntas 6, 7, 8) y la usabilidad como menú, reportes, guía de usuario (preguntas 4, 9 y 10). La aplicación cumplió con todas las expectativas del docente y las especificaciones funcionales, ya que detecta perfectamente tres enfermedades en el cultivo de arveja.



Figura 52. Aprobación por parte del docente

4.2. DISCUSIÓN

Los objetivos generales y específicos desempeñaron un papel importante en la orientación para la realización de este trabajo de investigación. Esto hizo posible la descripción y el análisis exhaustivo de todos los hallazgos, dado que cada resultado se fundamentó en las bases teóricas del Capítulo II y en la metodología XP, adaptada específicamente para el proyecto en el Capítulo III.

Los cambios climáticos representan un desafío para los agricultores al momento de detectar plagas y patologías en el cultivo de arveja. Se han presentado sistemas de identificación de lesiones necróticas basados en métodos de aprendizaje profundo como solución a este problema (indepth education). Estos métodos identifican rápido y eficazmente la presencia de ciertas plagas en cultivos de soja y arroz usando fotos tomadas con drones y teléfonos inteligente.

Con una alta intersección de valores de obligación (IoU), precisión media (mAP) y eficiencia de detección, los hallazgos de la investigación son alentadores en términos de precisión y eficacia de estos sistemas de detección. La agricultura 4.0 depende de los avances en el procesamiento de imágenes digitales porque permiten a los agricultores aumentar la productividad de manera crucial con el medio ambiente. Así mismo, existe la manera de mejorar la fiabilidad y precisión de estos sistemas, lo que estimulará más investigaciones en este ámbito.

V. CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

- La computadora por visión permite identificar con alta precisión las enfermedades que afectan el cultivo de arveja, contribuyendo a la detección temprana y prevención de daños significativos. Las técnicas de aprendizaje profundo, específicamente redes neuronales convolucionales, son efectivas para el procesamiento y clasificación automatizada de imágenes fitosanitarias en el cultivo.
- La metodología ágil XP facilitó el desarrollo iterativo y colaborativo del prototipo, asegurando que cumpliera con las necesidades de los usuarios y la adaptación constante según la retroalimentación recibida. La integración de tecnologías accesibles y plataformas de desarrollo modernas posibilita el uso de estos sistemas en contextos rurales, donde los recursos tecnológicos y económicos son limitados.
- La implementación del sistema contribuye a mejorar la productividad agrícola y la seguridad alimentaria, al permitir intervenciones oportunas y reducir las pérdidas económicas derivadas de enfermedades.

5.2. RECOMENDACIONES

- Ampliar el entrenamiento del modelo para incluir un mayor número de enfermedades y mejorar la exactitud del sistema en diferentes condiciones ambientales.
- Para futuras investigaciones se recomienda emplear el algoritmo YOLO para la creación de este sistema.
- Desarrollar un sistema de alertas en tiempo real que notifique a los agricultores sobre la detección de enfermedades para una rápida toma de decisiones.
- Extender la aplicación para la detección en otros cultivos relevantes, fomentando su uso en una mayor diversidad agrícola.
- Incorporar módulos complementarios que analizan variables ambientales y climáticas para anticipar el desarrollo de enfermedades.

- Capacitar a los agricultores en el uso y manejo adecuado de estas tecnologías para maximizar el impacto positivo en la producción y manejo fitosanitario.

VI. REFERENCIAS BIBLIOGRÁFICAS

- Alvarado, L. (31 de julio de 2022). ¿Qué son las TIC y cuál es su importancia? Politécnico GranColombiano: <https://www.poli.edu.co/blog/poliverso/que-son-las-tic>
- Amaguaña, J., Chuqui, F., Guamán, D., & Quishpe, C. (2023). El auge exponencial del lenguaje Python en el desarrollo tecnológico. *Revista Científica INGENIAR: Ingeniería, Tecnología e Investigación*. ISSN: 2737-6249, 6(12), 240-256. <https://journalingeniar.org/index.php/ingeniar/article/view/152>
- Angulo, A. (2019). *Identificación de las principales plagas y enfermedades en el cultivo de arveja (Pisum sativum), parroquia Bolívar, cantón Bolívar, Provincia del Carchi. [Tesis de grado]*. El Ángel, Carchi, Ecuador: Universidad Técnica de Babahoyo. Universidad Técnica de Babahoyo: <https://dspace.utb.edu.ec/bitstream/handle/49000/6395/E-UTB-FACIAG-ING%20AGRON-000152.pdf?sequence=1&isAllowed=y>
- Armendariz, V., Milla, A., Chandi, A., & Gallegos, L. (2023). Desarrollo de una herramienta para el aprendizaje del abecedario Ecuatoriano, basado en redes neuronales artificiales para la enseñanza inclusiva de lenguaje de señas. *Polo del conocimiento*, 8(7), 84-96. <https://doi.org/10.23857/pc.v8i7>
- Bergmann, D. (2025). *What are machine learning algorithms?* IBM: <https://www.ibm.com/think/topics/machine-learning-algorithms>
- Bodero, E., Lopez, M., Congacha, A., Cajamarca, E., & Morales, C. (2020). Google Colaboratory como alternativa para el procesamiento de una red neuronal convolucional. *Revista Espacios*, 41(07), 22. <https://www.revistaespacios.com/a20v41n07/a20v41n07p22.pdf>
- Campoverde, J. (2024). *DISEÑO DE APLICACIÓN PARA LA DETECCIÓN DE PLANTAS CON ANOMALÍAS EN CULTIVOS.* [Tesis de grado]. Guayaquil, Ecuador: UNIVERSIDAD POLITECNICA SALESIANA. <https://dspace.ups.edu.ec/bitstream/123456789/29066/1/UPS-GT005752.pdf>
- Chingo, W., & López, G. (2021). Paralelismos entre bases de datos relacionales y no relacionales (un enfoque en seguridad). *Revista electrónica de Computación, Informática, Biomédica y Electrónica*, 10(2), 1-16. <https://www.redalyc.org/journal/5122/512269058002/html/>

- Cimas, G. (22 de julio de 2022). *Visual Studio Code: Editor de código para desarrolladores*. OpenWebinars: <https://openwebinars.net/blog/que-es-visual-studio-code-y-que-ventajas-ofrece/>
- Darias, S. (09 de 09 de 2021). *Gestor de Base de datos: Qué es, Funcionalidades y Ejemplos*. Intelequia: <https://intelequia.com/es/blog/post/gestor-de-base-de-datos-qu%C3%A9-es-funcionalidades-y-ejemplos>
- Data Science Team. (03 de mayo de 2020). *R-CNN, R-CNN rápido, R-CNN más rápido, YOLO – Algoritmos de detección de objetos*. <https://datascience.eu/es/vision-artificial/r-cnn-r-cnn-rapido-r-cnn-mas-rapido-yolo-algoritmos-de-deteccion-de-objetos/>
- Diane, R., & Stryker, C. (27 de septiembre de 2024). *¿Qué es Google Gemini?* IBM: <https://www.ibm.com/think/topics/google-gemini>
- Díaz, J. (junio de 2021). *Aprendizaje Automático y Aprendizaje Profundo*. *Ingeniare. Revista chilena de ingeniería*, 29(2). https://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0718-33052021000200180
- Fernández, E. (26 de 08 de 2022). *Google Colab, una nueva herramienta de Google pensaba para especialistas de IA y Data Analysis*. Tokio School: <https://www.tokioschool.com/noticias/google-colab/>
- Franco, J., & Tokin, M. (2020). *Manejo integrado de la chupadera fungosa en el cultivo de arveja*. (I. N. Agraria, Ed.) <https://repositorio.inia.gob.pe/server/api/core/bitstreams/09c07428-aeca-40c4-9b37-3df5f9477e5f/content>
- Fuentes, J., Pierra, A., Fírvida, A., Pérez, H., López, A., & Sierra, L. (mayo de 2016). *Android para escritorio*. *Revista Cubana de Ciencias Informáticas*, 10, 82-93. <https://www.redalyc.org/pdf/3783/378346333007.pdf>
- Fuglie, K., Gautam, M., Goyal, A., & Maloney, W. (2020). *Harvesting Prosperity. Technology and Productivity Growth in Agriculture*. <https://openknowledge.worldbank.org/server/api/core/bitstreams/3621191c-15f3-5ede-a89c-f7190d7e1dba/content>
- Gómez, J., Jaramillo, H., & Guerrero, G. (2022). *Técnicas de procesamiento digital de imágenes para detección de plagas y enfermedades en cultivos: una revisión*.

González, L. (06 de abril de 2021). *Visión Computacional*. aprendeIA: <https://aprendeia.com/vision-computacional/>

Goodwin, M. (9 de abril de 2024). ¿Qué es una API (interfaz de programación de aplicaciones)? IBM: <https://www.ibm.com/think/topics/api>

Guan, Z. (2023). Real time object recognition based on YOLO model. *Theoretical and Natural Science*, 28(1), 137-143. <https://www.ewadirect.com/proceedings/tns/article/view/9302>

Guerrero, C. (2021). RECONOCIMIENTO DE LESIONES NECRÓTICAS PARA LA DETECCIÓN TEMPRANA DE LA PLAGA THRIPS (KAKOTHrips ROBUSTUS UZEL) EN LOS CULTIVOS DEL GUISANTE O ARVEJA MEDIANTE TÉCNICAS DEEP LEARNING [Tesis de posgrado]. Universidad Técnica del Norte. <https://repositorio.utn.edu.ec/bitstream/123456789/11654/2/PG%20909%20TRA%20BAJO%20GRADO.pdf>

Guerrero, C. (2021). Reconocimiento de lesiones necróticas para la detección temprana de la plaga thrips (kakothrips robustus uzal) en los cultivos del guisante o arveja mediante técnicas deep learning. [Tesis de grado]. Ibarra, Imbabura, Ecuador: Universidad Técnica del Norte. <https://doi.org/http://repositorio.utn.edu.ec/handle/123456789/11654>

Guevara, G., Verdesoto, A., & Castro, N. (2020). Metodologías de investigación educativa (descriptivas, experimentales, participativas, y de investigación-acción). *Revista Científica Mundo de la Investigación y el Conocimiento*, 4(3), 163-173. <https://recimundo.com/index.php/es/article/view/860>

Huaire, E. (2019). *Método de investigación*. Material de clase : <https://www.academica.org/edson.jorge.huaire.inacio/35.pdf>

Innovación. (11 de septiembre de 2024). *TensorFlow: la biblioteca de código abierto de Google para acelerar la adopción de la IA*. BBVA: <https://www.bbva.com/es/innovacion/tensorflow-la-biblioteca-de-codigo-abierto-de-google-para-acelerar-la-adopcion-de-la-ia/>

INTEF. (2019). *Programación y Robótica*. Instituto Nacional de Tecnologías Educativas y de Formación del Profesorado. <https://doi.org/https://formacion.intef.es/aulaenabierto/mod/book/view.php?id=2747>

- Integra. (18 de febrero de 2020). *Visión artificial en un ecosistema IoT aplicado a la agricultura de precisión. La innovación necesaria*: <https://www.integratecnologia.es/la-innovacion-necesaria/vision-artificial-en-un-ecosistema-iot-aplicado-a-la-agricultura-de-precision/#:~:text=En%20conclusi%C3%B3n%2C%20la%20visi%C3%B3n%20artificial,frente%20a%20situaciones%20inusuales%20>
- Keita, Z. (29 de enero de 2024). *Explicación de la detección de objetos YOLO*. datacamp: <https://www.datacamp.com/es/blog/yolo-object-detection-explained>
- Kili. (28 de diciembre de 2023). *Algoritmo YOLO: Detección de objetos en tiempo real de la A a la Z*. Tecnología Kili: <https://kili-technology.com/blog/yolo-algorithm-real-time-object-detection-from-a-to-z>
- Kurz, T., & Jayasuriya, S. (2024). The Impact of Teachable Machine on Middle School Teachers' Perceptions of Science Lessons after Professional Development. *Educ. Sci.*, 14(4), 417. <https://www.mdpi.com/2227-7102/14/4/417>
- Lago, M., Prieto, G., Gatti, I., Cointry, E., & Espósito, M. (diciembre de 2021). El oídio de la arveja: Comportamiento de variedades en el sur de Santa Fe. *Para Mejorar la Producción*, 137-142.
- Llenera, L., & Lombardo, L. (2021). APLICACIÓN WEB PARA LA GESTIÓN ADMINISTRATIVA EN LA ESTRATEGIA PLAN VACUNARSE. *Revista Universidad y Sociedad*, 12(2), 40-45. <https://doi.org/http://scielo.sld.cu/pdf/rus/v14n2/2218-3620-rus-14-02-40.pdf>
- Lui, N., Xu, S., Yao, X., Zhang, G., Mao, W., Hu, Q., Feng, Z., & Gong, Y. (11 de abril de 2016). Estudios sobre el control de la plaga de *Ascochyta* en guisantes de campo (*Pisum sativum* L.) causada por *Ascochyta pinodes* en la provincia de Zhejiang, China. *Portada. Microbiol.*, 7. <https://doi.org/10.3389/fmicb.2016.00481>
- Martínez, S., & Guerrero, C. (2022). Reconocimiento de lesiones necróticas para la detección de la plaga de trips en guisantes utilizando el modelo de aprendizaje profundo yolov4-tiny. *Revista Latinoamericana de Computación*, 9(1), 46-59. <https://doi.org/https://doi.org/10.5281/zenodo.5816314>
- Menegaz, M. (16 de marzo de 2018). *Entendiendo YOLO*. Hackernoon: <https://hackernoon.com/lang/es/compreesion-yolo-f5a74bbc7967>


- Mina, T. (2021). Desarrollo de aplicaciones web y móvil para la gestión de publicaciones científicas. *Polo del Conocimiento*, 6(6), 973-982. <https://doi.org/10.23857/pc.v6i6.2798>
- Mirmehdi, M. (2012). Visión artificial y por computadora. (E. Davies, Computer and Machine Visión, & Academic Press, Edits.) *ELSEVIER*, XXI-XXII. <https://doi.org/https://doi.org/10.1016/B978-0-12-386908-1.00032-X>
- Morán, D., & Quiranza, A. (2022). *Visión artificial para la detección de enfermedades en el cultivo de papa. [Tesis de grado]*. Tulcán, Carchi, Ecuador: Universidad Politécnica Estatal del Carchi. <http://repositorio.upec.edu.ec/handle/123456789/1705>
- Moreno, G. (2016). *JAVA como lenguaje universal de programación*. Universidad Autónoma del Estado de Hidalgo: <https://repository.uaeh.edu.mx/revistas/index.php/xikua/article/view/332/4434>
- Pabón, J., & Castaño, J. (8 de julio de 2012). MANEJO DE LA PUDRICIÓN RADICAL DE LA ARVEJA (*Pisum sativum* LINNEO) CAUSADA POR *Fusarium oxysporum* SCHLECHTEND.:FR. 20(2), 37-44. <https://www.cabidigitallibrary.org/doi/pdf/10.5555/20143270375>
- Palma, K., Garzón, J., Delgado, J., Zambrano, K., & Párraga, L. (junio de 2020). EL IMPACTO DE LAS APLICACIONES MÓVILES, ORIENTADO A LAS MIPYMES DE LA CIUDAD DE CHONE. *Uleam*, 1(1). <https://revistas.uleam.edu.ec/documentosbahia/EL%20IMPACTO%20DE%20LAS%20APLICACIONES%20MOVILES,%20ORIENTADO%20A%20LAS%20MIPYMES%20DE%20LA%20CIUDAD%20DE%20CHONE.pdf>
- Petru, P. (25 de Septiembre de 2023). ¿Qué es R-CNN? <https://blog.roboflow.com/what-is-r-cnn/>
- Piedad, A., Hernández, D., Lárraga, H., & Zacarías, E. (noviembre de 2020). TELEDETECCIÓN EN LA AGRICULTURA DE PRECISIÓN: ESTADO DEL ARTE DE LOS ÍNDICES DE VEGETACIÓN. *Revista de divulgación científica y tecnológica.*, 6(2), 46-58.
- Ponce, M. (2012). *Visión por Computador para UAS*. Universidad Politécnica de Madrid. https://doi.org/https://oa.upm.es/14316/2/PFC_MONICO_MANUEL_PONCE_GONZALEZ_B.pdf
- Rambauth, G. (june de 2022). Agricultura de Precisión: La integración de las TIC en la producción Agrícola. *Universidad de Sucre*, 3(1), 34-38. <https://revistascientificas.cuc.edu.co/CESTA/article/view/3978/4013>

- Ramos, M. (2024). Arveja. *Faqs.Zone*. <https://faqs.zone/arveja/>
- Rendón, Y. (28 de mayo de 2019). *Bases de datos relacionales vs. no relacionales*. <https://www.pragma.com.co/academia/lecciones/bases-de-datos-relacionales-vs.-no-relacionales#:~:text=Las%20bases%20de%20datos%20relacionales%20se%20basan%20en%20la%20organizaci%C3%B3n,o%20m%C3%A1s%20conjuntos%20de%20datos.>
- Ríos, R. (marzo de 2020). La Agricultura de Precisión. Una necesidad actual. *Revista Ingeniería Agrícola*, 11(1), 67-73.
- Rus, E. (1 de noviembre de 2020). *Investigación de campo*. <https://economipedia.com/definiciones/investigacion-de-campo.html>
- Saavedra, G. (2019). Arveja verde. Instituto de Investigaciones Agropecuarias INIA / MINISTERIO DE AGRICULTURA. <https://biblioteca.inia.cl/server/api/core/bitstreams/ce9665c6-c4a5-406c-a819-cb4eac975d29/content>
- Sampedro, C., Amable, S., & Patrón, E. (2016). DESARROLLO DE APLICACIONES DE ESCRITORIO HÍBRIDAS CON JAVASCRIPT, CSS Y HTML. *Mikarimin. Revista Científica Multidisciplinaria*, 2(2), 85-94. <https://revista.uniandes.edu.ec/ojs/index.php/mikarimin/article/view/380>
- Sampieri, R. (2014). *Metodología de la investigación. Sexta edición*. Mexicana. https://apiperiodico.jalisco.gob.mx/api/sites/periodicooficial.jalisco.gob.mx/files/metodologia_de_la_investigacion_-_roberto_hernandez_sampieri.pdf
- Sánchez, E. (2007). LAS TECNOLOGÍAS DE INFORMACIÓN Y COMUNICACIÓN (TIC) DESDE UNA PERSPECTIVA SOCIAL. *Revista Educare*, 12(Extraordinario), 155 - 162. <https://www.redalyc.org/pdf/1941/194114584020.pdf>
- Sánchez, J. (2023). Análisis comparativo de IDEs enfocados a Machine Learning. *Esprint Investigación*, 2(2), 5-13. <file:///C:/Users/NAYE/Downloads/Dialnet-AnalisisComparativoDeIDEsEnfocadosAMachineLearning-9698311.pdf>
- Silva, G., Chicaiza, D., Córdova, A., & Villagoméz, E. (junio de 2022). Implementación de una aplicación móvil informativa en tiempo real para el centro de propaganda y comunicación social. *Polo del Conocimiento*, 7(6), 1107-1121. <https://doi.org/10.23857/pc.v7i6.4124>


- Toapanta, V., Gavilanes, J., Tiban, A., & De la Torre, L. (2022). Aplicación web - móvil para la gestión de productores agropecuarios del gobierno autónomo descentralizado del cantón Mocha. *Revista Universal y Sociedad*, 14(3). [http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S2218-36202022000300487#:~:text=Flutter%20es%20un%20SDK%20\(Software,para%20Android%20y%20para%20iOS.](http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S2218-36202022000300487#:~:text=Flutter%20es%20un%20SDK%20(Software,para%20Android%20y%20para%20iOS.)
- Tyagi, A. (10 de enero de 2025). *¿Qué es Google Gemini? Características, uso y limitaciones.* Analytics Vidhya: <https://www.analyticsvidhya.com/blog/2023/12/what-is-google-gemini-features-usage-and-limitations/>
- Universidad Europea. (5 de julio de 2023). *Usos de Python: ¿qué ofrece este lenguaje de programación?* <https://universidadeuropea.com/blog/usos-python/>
- Valencia, F. (1 de diciembre de 2018). *Prevención y control de antracnosis en arveja: Ascochyta blight in faba bean.* <https://plantwiseplusknowledgebank.org/doi/10.1079/pwkb.20197800039>
- Valencia, F. (1 de December de 2018). *Prevención y control del oidium en arveja: Erysiphe polygoni. Factsheet for Farmers.* <https://plantwiseplusknowledgebank.org/doi/full/10.1079/pwkb.20197800040#con>
- Vina, A., & Vina, A. (7 de junio de 2024). *¿Qué es R-CNN? Una descripción general rápida.* <https://www.ultralytics.com/es/blog/what-is-r-cnn-a-quick-overview>
- Worsley, S. (11 de septiembre de 2024). *¿Qué es Python? Todo lo que necesitas saber para empezar.* <https://www.datacamp.com/es/blog/all-about-python-the-most-versatile-programming-language>
- Zambrano, D. (2024). *Desarrollo de una aplicación móvil para la detección temprana de plagas en cultivos de maíz (zea mays l.) Empleando técnicas de visión artificial basada en aprendizaje profundo.[Tesis de maestría].* (Q. UTEQ, Ed.) UNIVERSIDAD TÉCNICA ESTATAL DE QUEVEDO. <https://repositorio.uteq.edu.ec/items/ce349a42-eaf5-45b4-84ae-0b58ef492452>
- Zambrano, P., Bazurto, L., Rodríguez, T., & Macías, G. (2025). El desarrollo de interfaces de programación de aplicaciones (APIs) dinamiza el acceso a contenidos en plataformas de educación virtual. *LATAM Revista Latinoamericana de Ciencias Sociales y Humanidades, Asunción, Paraguay*, VI. <https://doi.org/DOI:https://doi.org/10.56712/latam.v6i2.3816>

VII. ANEXOS

Anexo 1. Acta de la sustentación de Predefensa del TIC



UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI



FACULTAD DE INDUSTRIAS AGROPECUARIAS Y CIENCIAS AMBIENTALES
CARRERA DE COMPUTACIÓN
ACTA

DE LA SUSTENTACIÓN ORAL DE LA PREDEFENSA DEL TRABAJO DE INTEGRACIÓN CURRICULAR CON ENFOQUE EN INVESTIGACIÓN

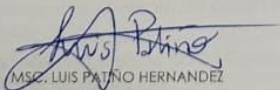
| | |
|---|---|
| ESTUDIANTE: Castro Pérez Washington Patricio | CÉDULA DE IDENTIDAD: 0402012264 |
| PERIODO ACADÉMICO: PAO 2025B | |
| PRESIDENTE TRIBUNAL: MSC. LUIS PATIÑO HERNANDEZ | DOCENTE TUTOR: MSC. CARLITOS GUANO CARDENAS |
| DOCENTE: MSC. JEFFERY NARANJO CEDEÑO | |
| TEMA DEL TIC: Visión por computadora para la detección de enfermedades en cultivo de la Pisum Sativum L. (Arveja) | |

| No. | CATEGORÍA | Evaluación cuantitativa | OBSERVACIONES Y RECOMENDACIONES |
|-----|---|-------------------------|---|
| 1 | PROBLEMA - OBJETIVOS | 7,33 | Revisar el planteamiento del problema y objetivos. |
| 2 | FUNDAMENTACIÓN TEÓRICA | 10,00 | |
| 3 | METODOLOGÍA | 10,00 | |
| 4 | RESULTADOS | 8,50 | Mayor énfasis en el entrenamiento del algoritmo. |
| 5 | DISCUSIÓN | 8,00 | Profundizar la discusión en función de resultados y antecedentes. |
| 6 | CONCLUSIONES Y RECOMENDACIONES | 9,00 | |
| 7 | DEFENSA, ARGUMENTACIÓN Y VOCABULARIO PROFESIONAL | 7,00 | Enfocarse en los resultados y funcionamiento del algoritmo. Mejorar la argumentación. |
| 8 | FORMATO, ORGANIZACIÓN Y CALIDAD DE LA INFORMACIÓN | 8,00 | Revisar la redacción y ortografía |


Obteniendo una nota de: **8,38** Por lo tanto, **APRUEBA**; debiendo el o los investigadores acatar el siguiente artículo:

Art. 66.- De la aprobación de la pre defensa del Informe final de TIC.- El estudiante deberá obtener una nota mínima de 7/10; al finalizar el proceso de pre-defensa se procederá a levantar el acta correspondiente. En el caso de aprobar con observaciones el estudiante deberá adjuntar el Informe final de cumplimiento de observaciones y recomendaciones emitido por el Tribunal previo a la defensa final en un término máximo de 10 días.

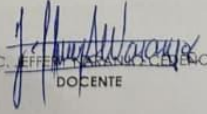
Para constancia del presente, firman en la ciudad de Tulcán el **jueves, 11 de diciembre de 2025**



MSC. LUIS PATIÑO HERNANDEZ
PRESIDENTE TRIBUNAL



MSC. CARLITOS GUANO CARDENAS
DOCENTE TUTOR



MSC. JEFFERY NARANJO CEDEÑO
DOCENTE

Anexo 2. Certificado del abstract por parte de idiomas



UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI- FOREIGN AND NATIVE LANGUAGES CENTER

Informe sobre el Abstract de Artículo Científico o Investigación.

Autor: Washington Patricio Castro Perez

Fecha de recepción del abstract: Miércoles, 17 de diciembre de 2025

Fecha de entrega del informe: Lunes, 22 de diciembre de 2025

El presente informe validará la traducción del idioma español al inglés si alcanza un porcentaje de: 9 – 10 Excelente.

Si la traducción no está dentro de los parámetros de 9 – 10, el autor deberá realizar las observaciones presentadas en el ABSTRACT, para su posterior presentación y aprobación.

Observaciones:

Después de realizar la revisión del presente abstract, éste presenta una apropiada traducción sobre el tema planteado en el idioma Inglés. Según la rúbrica de evaluación de la traducción en Inglés, ésta alcanza un valor de 9; por lo cual se valida dicho trabajo.

Atentamente



MA. Martha Viveros
Responsable del
CIDEN

Anexo 3. Toma de muestras



Figura 53. Comunidad de Chután Bajo" El Falso"



Figura 54. Sector "Cascada de Paluz"



Figura 55. Comunidad de Cumbaltar

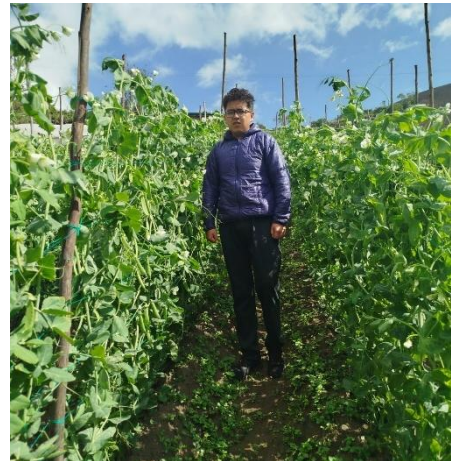


Figura 56. Comunidad de Tangüis



Figura 57. Comunidad de Tangüis Alto

Anexo 4. Preguntas de evaluación y aprobación



Universidad Politécnica Estatal del Carchi
Carrera de Computación



Preguntas de Evaluación - APLICACION ARVEJA CHECK

| Preguntas | Cumplimiento | | Observaciones |
|--|--------------|----|---------------|
| | Si | No | |
| 1. ¿El sistema de login y registro funciona correctamente? (Ingreso de credenciales, validación de email/contraseña, creación de nueva cuenta) | ✓ | | |
| 2. ¿Las tres opciones de detección funcionan adecuadamente? (Capturar foto con cámara, Seleccionar de galería, Análisis en tiempo real) | ✓ | | |
| 3. ¿El sistema detecta correctamente las tres enfermedades (Antracnosis, Fusarium, Oídio) y proporciona recomendaciones de tratamiento con IA? | ✓ | | |
| 4. ¿El menú principal permite acceder correctamente a todas las funciones? (Detección, Cómo Funciona, Información, Asistente Virtual, Clima, Calendario Lunar, Historial) | ✓ | | |
| 5. ¿El asistente virtual ARVEJA CHECK-AI responde adecuadamente preguntas sobre cultivo de arveja? ¿Funciona el análisis de fotos y la función de audio? | ✓ | | |
| 6. ¿El módulo de clima muestra correctamente temperatura, humedad, viento y pronóstico de 7 días basado en ubicación GPS? 7. ¿El calendario lunar muestra las fases correctamente y proporciona recomendaciones IA con Gemini basadas en ubicación GPS? | ✓ | | |
| 8. ¿El historial guarda todas las detecciones con imagen, porcentaje de confianza, fecha y recomendación? ¿Se pueden visualizar, eliminar y están ordenadas por fecha? | ✓ | | |



| | | | |
|---|---|--|--|
| 9. ¿El botón azul de impresión genera correctamente reportes PDF con toda la información (logo, imagen, confianza, recomendaciones, fecha) y permite imprimir directamente? | ✓ | | |
| 10. ¿La guía de usuario funciona correctamente? (Navegación entre pasos, función de texto a voz, zoom e interacción con las imágenes) | ✓ | | |

Revisor por: Mgs. David Herrera

Autor: Washington Castro

.....
FIRMA

.....
FIRMA

Anexo 5. Manual de usuario

Universidad Politécnica estatal del Carchi
Facultad de industrias Agropecuarias y Ciencias Ambientales
Carrera de Computación



Manual de Usuario
ARVEJA CHECK



Sistema Inteligente de Detección de Enfermedades en Arveja de Amarre

Versión:

1.0

Desarrollador: Washington Patricio Castro Pérez

ÍNDICE

| | |
|--|-----|
| 1. Introducción | 108 |
| 1.1. Propósito | 108 |
| 1.2. Alcance | 108 |
| 1.3. Requisitos del Sistema | 109 |
| 2. Acceso al Sistema..... | 109 |
| 2.1. Pantalla de Login | 109 |
| 2.2. Registro de Usuario..... | 111 |
| 2.3. Pantalla de Bienvenida | 112 |
| 3. Menú Principal..... | 113 |
| 4. Detección de Enfermedades | 114 |
| 4.1. Capturar Imagen | 114 |
| 4.2. Seleccionar desde Galería | 115 |
| 4.3. Análisis en Tiempo Real..... | 115 |
| 4.4. Interpretación de Resultados..... | 115 |
| 5. Asistente Virtual ARVEJA CHECK-AI | 116 |
| 5.1. Realizar Consultas | 117 |
| 5.2. Escuchar Respuestas en Audio | 117 |
| 6. Información del Clima | 118 |
| 6.1. Consultar Datos Meteorológicos | 118 |
| 6.2. Pronóstico de 7 Días..... | 119 |
| 7. Calendario Lunar IA..... | 119 |
| 7.1. Consultar Fases Lunares..... | 120 |
| 7.2. Recomendaciones IA | 120 |
| 8. Historial de Detecciones..... | 120 |
| 8.2. Ver Detalles de Detección | 121 |
| 8.3. Agregar Fotos Adicionales | 122 |
| 9. Generación de Reportes | 123 |

| | |
|--|-----|
| 9.1. Imprimir Reporte Directamente | 123 |
| 9.2. Generar Reporte PDF..... | 124 |
| 9.3. Compartir Reportes..... | 124 |
| 10. Guía Interactiva (Cómo Funciona) | 126 |
| 11. Información del Desarrollador | 128 |

1. Introducción

El presente manual de usuario está diseñado para proporcionar una guía clara y detallada sobre el uso de la aplicación móvil ARVEJA CHECK, desarrollada para la detección inteligente de enfermedades en cultivos de arveja de amarre. Este documento tiene como objetivo facilitar la comprensión y utilización de las diferentes funcionalidades del sistema, garantizando una experiencia óptima para agricultores y técnicos agrícolas.

Se explican las principales características de la aplicación, la estructura de las interfaces y los pasos necesarios para realizar tareas clave, como la detección de enfermedades mediante inteligencia artificial, consulta del clima, uso del calendario lunar y gestión del historial de análisis.

1.1. Propósito

Este manual de usuario proporciona una guía completa sobre el uso de la aplicación móvil ARVEJA CHECK, un sistema inteligente diseñado para la detección de enfermedades en cultivos de arveja de amarre mediante tecnología de inteligencia artificial. Está dirigido a agricultores, técnicos agrícolas e ingenieros agrónomos que deseen aprovechar esta herramienta tecnológica para el manejo eficiente de sus cultivos.

1.2. Alcance

ARVEJA CHECK es una aplicación móvil integral que ofrece:

- Detección automática de enfermedades (Antracnosis, Fusarium, Oídio y Mildu).
- Asistente virtual especializado con inteligencia artificial.

- Información climática en tiempo real.
- Calendario lunar con recomendaciones agrícolas.
- Historial completo de análisis realizados.
- Generación de reportes profesionales en PDF.
- Función de impresión directa.

1.3. Requisitos del Sistema

Requisitos mínimos:

- Sistema Operativo: Android 5.0+ o iOS 11.0+
- Memoria RAM: 2 GB mínimo.
- Almacenamiento: 100 MB disponibles.
- Cámara: 8 MP mínimo.
- Conexión a Internet (WiFi o datos móviles).

Permisos necesarios:

- Acceso a cámara.
- Acceso a galería de fotos.
- Acceso a ubicación GPS.
- Acceso a almacenamiento.
- Acceso a audio (para funciones de voz).

2. Acceso al Sistema

2.1. Pantalla de Login

La pantalla de login es el punto de entrada a ARVEJA CHECK.



Pasos para Iniciar Sesión:

1. Abra la aplicación ARVEJA CHECK en su dispositivo móvil
2. Ingrese su correo electrónico en el campo "Email"
3. Ingrese su contraseña en el campo "Contraseña"
4. Presione el botón "INICIAR SESIÓN"
5. El sistema validará sus credenciales y le dará acceso

Nota: Si es su primera vez usando la aplicación, deberá registrarse primero.

2.2. Registro de Usuario



Si no tiene una cuenta, siga estos pasos:

1. En la pantalla de login, presione "¿No tienes cuenta? Regístrate"
2. Complete el formulario con sus datos:
 - Nombre completo
 - Correo electrónico
 - Contraseña (mínimo 8 caracteres)
 - Confirmar contraseña
3. Presione el botón "REGISTRARSE"
4. El sistema creará su cuenta y le dará acceso automático

2.3. Pantalla de Bienvenida



Después de iniciar sesión, verá la pantalla de bienvenida con:

- Mensaje de bienvenida personalizado
- Logo animado de ARVEJA CHECK
- Información sobre las funcionalidades principales
- Botón para continuar al menú principal

3. Menú Principal



El menú principal les da acceso a todas las funcionalidades de ARVEJA CHECK:

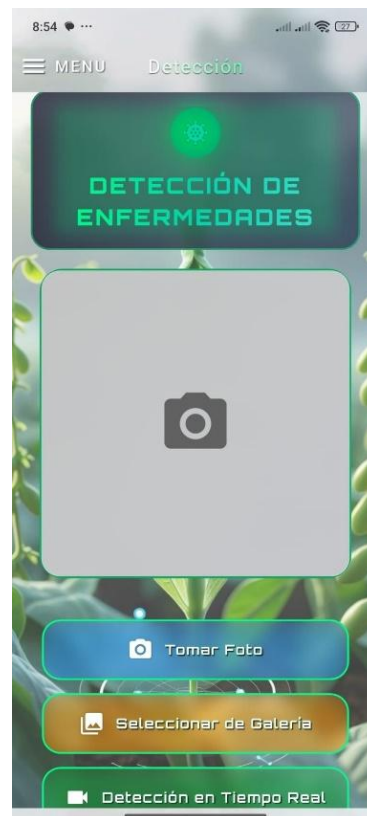
Opciones Disponibles:

- **Detección:** Analizar enfermedades en sus plantas
- **Cómo Funciona:** Guía interactiva de usuario
- **Información:** Datos del desarrollador
- **Asistente Virtual:** Chatbot especializado en arveja
- **Clima:** Información meteorológica actual
- **Calendario Lunar:** Fases lunares y recomendaciones
- **Historial:** Registro de detecciones anteriores

Navegación:

- Toque cualquier opción para acceder a esa funcionalidad
- Use el botón de retroceso para volver al menú principal
- Deslice horizontalmente para explorar opciones

4. Detección de Enfermedades



Esta es la función principal de ARVEJA CHECK para identificar enfermedades en sus plantas.

4.1. Capturar Imagen

Pasos:

1. En el menú principal, seleccione "Detección"
2. Presione el botón "CAPTURAR"
3. El sistema activará la cámara de su dispositivo
4. Apunte la cámara hacia la hoja o planta afectada
5. Asegúrese de tener buena iluminación natural
6. Mantenga una distancia de 15-30 cm de la planta
7. Presione el botón circular para tomar la foto
8. Confirme la imagen capturada

Recomendaciones:

- Use luz natural para mejores resultados

- Enfoque la zona afectada de la planta
- Evite sombras y reflejos
- Mantenga el dispositivo estable

4.2. Seleccionar desde Galería

Pasos:

1. En el módulo de detección, presione "GALERÍA"
2. El sistema abrirá la galería de fotos de su dispositivo
3. Navegue por sus carpetas y seleccione la imagen deseada
4. Presione "Seleccionar" o "Aceptar"
5. El sistema cargará la imagen para análisis

Formatos Aceptados:

- JPG, JPEG, PNG
- Tamaño máximo: 10 MB
- Resolución mínima recomendada: 800x600 px

4.3. Análisis en Tiempo Real

También puede realizar análisis en tiempo real:

1. Seleccione la opción "TIEMPO REAL"
2. Apunte la cámara hacia la planta
3. El sistema analizará continuamente lo que ve la cámara
4. Cuando detecte una enfermedad, mostrará el resultado en pantalla

4.4. Interpretación de Resultados

Una vez que el sistema procese la imagen, verá:

Pantalla de Resultados:

1. **Enfermedad Detectada**
 - Nombre de la enfermedad (Antracnosis, Fusarium u Oídio)
 - Icono representativo
2. **Nivel de Confianza**
 - Porcentaje de certeza del análisis (0-100%)
 - Interpretación:
 - ✓ 90-100%: Muy alta confianza

- ✓ 70-89%: Alta confianza
- ✓ 50-69%: Confianza moderada
- ✓ Menos de 50%: Se recomienda verificar

3. Imagen Analizada

- Vista de la foto procesada

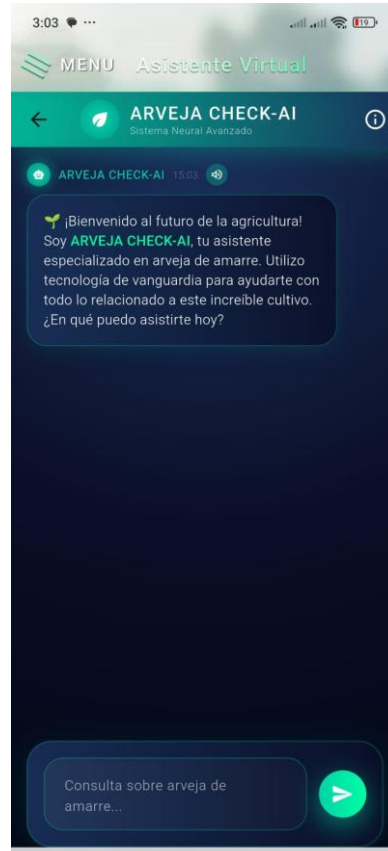
4. Recomendaciones

- Tratamiento químico sugerido
- Manejo cultural recomendado
- Medidas de prevención
- Frecuencia de monitoreo

Acciones Disponibles:

- El sistema presenta la opción de guardar al en el historial
- Puede compartir los resultados
- Puede consultar al Asistente Virtual
- Puede realizar una nueva detección

5. Asistente Virtual ARVEJA CHECK-AI



Un chat bot especializado con inteligencia artificial para responder sus consultas sobre arveja de amarre.

5.1. Realizar Consultas

Pasos:

- 1.- En el menú principal, seleccione "Asistente Virtual"
- 2.- Escriba su pregunta en el campo de texto
- 3.- Presione el botón de enviar (ícono de avión)
- 4.- El asistente procesará su consulta
- 5.- Recibirá una respuesta especializada en segundos

Ejemplos de Consultas:

- "¿Cuál es el mejor momento para sembrar arveja?"
- "¿Cómo prevenir el oídio en mis plantas?"
- "¿Qué fertilizante debo usar en arveja?"
- "¿Cuándo debo cosechar la arveja?"

5.2. Escuchar Respuestas en Audio

Las respuestas pueden ser escuchadas en voz alta:

1. Junto a cada respuesta del asistente verá un ícono de altavoz.
2. Presione el ícono para escuchar la respuesta
3. Presione nuevamente para detener el audio
4. La voz está en español y a velocidad moderada

Nota: El asistente está especializado únicamente en arveja de amarre y no responde consultas sobre otros temas.

6. Información del Clima



Consulte datos meteorológicos en tiempo real basados en su ubicación GPS.

6.1. Consultar Datos Meteorológicos

Pasos:

1. En el menú principal, seleccione "Clima"
2. Permita el acceso a su ubicación GPS (solo la primera vez)
3. El sistema mostrará información del clima actual:
 - Temperatura actual
 - Sensación térmica
 - Humedad relativa
 - Precipitación
 - Velocidad y dirección del viento

- Presión atmosférica
- Índice UV

6.2. Pronóstico de 7 Días

En la misma pantalla puede ver:

- Pronóstico extendido para la próxima semana
- Temperaturas máximas y mínimas por día
- Probabilidad de lluvia
- Condiciones climáticas esperadas (soleado, nublado, lluvia)
- Recomendaciones para actividades agrícolas

Aplicaciones Agrícolas:

- Planificar riegos según la precipitación esperada
- Programar aplicaciones de agroquímicos en días sin viento
- Preparar el campo según condiciones de humedad

7. Calendario Lunar IA



Consulte las fases lunares y reciba recomendaciones agrícolas basadas en IA.

7.1. Consultar Fases Lunares

Pasos:

1. En el menú principal, seleccione "Calendario Lunar"
2. El sistema mostrará:
 - Fase lunar actual (Nueva, Creciente, Llena, Menguante)
 - Porcentaje de iluminación
 - Edad lunar (días desde luna nueva)
 - Hora de salida y puesta de la luna
 - Distancia Tierra-Luna

7.2. Recomendaciones IA

La inteligencia artificial Gemini genera recomendaciones específicas:

Información Mostrada:

Recomendaciones para Hoy

- Actividades agrícolas favorables
- Actividades que debe evitar
- Basadas en la fase lunar actual

Ejemplos de Recomendaciones:

- Durante luna creciente: Siembra de arveja, aplicación de fertilizantes
- Durante luna llena: Crecimiento vegetativo, aplicación de bioestimulantes
- Durante luna menguante: Desarrollo radicular, control de malezas
- Durante luna nueva: Preparación de suelo, cosecha

Nota: Las recomendaciones consideran su ubicación GPS y la época del año específica.

8. Historial de Detecciones

Consulte todas las detecciones que ha realizado anteriormente.

8.1. Ver Historial

2. Se abrirá la vista detallada con:

- Imagen completa en alta resolución (puede hacer zoom)
- Resultados completos del análisis
- Recomendaciones detalladas
- Condiciones climáticas del momento
- Fase lunar del momento
- Ubicación GPS donde se realizó

8.3. Agregar Fotos Adicionales

Puede documentar la evolución de la enfermedad:

1. En la vista detallada de una detección, busque el botón "Agregar Fotos"
2. Capture o seleccione fotos adicionales
3. Las fotos se agregarán al reporte de esa detección
4. Útil para hacer seguimiento del tratamiento

Otras Acciones Disponibles:

- **Ver detalles** - Información completa
- **Eliminar** - Borrar la detección del historial
- El sistema pedirá confirmación antes de eliminar

9. Generación de Reportes



Cree documentos profesionales de sus detecciones para compartir o imprimir.

9.1. Imprimir Reporte Directamente

Función de Impresión Directa:

- En la lista del historial, busque el ícono de impresora azul en cada detección
- Toque el ícono de impresora
- El sistema enviará el reporte directamente a su impresora configurada
- Seleccione la impresora disponible
- Configure opciones de impresión (copias, color, etc.)
- Confirme la impresión

Contenido del Reporte Impreso:

- Encabezado con logo ARVEJA CHECK
- Información completa de la detección
- Imagen analizada
- Nivel de confianza

- Recomendaciones de tratamiento
- Fecha y hora del análisis
- Información del desarrollador

Nota: Asegúrese de tener una impresora configurada en su dispositivo o acceso a una impresora WiFi.

9.2. Generar Reporte PDF

Si prefiere crear un documento PDF:

- 1.- En la vista detallada de una detección o en el historial
- 2.- Busque el ícono rosa de PDF.
- 3.- Toque el ícono "Generar PDF".
- 4.- El sistema creará un documento PDF profesional
- 5.- Verá una vista previa del PDF generado

El PDF incluye:

- Diseño profesional con branding ARVEJA CHECK
- Toda la información de la detección
- Imágenes en alta calidad
- Gráficos de confianza
- Recomendaciones detalladas
- Datos del análisis (fecha, ubicación, clima)

9.3. Compartir Reportes

Una vez generado el PDF, puede:

1. **Compartir por Email**

- Toque el botón de compartir
- Seleccione su aplicación de email
- El PDF se adjuntará automáticamente
- Agregue destinatarios y envíe

2. **Compartir por WhatsApp**

- Seleccione WhatsApp en las opciones
- Elija el contacto o grupo
- El PDF se enviará como documento

3. **Guardar en el Dispositivo**

- Elija "Guardar en..."
- Seleccione la ubicación (Descargas, Drive, etc.)
- El PDF quedará guardado para uso posterior

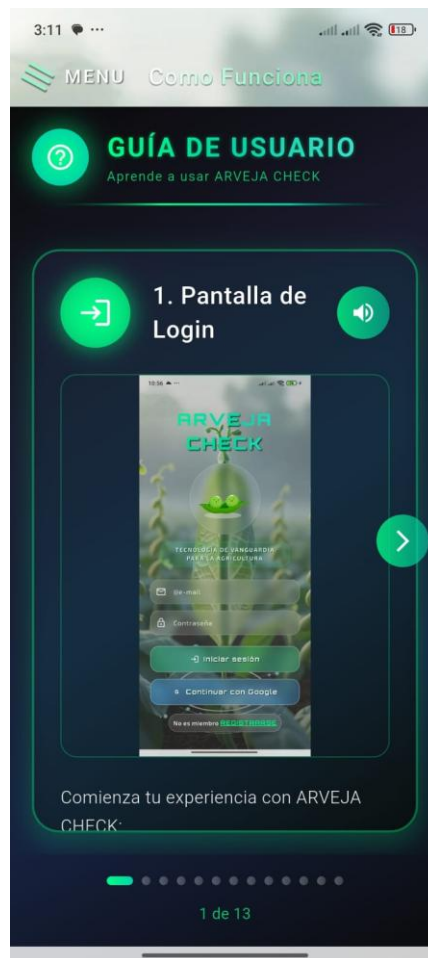
4. **Otras Opciones**

- Compartir por otras aplicaciones de mensajería
- Subir a servicios en la nube
- Enviar por cualquier app instalada

Usos Recomendados:

- Consultar con agrónomos profesionales
- Documentación de campo
- Reportes a autoridades
- Registro de actividades agrícolas

10. Guía Interactiva (Cómo Funciona)



ARVEJA CHECK incluye una guía interactiva paso a paso.

Acceso:

1. En el menú principal, seleccione "Cómo Funciona"
2. Se abrirá la guía interactiva con 13 pasos

Características de la Guía:

1. **Navegación:**
 - Deslice horizontalmente entre pasos
 - Use las flechas laterales para avanzar o retroceder
 - Los indicadores en la parte inferior muestran su progreso
2. **Contenido de Cada Paso:**
 - Título descriptivo

- Imagen de referencia (puede hacer zoom)
- Explicación detallada de la funcionalidad
- Íconos representativos

3. **Audio Narración:**

- Cada paso tiene un ícono de audio.
- Toque el ícono para escuchar la explicación
- Perfecto para aprender mientras trabaja en el campo

4. **Temas Cubiertos:**

- Pantalla de Login y Registro
- Bienvenida a ARVEJA CHECK
- Detección de Enfermedades
- Menú Principal
- Asistente Virtual
- Información del Clima
- Calendario Lunar IA
- Historial de Detecciones
- Cómo Guardar en el Historial
- Imprimir Reportes
- Visualizar PDFs Generados
- Información del Usuario

Recomendación: Revise toda la guía la primera vez que use la aplicación para familiarizarse con todas las funcionalidades.

11. Información del Desarrollador



Conozca al creador de ARVEJA CHECK:

Acceso:

1. En el menú principal, seleccione "Información"
2. Verá los datos del desarrollador

Información Mostrada:

- **Nombre:** Washington Patricio Castro Pérez
- **Profesión:** Desarrollador de Aplicaciones Móviles
- **Especialización:** Inteligencia Artificial Aplicada a la Agricultura
- **País:** Ecuador ec
- **Contacto:** Información de email disponible en la aplicación

Sobre la Aplicación:

- ARVEJA CHECK es tecnología ecuatoriana
- Desarrollada específicamente para agricultores de arveja de amarre
- Utiliza modelos de IA entrenados con datos locales
- Actualización constante de funcionalidades
- Soporte técnico disponible

Glosario de Términos

Antracnosis: Enfermedad fúngica que causa manchas circulares oscuras en hojas y vainas.

Fusarium: Hongo que causa marchitamiento vascular en plantas de arveja.

Oídio: Enfermedad fúngica que forma un polvo blanco en hojas y tallos.

IA (Inteligencia Artificial): Tecnología que permite a las máquinas aprender y tomar decisiones.

GPS: Sistema de posicionamiento global para determinar ubicación geográfica.

PDF: Formato de documento portátil para compartir archivos.

Confianza: Nivel de certeza del análisis de IA expresado en porcentaje.

Base64: Formato de codificación de imágenes usado internamente.

Solución de Problemas Comunes

Problema: No puedo iniciar sesión

Solución: Verifique su conexión a internet, revise que el email y contraseña sean correctos.

Problema: La cámara no funciona

Solución: Verifique los permisos de la aplicación en configuración de su dispositivo.

Problema: El análisis tarda mucho

Solución: Asegúrese de tener buena conexión a internet. El análisis requiere conectividad.

Problema: No puedo ver el historial

Solución: Verifique los permisos de almacenamiento de la aplicación.

Problema: El PDF no se genera

Solución: Verifique que tiene espacio disponible en su dispositivo.

Problema: El asistente virtual no responde

Solución: Revise su conexión a internet. El chatbot requiere conexión activa.

Para soporte adicional o reportar problemas, contacte al desarrollador a través de la sección "Información" de la aplicación.