

UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI



FACULTAD DE INDUSTRIAS AGROPECUARIAS Y CIENCIAS AMBIENTALES

CARRERA DE COMPUTACIÓN

Tema: “Visión artificial en la agricultura de precisión para la identificación de enfermedades en cultivos de arveja Quantum”

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

AUTORES: Pérez Enríquez Kimberlhy Nayelly.

Tutachá Delgado Wilmer Orlando.

TUTOR: Ing. Guano Cárdenas Carlitos Alberto, MSc

Tulcán, 2024.

CERTIFICADO DEL TUTOR

Certifico que los estudiantes(s) Pérez Enríquez Kimberlhy Nayelly y Tutachá Delgado Wilmer Orlando con el número de cédula 0401677901 y 0401933031 respectivamente han desarrollado el Trabajo de Integración Curricular: "Visión artificial en la agricultura de precisión para la identificación de enfermedades en cultivos de arveja Quantum"

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 Cárdenas Carlitos Alberto, Msc

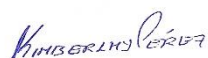
TUTOR

Tulcán, junio de 2024

AUTORÍA DE TRABAJO

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

Nosotros, Pérez Enríquez Kimberlhy Nayelly y Tutachá Delgado Wilmer Orlando con cédula de identidad número 0401677901 y 0401933031 respectivamente declaramos que la investigación es absolutamente original, auténtica, personal y los resultados y conclusiones a los que hemos llegado son de nuestra absoluta responsabilidad.



Pérez Enríquez Kimberlhy Nayelly

AUTORA




Tutachá Delgado Wilmer Orlando

AUTOR

Tulcán, junio de 2024

ACTA DE CESIÓN DE DERECHOS DEL TRABAJO DE INTEGRACIÓN CURRICULAR

Nosotros Pérez Enríquez Kimberlhy Nayelly y Tutachá Delgado Wilmer Orlando declaramos ser autor de los criterios emitidos en el Trabajo de Integración Curricular: "Visión artificial en la agricultura de precisión para la identificación de enfermedades en cultivos de arveja Quantum" y se exime expresamente a la Universidad Politécnica Estatal del Carchi y a sus representantes de posibles reclamos o acciones legales.



Pérez Enríquez Kimberlhy Nayelly

AUTORA



Tutachá Delgado Wilmer Orlando

AUTOR

Tulcán, junio de 2024

AGRADECIMIENTO

Deseamos extender nuestra profunda gratitud hacia todas aquellas personas y entidades que apoyaron a la creación de este trabajo de investigación. En primer lugar, extendemos nuestro reconocimiento al Msc. Carlitos Guano, nuestro tutor de tesis, cuya orientación invaluable, paciencia y apoyo han sido fundamentales a lo largo de todo el proceso de investigación. Sus conocimientos expertos y dedicación han dejado una marca significativa en el desarrollo de este trabajo.

Además, expresamos nuestra gratitud al Ing. Guillermo Jacome, quien nos brindó orientación en temas de agricultura para el proyecto. Su paciencia y compromiso fueron evidentes a lo largo del proceso, y sus enseñanzas y consejos han enriquecido e inspirado nuestro interés en esta área de estudio. Agradecemos también a todos nuestros amigos por su apoyo moral, ideas y experiencias compartidas, así como a nuestras familias por su amor, aliento y comprensión incondicionales durante este arduo proceso.

Asimismo, queremos reconocer el respaldo a la Universidad Politécnica Estatal del Carchi por habernos dado todo el apoyo para obtener el título de Ingeniería en Ciencias de la Computación.

Wilmer Tutachá y Kimberlhy Pérez

DEDICATORIA

Dedico este gran logro a mis padres José Tutachá, Rocío Delgado, quienes han sido el pilar fundamental a lo largo de mi formación. Gracias por depositar toda su confianza en mí y por todo el esfuerzo que han realizado pude completar mi carrera profesional. Ellos han sido mi fuerza y mi inspiración para seguir adelante, inculcándome valores, principios y brindándome consejos que han sido fundamentales en mi vida. Además, quiero dedicar este logro a mis queridos hermanos y hermanas: Luis, Guido, Nelly, Isabel y Germania. Ellos estuvieron presentes en todo momento, brindándome su apoyo incondicional durante los momentos difíciles de mi formación.

Asimismo, quiero expresar mi gratitud a mis amigos y seres queridos que, con su amistad sincera, han contribuido de manera significativa a mi proceso de crecimiento. Su apoyo ha sido un factor determinante en cada paso dado.

Tutachá Delgado Wilmer Orlando

Quiero expresar mi más profundo agradecimiento primeramente a mi madre Adriana Pérez por ser el motor más grande para realizar y cumplir esta meta, por ser incondicional en mi vida y por ser el motivo de superación y éxito, a mi abuelita Yolanda Enríquez por darme consejos y por no dejarme caer en cada paso que doy, a mi abuelito German Pérez por ser el más consentidor y por ser mi apoyo en todo lo que necesito, y a mi mascotita Keyla por ser la más fiel compañera, por estar conmigo en todo momento, por ser parte de los desvelos y llanos y por brindarme todo su cariño y amor en este tiempo.

Pérez Enríquez Kimberlhy Nayelly

ÍNDICE

RESUMEN	16
ABSTRACT	17
INTRODUCCIÓN	18
I. EL PROBLEMA	19
1.1. PLANTEAMIENTO DEL PROBLEMA	19
1.2. FORMULACIÓN DEL PROBLEMA	21
1.3. JUSTIFICACIÓN	21
1.4. OBJETIVOS Y PREGUNTAS DE INVESTIGACIÓN	22
1.4.1. Objetivo General.....	22
1.4.2. Objetivos Específicos.....	22
1.4.3. Preguntas de Investigación	22
II. FUNDAMENTACIÓN TEÓRICA	23
2.1. ANTECEDENTES DE LA INVESTIGACIÓN	23
2.2. MARCO TEÓRICO	25
2.2.1. Enfermedades en plantas	26
2.2.2. Enfermedades en leguminosas.....	27
2.2.3. Arveja Quantum.....	27
2.2.4. Visión artificial en la agricultura	33
2.2.5. Algoritmos de aprendizaje para detección de objetos en imágenes	33
2.2.6. Tecnología en la nube	39
2.2.7. Tecnologías de desarrollo	40
2.2.8. Manejadores de Base de Datos	42
2.2.9. Metodologías de desarrollo.....	45
III. METODOLOGÍA	48

3.1. ENFOQUE METODOLÓGICO	48
3.1.1. Enfoque	48
3.1.2. Tipo de Investigación.....	48
3.2. IDEA A DEFENDER	50
3.3. DEFINICIÓN Y OPERACIONALIZACIÓN DE LAS VARIABLES	51
3.4. MÉTODOS UTILIZADOS	53
3.4.1. Método deductivo	53
3.4.2. Método inductivo	53
3.4.3. Técnicas	54
3.5. ANÁLISIS ESTADÍSTICO	55
3.5.1. Población y muestra	55
3.6. RECURSOS	56
3.6.1. Recursos materiales.....	56
3.6.2. Recursos humanos.....	56
3.6.3. Recursos financieros	56
3.6.4. Recursos tecnológicos	57
IV. RESULTADOS Y DISCUSIÓN	58
4.1. RESULTADOS	58
4.1.1. Análisis de la encuesta	58
4.1.2. Propuesta.....	66
4.2. DISCUSIÓN	118
V. CONCLUSIONES Y RECOMENDACIONES	122
5.1. CONCLUSIONES	122
5.2. RECOMENDACIONES	122
VI. REFERENCIAS BIBLIOGRÁFICAS	124
VII. ANEXOS	130

ÍNDICE DE TABLAS

Tabla 1. Ventajas y características de la variedad de arveja Quantum	28
Tabla 2. Matriz de correlación entre enfermedad y causas	31
Tabla 3. Plagas y condiciones climáticas.....	32
Tabla 4. Enfermedades y efectos	32
Tabla 5. Comparativa de algoritmos de aprendizaje para detectar de objetos en imágenes.....	35
Tabla 6. Comparación de algoritmos.....	37
Tabla 7. Análisis del modelo MobileNetV2	39
Tabla 8. Lenguajes de programación.....	40
Tabla 9. Frameworks.	41
Tabla 10. Beneficios y consecuencias de PostgreSQL.....	43
Tabla 11. MySQL	43
Tabla 12. SQLserver	44
Tabla 13. Bases de Datos en aplicaciones móviles.....	45
Tabla 14. Metodologías de Desarrollo.	45
Tabla 15. Operacionalización de variables	51
Tabla 16. Recursos Humanos.....	56
Tabla 17. Recursos Financieros.....	56
Tabla 18. Roles.....	67
Tabla 19. Tiempo	67
Tabla 20. Módulos de la aplicación	67
Tabla 21. H_usuario 1	68
Tabla 22. H_usuario 2	68
Tabla 23. H_usuario 3	68
Tabla 24. H_usuario 4	68
Tabla 25. H_usuario 5	69
Tabla 26. H_usuario 6	69
Tabla 27. H_usuario 7	69
Tabla 28. H_usuario 8	69
Tabla 29. H_usuario 9	69
Tabla 30. H_usuario 10.....	70

Tabla 31. H_usuario 11	70
Tabla 32. T_usuario 1	70
Tabla 33. T_usuario 2.....	70
Tabla 34. T_usuario 3.....	70
Tabla 35. T_usuario 4.....	70
Tabla 36. T_usuario 5.....	71
Tabla 37. T_usuario 6.....	71
Tabla 38. T_usuario 7	71
Tabla 39. T_usuario 8.....	71
Tabla 40. T_usuario 9.....	71
Tabla 41. T_usuario 10.....	71
Tabla 42. T_usuario 11	71
Tabla 43. T_usuario 12.....	72
Tabla 44. T_usuario 13.....	72
Tabla 45. T_usuario 14.....	72
Tabla 46. Estimación de tiempo en tareas de usuario	72
Tabla 47. Plan de entrega del proyecto	73
Tabla 48. Pruebas de rendimiento	114
Tabla 49. Registro de usuario	115
Tabla 50. Registro con la cuenta de Google	115
Tabla 51. Restablecimiento de la contraseña.....	115
Tabla 52. Funcionamiento de opciones de navegación	115
Tabla 53. Captura de imágenes	116
Tabla 54. Selección de imágenes en galería	116
Tabla 55. Recorte de imágenes	116
Tabla 56. Envío de imagen al análisis	116
Tabla 57. Recepción de imagen etiquetada con resultados.....	116
Tabla 58. Cuadro de diálogo emergente con resultados.....	117
Tabla 59. Resumen iteraciones	117
Tabla 60. Historial revisiones.....	142
Tabla 61. Recursos humanos y profesionales.....	143
Tabla 62. Roles y responsabilidades.....	143

Tabla 63. Plan del proyecto.....	144
Tabla 64. Fases y líneas base	144
Tabla 65. Objetivos y riesgos de cada fase.....	144
Tabla 66. Diagrama de Gantt	145
Tabla 67. Calendario del proyecto	145
Tabla 68. Control de desviación a la planificación	145
Tabla 69. Matriz de riesgo	145
Tabla 70. Matriz de calor.....	146

ÍNDICE DE FIGURAS

Figura 1. Experiencia del agricultor	58
Figura 2. Hectáreas destinadas al cultivo.....	59
Figura 3. Variedades de arveja más comunes en la zona	59
Figura 4. Siembras al año	60
Figura 5. Enfermedades en el cultivo	60
Figura 6. Plagas más frecuentes en el cultivo	61
Figura 7. Monitoreo en el cultivo	61
Figura 8. Indicaciones técnicas	62
Figura 9. Formas para detectar una enfermedad en el cultivo.....	62
Figura 10. Recomendaciones para un cultivo libre de enfermedades	63
Figura 11. Uso de fungicidas para reducir enfermedades en el cultivo	64
Figura 12. Aplicativo para la detección de enfermedades.....	64
Figura 13. Uso de herramientas tecnológicas.....	65
Figura 14. Opciones del aplicativo.....	65
Figura 15. Uso de la visión artificial.....	66
Figura 16. Arquitectura cliente - servidor	74
Figura 17. Prototipo login	75
Figura 18. Prototipo menú inicio.....	75
Figura 19. Prototipo Manejo agronómico	76
Figura 20. Prototipo Taxonomía del cultivo	76
Figura 21. Prototipo requerimientos nutricionales.....	77
Figura 22. Prototipo requerimientos edafoclimáticos	77

Figura 23. Prototipo plagas	78
Figura 24. Prototipo de enfermedades	78
Figura 25. Prototipo cosecha	79
Figura 26. Prototipo información del cultivo	80
Figura 27. Prototipo preparación suelo	80
Figura 28. Prototipo siembra y germinación	81
Figura 29. Prototipo cuidado y riego	81
Figura 30. Prototipo síntomas característicos.....	82
Figura 31. Prototipo métodos de diagnóstico	82
Figura 32. Prototipo medidas de manejo y control	83
Figura 33. Prototipo cosecha y almacenamiento	83
Figura 34. Prototipo variedad de arvejas	84
Figura 35. Prototipo épocas de siembra y regiones	84
Figura 36. Prototipo buenas prácticas de cultivo	85
Figura 37. Prototipo detección enfermedades.....	85
Figura 38. Prototipo control antracnosis	86
Figura 39. Prototipo control fusarium	86
Figura 40. Prototipo control oídio	87
Figura 41. Prototipo clima.....	87
Figura 42. Prototipo perfil de usuario.....	88
Figura 43. Caso de uso del agricultor	88
Figura 44. Herramienta SDK.....	89
Figura 45. Autenticación de usuarios a través de Firebase	89
Figura 46. Dataset de imágenes	90
Figura 47. Instalación de la librería de TensorFlow	90
Figura 48. Asignamos la dirección del directorio de la dataset	91
Figura 49. Cantidad de imágenes	91
Figura 50. Etiquetas de cada patología.	91
Figura 51. Modelo de red neuronal	92
Figura 52. Proceso de entrenamiento del modelo	92
Figura 53. Número de épocas	92
Figura 54. Progreso de entrenamiento	92

Figura 55. App de clasificación	93
Figura 56. Dataset de imágenes.....	94
Figura 57. Preparación para etiquetar imágenes de enfermedades	95
Figura 58. Etiquetado de imágenes	95
Figura 59. Archivo data.yaml	96
Figura 60. Instalación y uso de la biblioteca Ultralytics.....	96
Figura 61. Entorno para entrenar un modelo con Roboflow	96
Figura 62. Número de épocas del algoritmo	97
Figura 63. Rendimiento de la detección en diferentes clases	97
Figura 64. Test del modelo para la detección de antracnosis en arveja	98
Figura 65. Rendimiento del modelo Yolov8 en imágenes de prueba.....	98
Figura 66. Búsqueda de imágenes según un patrón en un directorio específico	99
Figura 67. Inferencias en un modelo utilizando una API.....	99
Figura 68. ID de detección y la ruta de la imagen procesada.....	99
Figura 69. Dependencias en el archivo pubspec.yaml	100
Figura 70. Función main	101
Figura 71. Script registro usuario	102
Figura 72. Script inicio de sesión.....	103
Figura 73. Script reestablecer contraseña.....	104
Figura 74. Script Homepage	105
Figura 75. Script hometab.....	105
Figura 76. Script informacióntab	106
Figura 77. Script cámara	107
Figura 78. Script detección enfermedades.....	108
Figura 79. Script control enfermedades.....	109
Figura 80. Script clima arveja	110
Figura 81. Script perfil de usuario	111
Figura 82. Prueba Robo, Quantum Arveja	112
Figura 83. Rendimiento de la aplicación en Test lab.....	112
Figura 84. Rendimiento CPU, Memoria	113
Figura 85. Modelo de detección.....	118

Figura 86. Verificación de enfermedades en el cultivo de arveja	139
Figura 87. Demostración del software a los agricultores.....	140
Figura 88. Detección de las enfermedades en tiempo real	140
Figura 89. Icono de la aplicación	148
Figura 90. Inicio de sesión	149
Figura 91. Restablecer contraseña	149
Figura 92. Registrarse con Google	150
Figura 93. Elegir cuenta de correo electrónico.....	150
Figura 94. Menú inicio	151
Figura 95. Manejo Agronómico	151
Figura 96. Taxonomía del cultivo	152
Figura 97. Requerimientos nutricionales	152
Figura 98. Requerimientos edafoclimáticos.....	153
Figura 99. Plagas.....	153
Figura 100. Enfermedades.....	154
Figura 101. Cosecha	154
Figura 102. Información del cultivo	155
Figura 103. Detección enfermedades.....	155
Figura 104. Icono tomar foto desde cámara	156
Figura 105. Icono seleccionar desde galería	156
Figura 106. Detección de enfermedades.....	156
Figura 107. Icono control	157
Figura 108. Dosificación preventiva y curativa	157
Figura 109. Zona meteorológica Tulcán.....	158
Figura 110. Zona meteorológica Huaca	158
Figura 111. Zona meteorológica Carmelo	159
Figura 112. Zona meteorológica Mira.....	159
Figura 113. Perfil	160
Figura 114. Manual de usuario.....	160
Figura 115. Programadores	161

ÍNDICE DE ANEXOS

Anexo 1. Acta de la sustentación de Predefensa del TIC	130
Anexo 2. Certificado del abstract por parte de idiomas.....	132
Anexo 3. Informe de Turniting.....	134
Anexo 4. Encuesta dirigida a agricultores.....	135
Anexo 5. Entrevista dirigida a agrónomos.....	138
Anexo 6. Registro fotográfico.....	139

RESUMEN

La presente investigación titulada "Visión artificial en la agricultura de precisión para la identificación de enfermedades en cultivos de arveja Quantum" se centra en los sembríos de arveja Quantum para la detección de enfermedades mediante el uso de herramientas tecnológicas. El objetivo principal consistió en desarrollar una aplicación móvil haciendo uso de visión artificial enfocada a la agricultura de precisión, específicamente dirigida a los sembríos mencionados. El presente proyecto se llevó a cabo en la provincia del Carchi, con énfasis en el cantón Montúfar, ciudad de San Gabriel, comunidad La Delicia, donde se identificó un problema recurrente en la detección temprana de enfermedades como: antracnosis, fusarium y oídio. La metodología empleada combinó enfoques cualitativos y cuantitativos. El análisis cualitativo se centró en las características particulares de las enfermedades, por otro lado, el enfoque cuantitativo se basó en la información recopilada mediante instrumentos aplicados a 192 agricultores. La investigación abarcó métodos: descriptivos, explicativos, exploratorios, de campo y documentales, proporcionando información valiosa para el estudio. En relación con la detección de enfermedades, se observó que los agricultores la realizaban de manera empírica, tomando como base su experiencia e identificando las enfermedades en etapas avanzadas del crecimiento de la planta. Se propuso entonces, el desarrollo de un aplicativo haciendo uso de algoritmos de visión artificial con el fin de mejorar el tiempo en la detección de diferentes enfermedades en el cultivo. Se optó por la metodología ágil XP (Extreme Programming), por su adaptabilidad en el desarrollo de software mejorando su calidad y capacidad de ajustarse a los cambios en los requisitos del cliente. Esta metodología por su versatilidad permitió una comunicación efectiva entre el cliente y el desarrollador, ajustándose adecuadamente a los objetivos de la investigación.

Palabras claves: Visión Artificial, Agricultura de precisión, metodología XP.

ABSTRACT

This research named "Computer Vision in Precision Agriculture for the identification of diseases in Quantum pea crops" focuses on Quantum pea crops for disease detection using technological tools. The main objective was to develop a mobile application using artificial vision that focused on precision agriculture, specifically aimed at the crops. This project was carried out in the province of Carchi; in the Montúfar canton; city, San Gabriel; community, La Delicia, where we identified a recurring problem in the early detection of diseases such as: anthracnose, fusarium, and powdery mildew. The methodology used combined qualitative and quantitative approaches. The qualitative analysis focused on the characteristics of the diseases, on the other hand, the quantitative approach was based on information collected through instruments applied to 192 farmers. The research encompassed descriptive, explanatory, exploratory, field, and documentary methods, providing valuable information for the study. In relation to disease detection, it was observed that farmers did it empirically, based on their experience and identifying diseases in advanced stages of the plant growth. We proposed to develop an application using computer vision algorithms to improve the time to detect different diseases in the crop. The agile XP (Extreme Programming) methodology was chosen for its adaptability in software development, improving its quality and ability to adjust to changes in customer requirements. This methodology, due to its versatility, allowed effective communication between the client and the developer, fitting adequately to the objectives of this research.

Keywords: Computer Vision, Precision Agriculture, XP methodology.

INTRODUCCIÓN

El impulso de la investigación nace de la selección de un tema de investigación que se centra en abordar los desafíos presentes en la agricultura, específicamente en la tarea de reconocer enfermedades durante las etapas tempranas del cultivo de arvejas específicamente en el cantón Montufar ciudad de San Gabriel comunidad la Delicia, cabe señalar que los agricultores llevan a cabo la detección de enfermedades de manera empírica, lo que implica que utilizan métodos tradicionales y basados en su experiencia práctica para este propósito. Por otro lado, debido a que cada enfermedad en el cultivo exhibe un conjunto único de características, es importante tener en cuenta la variedad de enfermedades que están presentes, por ende, es necesario la utilización de algoritmos de visión artificial para lograr la detección de enfermedades específicas como lo es antracnosis, fusarium y oídio.

Durante este estudio, se aplicó el enfoque mixto para tener una visión global de la problemática, es decir obtener información completa y precisa de las anomalías que afectan al sembrío de arveja quantum, por la parte cualitativa se obtuvo información detallada de las enfermedades (antracnosis, fusarium y oídio), esto permitió saber las características específicas, las cuales son importantes para lograr una buena eficiencia del aplicativo móvil, por otro lado, lo cuantitativo se lo recopiló de los instrumentos aplicados a los agricultores de la comunidad, dicha información ayudo a mejorar en diferentes aspectos la propuesta de la investigación además, mencionaron aspectos de vital importancia como el clima, fungicidas y circunstancias específicas que se debe de tomar en cuenta para el diseño, comportamiento y satisfacción que debe de brindar el aplicativo. Además, se empleó la metodología XP que brinda un contexto de entendimiento entre usuario y desarrollador, gracias a esto mejora la calidad del software, es decir que la metodología XP se centra en la colaboración, la adaptabilidad y la mejora continua para atender las necesidades en constante cambio del cliente y entregar un software de calidad y eficiente.

I. EL PROBLEMA

1.1. PLANTEAMIENTO DEL PROBLEMA

En el mundo el cultivo de arveja se enfrenta a un gran desafío, puesto que la oferta de este producto es muy alta, de tal manera Agrositio (2020), menciona que la demanda de este tipo de legumbre es alta en naciones como China, India y Pakistán. Canadá se destaca como el principal exportador y productor, en segundo lugar, la región del Mar Negro, la Unión Europea, Asia y Australia en términos de importancia. A pesar de ello, Argentina desempeña un papel significativo en el mercado, especialmente en la fabricación de guisantes verdes. No obstante, surge un problema con la variedad de guisante amarillo, tomando en cuenta que esta no goza de suficiente aceptación en el mercado.

Según Roldán et al. (2019), afirma que el problema que experimenta este producto son las diferentes enfermedades que adquieren, ya sea por los cambios de clima o una mala elección de semilla, es decir, que la semilla está infectada por hongos, por tal motivo, es importante realizar investigaciones específicamente en la agricultura de precisión por lo que utiliza tecnologías como la visión artificial con el fin de potenciar la eficacia y rendimiento de los cultivos.

Es por eso por lo que algunos beneficios del uso de la visión artificial enfocada en la agricultura de precisión incluyen: la detección automática de enfermedades y plagas, análisis de la salud de los cultivos, identificación de plantas individuales, optimización de riego, análisis de suelo y fertilización que ayuden a la detección de estas enfermedades como: fusarium, antracnosis, oídio, entre otras.

En Pamplona, según Torrado (2019), menciona que el cultivo de arveja es de baja calidad, puesto que los agricultores no hacen el uso de asistencia técnica para el proceso de cultivo de esta leguminosa, sino más bien se enfocan en cultivar este producto de manera tradicional con la utilización de químicos no muy bien recomendados por una persona experta en el tema, además en estas etapas del cultivo se destacan hongos pertenecientes al género *Ascochyta* (*Ascochyta pisi*, *Ascochyta pinodes* y *Ascochyta pinodella*).

Por ende, se presenta la enfermedad fusarium que es visualizada a la larga distancia con manchones amarillos, marchitamiento o su vez también el deterioro en las hojas de la planta causando pérdidas del cultivo.

Según Villota (2019), afirma que en el Ecuador la agricultura representa el 7% de todo el producto interno del año 2017 la cual se la denomina la actividad que produce más empleos en todo el país, pero debido a varios avances tecnológicos, toda esta producción de campos en el Ecuador se la considera baja, las causas que generan esta baja producción son varias como el limitado acceso a fertilizantes o el bajo despliegue de sistemas de riego, sin embargo, estos inconvenientes tienen muy poca probabilidad lo que hace que el Ecuador tenga reducida la aplicación tecnológica que permiten generar la optimización y mejorar los procesos agrícolas.

Según Angulo (2019), afirma que, en la provincia del Carchi, especialmente en el Cantón Bolívar, los agricultores tienen un conocimiento limitado sobre los diversos hongos que afectan el cultivo de arvejas. Esta falta de conocimiento ha generado problemas en las plantas, aunque se han implementado medidas de control en los últimos años, estas no han sido eficaces para proteger completamente las plantas. Como resultado, se han producido daños irreparables en los cultivos.

Una de las mejores variedades de arveja es la Quantum, este es uno de los granos más resistentes además de poseer mejor calidad, según Moreno (2019), manifiesta que en la actualidad los sembríos de arveja variedad Quantum en la comunidad de Monte Verde, provincia del Carchi, los agricultores eligen utilizar semillas certificadas para aumentar la eficiencia de sus cosechas. Además, también prefieren emplear semillas que hayan sido cosechadas previamente y hayan demostrado ser efectivas. Destacan que, aparte de estas variedades, también siembran otras como Sindamanoy y San Isidro, reconocidas por su alta calidad.

La presente investigación se desarrollará en la Provincia del Carchi, Cantón Montúfar, específicamente en la ciudad de San Gabriel, en la comunidad de la Delicia, puesto que, en esta comunidad no existe un control fitosanitario, es decir que carece de medidas destinadas a proteger y prevenir la salud de las plantas como también de controlar o erradicar enfermedades, al mismo tiempo se evidencia agentes patógenos que por el desconocimiento de fungicidas y el control actual se lo realiza mediante prácticas tradicionales, los cultivos de arveja Quantum

se encuentran con un alto riesgo de propagación de enfermedades principalmente de antracnosis, fusarium y oídio.

1.2. FORMULACIÓN DEL PROBLEMA

El desconocimiento de la visión artificial en cultivos de arveja Quantum provoca la propagación no controlada de agentes patógenos generando una identificación de enfermedades tardía mediante métodos tradicionales, en la comunidad de La Delicia, ciudad de San Gabriel, durante el periodo 2023.

1.3. JUSTIFICACIÓN

El propósito de este proyecto es proporcionar a los agricultores una herramienta tecnológica, haciendo el uso de la visión artificial para detectar de forma temprana de las enfermedades denominadas antracnosis, oídio y fusarium debido a que son las principales que afectan a las plantaciones de arveja, durante el proceso de siembra, puesto que dichas enfermedades no se han podido identificar anticipadamente, generan pérdidas en el cultivo y una baja calidad del producto, al igual que su producción.

Según Garzón (2020), manifiesta que el uso de la visión artificial en la agricultura se enfoca en el mejoramiento y la eficiencia de los productos de siembra y a su vez aumenta la producción más de un 50% en la calidad del producto. Por lo que los pequeños y grandes agricultores pueden tener una mejor manipulación de los campos, ya sean estos grandes o pequeños en el proceso de detectar las anomalías de los productos que se encuentran en progreso.

Los resultados que arroja el estudio, mediante la evaluación de algoritmos de detección de objetos ayudarán a los agricultores a identificar de manera eficiente las enfermedades que provocan daños en las diferentes partes de la planta, es decir; tallo, hoja y fruto, además se puede establecer un mejor rendimiento en la producción, beneficiando así a los consumidores, aumentando su calidad y a los agricultores económicamente, por lo que esta herramienta tecnológica también brinda un control total del cultivo. De esta manera, la investigación adquiere la importancia en este sector debido a que al hacer la utilización de visión artificial en la identificación de enfermedades en los cultivos de arveja mejorará la estabilidad del agricultor y la satisfacción del consumidor en la comunidad de la Delicia.

1.4. OBJETIVOS Y PREGUNTAS DE INVESTIGACIÓN

1.4.1. Objetivo General

Desarrollar una solución informática haciendo uso de la visión artificial en la agricultura de precisión para la identificación temprana de enfermedades en los cultivos de arveja Quantum.

1.4.2. Objetivos Específicos

- Fundamentar bibliográficamente las variables de estudio para conocer una solución al problema.
- Seleccionar un algoritmo de visión artificial que permita la identificación de enfermedades en plantas.
- Identificar las principales enfermedades que afectan al cultivo de arveja a edad temprana.
- Proponer un aplicativo con base en la visión artificial haciendo el uso de bounding boxes para la identificación de enfermedades fúngicas en arveja variedad Quantum.

1.4.3. Preguntas de Investigación

- ¿Las variables de estudio proporcionan una posible solución al problema de cultivo de arveja?
- ¿EL algoritmo de visión artificial permite la identificación de enfermedades en las plantas?
- ¿Cuáles son las principales enfermedades que afectan al cultivo de arveja?
- ¿Cómo es el aplicativo con visión artificial que ayude la identificación de enfermedades fúngicas en arveja?

II. FUNDAMENTACIÓN TEÓRICA

2.1. ANTECEDENTES DE LA INVESTIGACIÓN

Como respaldo del avance de este proyecto, se ha empleado como base teórica el análisis de investigaciones previas enfocadas con el empleo de la visión artificial en la agricultura de precisión para la identificación de enfermedades en sembríos.

Según Huanca (2022), en su proyecto de investigación propuso la elaboración de una app móvil para el diagnóstico preliminar del oídio mediante imágenes digitales de la planta de uva. Además, empleó una metodología de investigación cuantitativa y experimental. Tras el desarrollo de la aplicación, se logró una precisión del 99,00% en el indicador de sensibilidad, mientras que la especificidad osciló entre el 92,38% y el 96,00%. Esto lleva a concluir que la aplicación móvil contribuye al diagnóstico preliminar del oídio mediante imágenes digitales.

Según Serrano y Torres (2020), en su investigación, plantea la creación de un prototipo de aplicación móvil destinada a identificar mazorcas de cacao afectadas por *Monilia* y/o *Fitóftora*, haciendo uso de tecnologías de visión por computadora y aprendizaje automático. Después de recopilar datos, compila un conjunto de información que abarca las enfermedades más preocupantes, como la *Fitóftora* y la *Monilia*. Los resultados obtenidos mediante el uso del modelo YOLOv4 arrojan una precisión del 60% en la detección de las mazorcas de cacao. Estos hallazgos insinúan que dicha herramienta móvil podría resultar beneficiosa tanto para los agricultores como para los investigadores agrícolas en la toma de decisiones.

El estudio realizado por, Cusme y Loor (2019), tuvo como propósito crear una app móvil mediante el uso de aprendizaje automático para identificar y clasificar "la roya" en los cultivos de café robusta, buscando automatizar este proceso. Se recolectaron imágenes de hojas y plantas afectadas para entrenar los algoritmos seleccionados. Tras el desarrollo e implementación de la aplicación, se consiguió una efectividad del 95% en la detección de la roya. Se logró una precisión del 97% en la detección y categorización de la enfermedad en las hojas de café robusta.

Las repeticiones llevadas a cabo mostraron un promedio del 93% de precisión en la detección, lo que destaca la exactitud y utilidad de la aplicación creada.

Además, se toma en cuenta la presencia de consumo y producción en la provincia del Carchi para identificar sus ventajas y desventajas en el cultivo por lo que, Issuu (2019), afirma que en la provincia de Carchi se registra un mayor índice tanto de consumo como de producción de arveja, representando el 47,4% a nivel nacional. Esto indica que una parte significativa de las pérdidas financieras para quienes cultivan se reducen por plagas y enfermedades, como resultado de una mayor atención a las técnicas y el manejo del cultivo. Actualmente, se puede observar que estas atacan con mayor frecuencia a los cultivos son; barrenador, minador, oídio, fusarium y antracnosis con los porcentajes de 67%, 33%, 47%, 33%, 20% respectivamente, lo que reduce la productividad de los cultivos.

La identificación y categorización de granos de café resulta fundamental para mantener los precios internacionales en niveles elevados. Sin embargo, en las distintas prácticas de selección de productos, aún se emplea un enfoque tradicional. En este contexto, Gonzabay (2023), llevó a cabo un proyecto de desarrollo que se basa en el uso de redes neuronales convolucionales (CNN) para el procesamiento de imágenes. Este método permite una selección más eficiente de cada producto. Para la clasificación de los granos de café, se empleó MobileNet, una arquitectura optimizada. El análisis de rendimiento se realizó para las arquitecturas MobileNetV2 y MobileNetV1, arrojando precisiones del 99.18% y 88.72%, respectivamente. Además, se obtuvo una validación del 99.20% y 81.25% respectivamente. De estos resultados, se concluyó que MobileNetV2 es más apropiado para lograr una clasificación óptima.

La tarea de identificar y clasificar las plagas presentes en los cultivos de tomate riñón demanda el uso de algoritmos que tengan la capacidad de procesar información de manera efectiva para su posterior análisis. Según Jácome (2022), en su investigación se resalta la importancia de la utilización de diversas tecnologías para obtener resultados óptimos. Su objetivo primordial fue detectar plagas como el minador y la mosca blanca, así como la enfermedad fusarium en el tomate riñón de manera temprana mediante la aplicación de técnicas de visión artificial, aprendizaje automático y Edge Computing. Estas técnicas permiten optimizar los recursos de producción. Los datos finales muestran una exactitud del 96.875% en la validación del

modelo original, del 98.438 % para el modelo LITE diseñado para la tarjeta de desarrollo, y del 92.54 % en pruebas reales, validadas con la asistencia de expertos.

Según Mórán y Quiranza (2022), en su trabajo de investigación destaca una solución informática que permite la detección de enfermedades de papa que hace el uso de visión artificial. Este enfoque se centra específicamente en el ámbito agrícola, abordando de manera efectiva los desafíos asociados a las patologías que impactan en los cultivos de papa. Para la implementación exitosa de esta aplicación, se ha centrado en un enfoque mixto que adopta técnicas de investigación, respaldado además por la metodología XP (Programación Extrema), que se ha revelado como un enfoque eficaz para el desarrollo ágil de software.

Según Guerrero (2021), se destaca la importancia crucial del seguimiento de cultivos en parcelas utilizando la visión artificial, dado que esta actividad es fundamental para evaluar el rendimiento y anticipar la cosecha. Sin embargo, los productores en el sector enfrentan desafíos significativos debido a factores climáticos que complican la detección temprana de enfermedades y plagas. Durante la evolución del sistema, se utilizó un conjunto de datos compuesto por 502 imágenes y se ejecutaron un total de 6000 iteraciones de entrenamiento. Los resultados obtenidos revelaron que, en un contexto de datos densamente poblado, la Intersección sobre la Unión (IoU) alcanzó el 59.23%, mientras que la Precisión Media (mAP) fue del 81.60%, lo que demuestra que el sistema logra detectar el objeto de estudio con una eficacia del 86%. Además, el modelo propuesto exhibe un rendimiento altamente satisfactorio, con un promedio de precisión del 96%.

2.2. MARCO TEÓRICO

El proyecto se estructura en dos fases fundamentales. En la primera etapa, se explorará el ámbito de la visión artificial enfocada en la agricultura de precisión, abordando algoritmos específicos diseñados para la identificación de objetos en ilustraciones. Para llevar a cabo esta tarea, se hará uso de tecnologías de desarrollo punteras como Python, Flutter, Visual Code, entre otras. En la segunda fase, se enfocará en detectar enfermedades en los cultivos de guisantes. Esta fase comprenderá un análisis detallado de las enfermedades más frecuentes, explorando sus características distintivas. Con este enfoque dual, la investigación busca no solo comprender las herramientas tecnológicas utilizadas en la detección visual, sino

también proporcionar una comprensión profunda de las enfermedades que afectan a los cultivos de arveja.

2.2.1. Enfermedades en plantas

Según Cherlinka (2023), afirma que las enfermedades que afectan a las plantas representan una seria situación que pone en peligro la producción agrícola. Es crucial que los agricultores estén debidamente preparados para hacer frente a estos problemas y puedan implementar estrategias de control efectivas a lo largo del proceso de cultivo. Estas enfermedades pueden manifestarse en diferentes partes de la planta, incluyendo las hojas, el tallo y las raíces, incluso pueden afectar a la planta en su totalidad.

2.2.1.1. Tipos de Enfermedades de las plantas y agentes causales

Dentro del ámbito de las enfermedades que afectan a las plantas, se pueden distinguir dos categorías principales: las enfermedades abióticas, las cuales son conocidas como no infecciosas, y las enfermedades bióticas, que se caracterizan por ser infecciosas.

Las enfermedades no contagiosas tienden a surgir en situaciones ambientales desfavorables, como temperaturas extremas o niveles inadecuados de humedad. Estas afecciones son causadas por factores abióticos, como la presencia de sustancias dañinas en el aire.

Es esencial destacar los principales factores responsables de las enfermedades en las plantas, entre los cuales se encuentran bacterias, hongos, nematodos, virus y plantas parasitarias. Reconocer estos agentes es crucial para implementar medidas preventivas y estrategias de control efectivas, asegurando la eficiencia en la producción agrícola.

2.2.1.2. Enfermedades causadas por bacterias

Las enfermedades ocasionadas por bacterias representan una de las afecciones más comunes en la agricultura, manejo y prevención de patologías son tareas considerablemente complejas. Para que se dé la infección, el agente causal debe penetrar en el tejido del cultivo, usualmente a través de áreas dañadas, como las producidas por herramientas agrícolas, la presencia de insectos como pulgones, o debido a condiciones climáticas desfavorables, como polvo, viento o lluvias intensas. Es relevante destacar que las bacterias pueden permanecer latentes, ya sea en la planta misma o en el suelo, durante prolongados periodos de tiempo hasta que se presenten las condiciones óptimas para su desarrollo completo. La comprensión

detallada de estos procesos es fundamental para implementar estrategias efectivas de prevención y control, contribuyendo así a la salud y productividad del cultivo.

2.2.1.3. Enfermedades causadas por hongos

La presencia de hongos patógenos es uno de los problemas más frecuentes en la agricultura. Según Almeida et al. (2019), anualmente, aproximadamente la tercera parte de los sembríos de alimentos se ven afectados por enfermedades de las plantas causadas por hongos. Esta situación no solo tiene importantes implicaciones humanitarias, sino también considerables consecuencias económicas. Similar a las enfermedades bacterianas, las infecciones fúngicas en las plantas se manifiestan mediante heridas, aberturas naturales y áreas húmedas, así como por medio del viento, se suelen dispersar las esporas de los hongos, facilitando su propagación en los cultivos.

2.2.2. Enfermedades en leguminosas

Según Palomo (2022), indica que la fusariosis es una enfermedad con origen en el suelo, y se manifiesta cuando se presentan condiciones ideales como temperaturas elevadas y alta humedad. Su principal objetivo es atacar los vasos de la planta, esta enfermedad puede surgir en cualquier etapa del proceso del cultivo. además, tiende a desarrollarse de manera rápida, especialmente en ambientes con temperaturas elevadas y una humedad considerable.

El mildiu es otra enfermedad común en las leguminosas, la cual tiene su origen tanto en los restos de la cosecha como en las semillas. Esta afección se inicia desde la base de la planta y se propaga hacia la parte superior, afectando los tallos, las hojas y las vainas. La identificación del mildiu se realiza a través del amarillamiento de las hojas, y otra característica distintiva es la presencia de un micelio gomoso en el envés. Esta enfermedad se desarrolla en condiciones muy húmedas y temperaturas bajas.

2.2.3. Arveja Quantum

La arveja Quantum posee un alto potencial de producción, su porte es mediano, sus vainas están entre 7 y 8 granos, además, esta variedad presenta una notable ventaja en su producción debido a la presencia de nudos que se duplican, triplican e incluso cuadruplican, lo que la distingue claramente de otras variedades en términos de rendimiento. Es conocida por su sabor dulce y su capacidad para producir altos rendimientos.

Se necesita de riegos ligeros y frecuentes especialmente cuando está se encuentra en floración y llenado de vaina, hay que evitar el exceso de humedad y se debe

realizar cambio de surco entre 25 a 35 días después de la siembra. Hacer el aporte después del abonamiento, para optimizar el uso del fertilizante y alejar el tallo del contacto con el agua para evitar problemas de pudrición.

Esta hortaliza posee una gran producción y contiene mucho carbohidrato al igual que proteínas, por otro lado, la arveja Quantum es recomendable, puesto que se adapta al clima frío además de un excelente rendimiento de acuerdo con el manejo del cultivo que brindan los agricultores.

Tabla 1. Ventajas y características de la variedad de arveja Quantum

Ventajas de la arveja Quantum	Características
Muestra una alta capacidad para resistir condiciones climáticas desfavorables. Cada vaina contiene entre 7 y 9 granos. Reducción en el uso de pesticidas.	Es una planta de buen vigor y rustica. Promedio de altura 70cm. La floración ocurre en un período muy concentrado.
Tolera las épocas lluviosas.	En cada nudo, se pueden observar de 2 a 4 flores en floración.
Vainas muy compactas de excelente llenado y duración.	Muestra resistencia a las infecciones causadas a <i>Fusarium</i> sp. y Mildiú.
Plantas muy sanas.	Las vainas son sólidas y bien formadas.

Fuente: Guerrero (2023).

2.2.3.1. Requerimientos edafoclimáticos

La arveja es una leguminosa que se cultiva de manera óptima en invierno, muestra resistencia a las heladas durante su fase de desarrollo, mientras crece, pero no durante la etapa de floración (Cordova, 2022).

1. Condiciones edáficas

- **Textura de suelo:** Las características óptimas del suelo incluyen texturas de nivel medio, que abarcan desde franco limoso hasta franco-arcilloso arenoso, con una profundidad efectiva que se sitúa en el rango de 45 a 60 centímetros.
- **pH del suelo:** El rango óptimo se sitúa entre 5.5 y 6.5, indicando un nivel ligeramente ácido a neutro.
- **Drenaje:** Debe permitir que el agua se filtre, y se construye zanjas de 15 a 20 metros, para sacar el agua.
- **Materia orgánica:** Un nivel de materia orgánica en el suelo que oscile entre el 2% y el 5% se considera óptimo.
- **Fertilización:** Implica agregar nutrientes al suelo, aplicar 200 kg. ha⁻¹ de 18-46-00 10-30-10

2. Condiciones climáticas

- Temperatura: Debe estar dentro de un rango específico entre 15°C a 18°C y la mínima en 10°C.
- Luz: Esencial para la fotosíntesis de las plantas en requiriéndose de 5-9 horas/sol/día
- Agua: La arveja necesita 250 a 380 milímetros de agua
- Estación de cultivo: Estación adecuada maximiza la producción entre los 2.000 a 3.000 msnm.
- Viento: La velocidad del viento puede afectar la estructura de las plantas.
- Heladas: Pueden dañar plantas sensibles al frío, sometidos a temperaturas por debajo de +12,5 °C.

2.2.3.2. Enfermedades en el cultivo de arveja variedad Quantum

Debido a la sensibilidad y susceptibilidad del cultivo de arveja a enfermedades durante las etapas de cultivo, uno de los inconvenientes de hoy en día es lograr una producción de alta calidad. Cabe señalar que unas de las enfermedades más frecuentes son; antracnosis, fusarium y oídio, estas son enfermedades causadas por hongos (Castellanos et al., 2021).

Según Sáenz (2020), menciona que en los primeros meses en esta variedad de guisante las enfermedades ya son notorias. Cabe señalar que *Ascochyta Pisi* y *Fusarium* son hongos notables en este tiempo por lo que dañan tanto las hojas como las raíces de la planta, debido a que atacan las zonas más vulnerables de la planta es muy complicado que pueda sobrevivir. Además, si la enfermedad no se controla adecuadamente y en el tiempo exacto puede propagarse con el viento así contagiando a todo el cultivo.

2.2.3.2.1. Antracnosis

Según Guerrero (2023), menciona que esta enfermedad es causada por *Ascochyta pisi* o *Colletotrichum spp*, afectando directamente a los tallos, las hojas y los frutos. Inicialmente, se presenta como áreas redondas de color amarillo, con un diámetro de aproximadamente 5 mm y bordes oscuros. Estas áreas son numerosas y pueden expandirse a lo largo de grandes secciones del órgano afectado. En las vainas, estas manchas pueden aumentar en profundidad y ocasionar daños en las semillas. En climas con alta humedad, hay mayor riesgo de propagación de esta enfermedad, lo que podría llevar al deterioro total de la planta.

La antracnosis o más conocida como ojo amarillo presenta una serie de signos distintivos que impactan diversas partes de la planta. Estos signos comprenden manchas en las hojas, tallos y vainas, las cuales pueden empezar como lesiones pequeñas y redondeadas, extendiéndose con el tiempo y rodeadas por una zona amarillenta. Además, la enfermedad puede causar la caída prematura de las hojas afectadas, provocando una defoliación temprana. Las manchas en las vainas pueden generar ablandamiento y deformación de los tejidos, y en casos más graves, pueden resultar en la putrefacción y disminución del rendimiento de la cosecha. Asimismo, los frutos también pueden ser afectados en etapas avanzadas de la enfermedad, mostrando síntomas de pudrición y descomposición. La intensidad de estos síntomas puede variar según factores como la variedad de arveja, las condiciones ambientales y la susceptibilidad de la planta a la enfermedad.

2.2.3.2.2. Fusarium (oxysporium)

Según Castillo (2019), menciona que fusarium es una enfermedad que es ocasionada por un hongo denominado (fusarium oxysporum f.sp. pisi) el cual afecta a los cultivos, provocando daños en el tallo y su follaje, por lo cual este hongo se muestra en la planta con un color amarillento debido a que ataca en un inicio en lo que se le denomina cuello de la raíz hasta que llega a cubrir toda la planta, los agricultores la conocen esta enfermedad por su nombre tradicional denominado marchitez por fusarium.

De acuerdo con la información proporcionada por, Guerrero(2023), menciona que esta patología es provocada por el hongo Fusarium spp y se manifiesta durante la fase en que la planta está floreciendo. Durante esta etapa, los síntomas se manifiestan inicialmente como un amarillamiento de las hojas que avanza gradualmente con el tiempo, y conforme avanza se produce un deterioro adicional, culminando en el marchitamiento completo de la planta.

Según Waldimar (2022), la presencia de Fusarium se manifiesta a través de un amarillamiento blanquecino. Este fenómeno resulta en una pudrición seca de los componentes de las raíces internas y la región inferior del tallo, exhibiendo un tono rojizo. Este cambio de color se atribuye a las toxinas generadas por el hongo, que provocan la obstrucción de la xilema, ocasionando en última instancia, el fallecimiento de la planta. Además, resalta las circunstancias apropiadas para la

propagación de esta enfermedad se encuentran en temperaturas entre 20 y 25 °C, acompañadas de una humedad que oscile entre el 75% y el 95%.

2.2.3.2.3. Oídio (*Erysiphe polygoni*)

Según Angulo (2019), el Oídio también conocida como *erysiphe polygoni*, es una afección fúngica que afecta a las plantas de arveja (*Pisum sativum*). Es causada por diversos hongos pertenecientes a los géneros *Erysiphe* o *Sphaerotheca*. Se manifiesta como una capa de polvo blanco que aparece en los tallos, hojas, flores y vainas.

Esta enfermedad suele comenzar con la aparición de pequeñas manchas blancas en las hojas, que con el tiempo se agrandan y se fusionan, dando lugar a áreas más amplias cubiertas por el hongo. Si no se controla adecuadamente, puede debilitar la planta, reduciendo su capacidad para realizar la fotosíntesis y producir alimentos, lo que puede llevar a una disminución en el rendimiento de la cosecha.

La propagación del oídio ocurre mediante esporas que son transportadas por el viento o a través del contacto directo entre plantas. Condiciones que involucran una alta humedad relativa y temperaturas moderadas suelen ser propicias para su desarrollo. Para controlar esta enfermedad, se pueden implementar prácticas culturales, como la eliminación de plantas infectadas, así como el uso de fungicidas específicos para combatir el hongo.

Tabla 2. Matriz de correlación entre enfermedad y causas

Hoja	Tallo	Fruto	Enfermedades
X	X	X	
	X		
X	X	X	

Antracnosis
Fusarium
Oídio

Tabla 3. Plagas y condiciones climáticas

Plagas				Condiciones climáticas					
Barrenador del tallo (Melanagromyza lini)	Minador (Lyriomyza trifolii)	Trips (Frankliniella spp.)	Pulgón verde (Myzus persicae)	Enfermedades	Alta Humedad (70%)	Temperaturas Moderadas (10°C - 25°C)	Suelos Mal Drenados	Viento (20-30 km/h)	Deficiente fertilización
X	X	X	X	Antracnosis	X	X		X	X
				Fusarium	X	X	X		X
	X	X	X	Oídio		X		X	X

Tabla 4. Enfermedades y efectos

Enfermedades	Agente Causal		Síntomas en Hojas	Síntomas en Tallo	Síntomas en Vainas	Síntomas en Semillas	Efectos en el Rendimiento	Condiciones Favorables
	Antracnosis	<i>Colletotrichum</i> spp.	Manchas oscuras, hundidas, fusión de manchas.	Lesiones alargadas, oscuras y deprimidas.	Manchas oscuras y hundidas.	Manchas negras o marrones.	Reducción de rendimiento, calidad de vainas.	Humedad alta, temperaturas moderadas a cálidas.
Fusarium	<i>Fusarium oxysporum</i>	Amarillamiento, marchitamiento	Cancros en la base del tallo, marchitez	En casos severos, manchas marrones	Podredumbre, reducción de viabilidad	Muerte de plantas, reducción significativa de rendimiento.	Suelos mal drenados, temperaturas cálidas	
Oídio	<i>Erysiphe pisi</i>	Polvo blanco en hojas, hojas amarillas	Lesiones blancas, polvo blanco	Manchas blancas, vellosas en vainas	No es común	Reducción de crecimiento, vigor de plantas	Humedad baja a moderada, temperaturas cálidas	

2.2.4. Visión artificial en la agricultura

La visión artificial en la agricultura es una tecnología cada vez más utilizada para mejorar los resultados del sector agrícola. Esta tecnología utiliza técnicas avanzadas de visión artificial, como el reconocimiento facial, para detectar y clasificar plantas, animales y objetos en los cultivos, según Cuauhtemoc (2019), esto permite a los agricultores identificar y diagnosticar los problemas de sus cultivos de manera rápida y precisa. La visión artificial también se ha utilizado para monitorear el progreso y la salud de las plantaciones a través de la detección de cambios en la luz, el color y la madurez de las frutas y verduras. Además, algunos productores agrícolas han empezado a utilizar la visión artificial para gestionar y optimizar los recursos y los procesos relacionados con la agricultura.

2.2.5. Algoritmos de aprendizaje para detección de objetos en imágenes

Según Anitha y Jayalakshmi (2021), mencionan que los algoritmos de aprendizaje para identificación de imágenes son programas informáticos diseñados para reconocer patrones y características en imágenes, conocidos como algoritmos de aprendizaje para identificación de imágenes, emplean técnicas de aprendizaje automático, especialmente aprendizaje supervisado. En este proceso, se realizan entrenamientos utilizando conjuntos de datos que incluyen imágenes etiquetadas, es decir, imágenes acompañadas de sus respectivas categorías o clases.

Durante el entrenamiento, el algoritmo se alimenta con miles o incluso millones de imágenes junto con las etiquetas correctas, permitiéndole aprender a asociar patrones visuales con las clases a las que pertenecen. En este contexto, la red neuronal convolucional (CNN) emerge como el enfoque de aprendizaje más predominante, destacando por su eficacia en actividades relacionadas con la visión artificial en computadoras, como la identificación de imágenes.

Una vez que el algoritmo ha sido entrenado, se vuelve capaz de clasificar nuevas imágenes asignándoles la etiqueta o categoría aprendida durante el entrenamiento, en un proceso conocido como inferencia. Estos algoritmos de identificación de imágenes encuentran aplicación en diversos campos, como el reconocimiento facial, diagnóstico médico, vehículos autónomos, sistemas de seguridad, entre otros.

Existen diferentes algoritmos de aprendizaje tal y como se muestra en la tabla 5, en la cual se realiza una comparativa entre algoritmos, en donde figura características

esenciales para la identificación del algoritmo más calificado para la presente investigación.

Tabla 5. Comparativa de algoritmos de aprendizaje para detectar de objetos en imágenes

Algoritmo de aprendizaje	Características	Ventajas	Desventajas	Aplicaciones	Escalabilidad	Requisitos computacionales	Análisis
R-CNN	Propone regiones de interés. Procesamiento en etapas.	Alta precisión en la detección. Capacidad de detectar objetos variados.	Lento en la inferencia. Requiere mucho almacenamiento y poder computacional.	Detección de objetos en imágenes.	Limitada por su velocidad en el tiempo de ejecución	Requiere altos requisitos computacionales.	Es reconocido por su alta precisión en la detección de una variedad de objetos, aunque su lentitud en la inferencia y alta demanda de recursos computacionales limitan su utilidad en aplicaciones que requieren tiempos de ejecución rápidos.
Fast R-CNN	Utiliza una red convolucional para extracción de características. Propone regiones de interés mediante una sola pasada	Eficiente en la extracción de características y detección. No necesita regiones de interés previamente definidas.	Requiere un preprocesamiento adicional. Puede perder detalles finos en objetos pequeño	Detección de objetos en imágenes	Mejora en velocidad en comparación con R-CNN	Necesita recursos computacionales considerables.	Mejora la eficiencia de R-CNN al realizar la extracción de características y propuesta de regiones de interés en una sola pasada, pero aún requiere recursos computacionales considerables y puede perder detalles en objetos pequeños.
Faster R-CNN	Propone regiones de interés mediante redes convolucionales y de detección. Buena precisión en la detección.	Eficiente en la generación de regiones de interés. Buena precisión en la detección.	Complejo y requiere ajustes de hiperparámetros. Requiere más ajustes de hiperparámetros.	Detección de objetos en imágenes	Más escalable que R-CNN y Fast R-CNN.	Es necesario tener recursos computacionales significativo	Integra redes convolucionales y de detección para perfeccionar la eficiencia en la producción de regiones de interés, siendo más escalable que R-CNN y

Algoritmo de aprendizaje	Características	Ventajas	Desventajas	Aplicaciones	Escalabilidad	Requisitos computacionales	Análisis
Yolo	Realiza la detección de objetos en una sola pasada. Utiliza una única red convolucional para detección	Alta velocidad de detección. No requiere regiones de interés previamente definidas.	Puede tener inconvenientes en detectar objetos pequeños.	Detección de objetos en tiempo real.	Buena escalabilidad y velocidad.	Solicita menos recursos computacionales.	Fast R-CNN, pero su implementación es más compleja y requiere ajustes adicionales de hiperparámetros. Se destaca por su alta velocidad de detección y capacidad para realizar la detección en tiempo real utilizando una única red convolucional, aunque puede tener dificultades con objetos pequeños debido a su enfoque de una sola pasada.

Como se muestra en la tabla 5, se realizó la comparativa entre algoritmos de aprendizaje en donde se llegó a la conclusión de que el algoritmo Yolo es el más apropiado para la detección de objetos, puesto que realiza la identificación en tiempo real a diferencia de los otros, cabe señalar que este algoritmo requiere menos recursos computacionales lo que es conveniente para la investigación. Yolo es eficaz de modo que se puede adaptar a diferentes tamaños de objetos y a su vez a diferentes escenarios cabe mencionar que tiene una alta velocidad en el proceso y detección es por ello por lo que, esta elección es la más idónea para el presente proyecto.

En la tabla 6, se realiza una comparativa de algoritmos de detección de objetos en dos diferentes SO.

Tabla 6. Comparación de algoritmos

Modelo	Ítems de evaluación	Windows 10	Windows 11	Comparación Específica de Modelos
YOLOV8	Número de Imágenes	1320	1320	<ul style="list-style-type: none"> - En ambos sistemas operativos, tiene un porcentaje de validación bastante alto, aunque ligeramente mejor en Windows 10. - El margen de error es menor en Windows 11. - El tiempo de ejecución es comparable en ambos sistemas operativos.
	Número de épocas	100	100	
	Tiempo	15.50H	18.08H	
	Porcentaje de validación	0.916%	0.873%	
	Margen de error	0.09%	0.13%	
YOLOV5	Número de Imágenes	1320	1320	<ul style="list-style-type: none"> - El porcentaje de validación es más alto en Windows 11. - El margen de error es menor en Windows 11. - El tiempo de ejecución es comparable en ambos sistemas operativos.
	Número de épocas	100	100	
	Tiempo	20.66H	22.16H	
	Porcentaje de validación	0.83%	0.75%	
	Margen de error	0.17%	0.24%	
R-CNN	Número de Imágenes	1320	1320	<ul style="list-style-type: none"> - El porcentaje de validación es más alto en Windows 11. - El margen de error es menor en Windows 11. - El tiempo de ejecución es mayor en Windows 11 en comparación con Windows 10.
	Número de épocas	100	100	
	Tiempo	28.12H	34.75H	
	Porcentaje de validación	0.66%	0.74%	
	Margen de error	0.34%	0.26%	
Fast R-CNN	Número de Imágenes	1320	1320	<ul style="list-style-type: none"> - El porcentaje de validación es más alto en Windows 11. - El margen de error es menor en Windows 11. - El tiempo de ejecución es mayor en Windows 11 en comparación con Windows 10.
	Número de épocas	100	100	
	Tiempo	22.47H	31.57H	
	Porcentaje de validación	0.68%	0.78%	
	Margen de error	0.32%	0.21%	

La ejecución del modelo YOLOv8 se llevó a cabo en dos sistemas operativos distintos, tanto en la versión 10 como en la versión 11, como se muestra en la tabla 6. Tras realizar un análisis, se llegó a la conclusión de que YOLOv8 es considerablemente más eficiente en cuanto al rendimiento de tiempo en comparación con las otras opciones. Es importante destacar que, a pesar de las diferencias en el tiempo de ejecución, los resultados en términos de precisión fueron similares para todas las configuraciones evaluadas.

2.2.5.1. MobileNetV2

Según González y Pacheco (2022), MobileNetV2 ha sido desarrollado como una arquitectura de redes neuronales convolucionales específicamente diseñada para su implementación en dispositivos móviles y sistemas con recursos limitados, centrándose en aplicaciones de visión por computadora. Esta versión representa una evolución de la arquitectura inicial de MobileNet, introduciendo mejoras sustanciales tanto en eficiencia como en precisión. Utiliza técnicas de optimización como la separación de convoluciones y el uso de bloques residuales para lograr un equilibrio entre precisión y eficiencia. Además, permite ajustar parámetros como el "ancho" y la "resolución" para adaptarse a diferentes requisitos de aplicaciones y recursos de hardware. MobileNetV2 es muy útil en tareas como categorización de imágenes, la identificación de objetos y el análisis semántico de segmentos, siendo especialmente adecuado para implementaciones en dispositivos móviles y sistemas embebidos debido a su eficiencia computacional.

De acuerdo con González y Pacheco (2022), La estructura de MobileNetV2 se ha diseñado especialmente para dispositivos móviles que tienen capacidades computacionales limitadas. Su eficiencia sobresaliente permite la ejecución de tareas de visión artificial en tiempo real. MobileNetV2 emplea convoluciones profundas separables y bloques residuales invertidos para reducir la cantidad de parámetros y operaciones requeridas, manteniendo la precisión. Además, incorpora técnicas como la ampliación de canales y la activación de ReLU6 para mejorar aún más la eficiencia y la capacidad de representación. Esta arquitectura ha probado ser valiosa en una variedad de aplicaciones de visión artificial en dispositivos móviles, simplificando la implementación de modelos de aprendizaje profundo en entornos con recursos limitados.

Tabla 7. Análisis del modelo MobileNetV2

Modelo	Ítems de evaluación	Windows 10	Windows 11	Análisis de rendimiento
MobileNetV2	Número de imágenes	de 1320	1320	- Aunque el tiempo de entrenamiento es significativamente mayor en Windows 11, no hay una mejora notable en la precisión del modelo en comparación con Windows 10. - La leve reducción en la pérdida de validación en Windows 11 no parece justificar el aumento significativo en el tiempo de entrenamiento.
	Número de épocas	de 100	100	
	Tiempo	20.17h	31.23h	
	Precisión entrenamiento	1.0000	1.0000	
	Pérdida validación	0.1773	0.1646	
	Precisión validación	0.9451	0.9451	

En la tabla 7, se examinó minuciosamente el rendimiento de un modelo de clasificación denominado MobileNetV2 en dos sistemas operativos diferentes, evidenciando que Windows 10 es el más apto, además, su tiempo es el más conveniente para su ejecución.

2.2.6. Tecnología en la nube

2.2.6.1. Google Colab

Según Das (2023), menciona que Google Colab ofrece acceso gratuito a una amplia gama de recursos informáticos, que incluyen espacio de almacenamiento, memoria RAM, capacidad de procesamiento, tarjetas gráficas (GPU) y unidades de procesamiento tensorial (TPU). Además de esto, Colab simplifica el proceso de reutilización de archivos de Jupyter Notebook alojados en GitHub y permite la importación de proyectos relacionados con ciencia de datos y aprendizaje automático desde diversas fuentes. Esta plataforma también facilita la colaboración eficiente al permitir la importación de activos, lo que resulta en la presentación de código Python de manera clara y libre de errores. Los usuarios de Free Colab tienen la oportunidad de acceder de manera gratuita a tiempos de ejecución de GPU y TPU por un máximo de 12 horas. Este servicio brinda una variedad de recursos tecnológicos valiosos, que van desde una CPU Intel Xeon hasta un poderoso acelerador Tesla K80. Asimismo, el tiempo de ejecución de TPU ofrece una impresionante capacidad de potencia computacional gracias a una TPU en la nube con una potencia de 180 teraflops. Para aquellos que requieran una mayor capacidad de procesamiento, Colab Pro y Pro+ ofrecen la opción de ampliar estos recursos, permitiendo a los usuarios prolongar su tiempo de uso más allá de las 12 horas iniciales.

Gracias a todos los beneficios antes mencionados esta herramienta es la opción ideal para llevar a cabo esta investigación, permitiendo ejecutar el algoritmo en un mínimo tiempo por su gran capacidad de forma gratuita.

2.2.7. Tecnologías de desarrollo

Según Layedra et al. (2022), afirma que las tecnologías de desarrollo engloban las herramientas, plataformas, lenguajes de programación, frameworks y entornos empleados por los desarrolladores en la creación de software y aplicaciones. Su importancia radica en simplificar el procedimiento de la creación del software, incrementar la eficiencia y ofrecer un entorno coherente que favorece a los desarrolladores.

2.2.7.1. Lenguajes de programación

Es un sistema formal que permite a los programadores transmitir instrucciones a una máquina, generalmente una computadora, mediante el uso de símbolos y códigos específicos. Estos expertos utilizan estos lenguajes para crear y desarrollar aplicaciones móviles que posteriormente son utilizadas a diario por los usuarios. En el ámbito de la programación, es crucial mencionar que hay una variedad de lenguajes disponibles específicamente diseñados para ciertos sistemas operativos, lo que significa que algunos son exclusivos para plataformas particulares (Layedra et al., 2022).

Tabla 8. Lenguajes de programación.

Lenguaje d programación	Plataformas compatibles	Tipo de aplicaciones	Ventajas	Desventajas
Java	Android, Windows, Linux	Móvil, web, escritorio	Tiene gran cantidad de librerías Portabilidad	Tiene gran consumo de recursos Es menos conciso que algunos lenguajes modernos
Swift	iOS, macOS	Móvil, escritorio	Sintaxis clara y concisa Tiene Desarrollo rápido y eficiente Compatibilidad con apple	Tiene compatibilidad limitada
Kotlin	Android	Móvil, escritorio, web	Compatibilidad con java Tiene mejoras en seguridad Posee mayor productividad	Soporte a medias
Dart	iOs, web, Android,	Móvil, escritorio	Soporte multiplataforma Tiene facilidad de aprendizaje	Nuevo a diferencia de otros lenguajes Tiene menos popularidad que otros lenguajes

Lenguaje d programación	Plataformas compatibles	Tipo de aplicaciones	Ventajas	Desventajas
JavaScript	iOS, Android, web	móvil, escritorio	web, Posee un rendimiento sólido en aplicaciones Gran cantidad de bibliotecas y frameworks Facilidad de utilizar Versatil	Poco rendimiento en aplicaciones complejas No es compatible en todos los navegadores

Como se muestra en la tabla 8, se realizó un análisis de varios lenguajes de programación para detectar cual es el más apto para la presente investigación, de acuerdo a la elección el lenguaje de programación Dart es el más adecuado puesto que, se fundamenta en su facilidad de aprendizaje y su adaptabilidad para diversas aplicaciones, incluyendo móviles, web y escritorio. Además, destaca por ser multiplataforma y exhibir un rendimiento sólido, convirtiéndolo en la opción ideal para este tipo de investigación.

2.2.7.2. Frameworks

Según Ticono (2022), menciona que un framework en el ámbito de la creación de programas informáticos representa una estructura conceptual y tecnológica que suministra funcionalidades predefinidas y establece una base organizativa para la creación de aplicaciones. Su propósito es simplificar y agilizar el proceso de creación al ofrecer un conjunto de reglas, convenciones y herramientas que capacitan a los desarrolladores para construir software de manera más eficiente y uniforme.

Tabla 9. Frameworks.

Framework	Plataformas compatibles	Lenguaje principal	Ventajas	Desventajas
React Native:	iOS, Android	JavaScript	Desarrollo rápido	Rendimiento en algunas situaciones puede ser inferior al desarrollo nativo
Flutter:	iOS, Android, Web	Dart	Amplia personalización de la interfaz de usuario según las preferencias individuales.	Mayor tamaño del programa
Xamarin:	iOS, Android, Windows	C#	Reutilización de código para múltiples plataformas	Menos comunidad en comparación con algunos otros frameworks
Ionic:	iOS, Android, Web	TypeScript	Desarrollo multiplataforma	Rendimiento puede ser inferior

Framework	Plataformas compatibles	Lenguaje principal	Ventajas	Desventajas
NativeScript:	iOS, Android, Web	JavaScript, TypeScript	Acceso nativo a API del dispositivo	a las aplicaciones nativas Curva de aprendizaje más pronunciada para algunos desarrolladores

Tal y como se muestra en la tabla 9, se realizó una comparativa entre frameworks en donde se puede evidenciar que Flutter es el más calificado para el presente proyecto considerando que es una solución con una multiplataforma con una interfaz de usuario altamente personalizable y un ciclo de desarrollo rápido. Por otra parte, Flutter es un framework de desarrollo de interfaz de usuario (UI) el cual cuenta con un desarrollo rápido, es decir, ofrece un ciclo con funciones de recarga en caliente que permiten ver los cambios en tiempo real sin reiniciar la aplicación por completo.

2.2.8. Manejadores de Base de Datos

Según Marín (2019), lo define como un sistema de gestión de database como una herramienta que se especializa en la creación, control y supervisión de bases de datos, abordando aspectos como también la selección y administración de las estructuras esenciales para almacenar y recuperar datos de manera óptima.

2.2.8.1. PostgreSQL

Según Borges (2019), menciona que PostgreSQL se destaca en ser un sistema de código abierto en cuál no tiene complejidad en descargarlo, debido a que no tiene un costo adicional, más bien es gratuito por ello, es una buena opción emplearle en proyectos por su buen rendimiento. Además, es muy popular en aplicaciones empresariales y de alto rendimiento debido a sus capacidades de gestión de datos robustas y flexibles. Además, es ampliamente utilizado en aplicaciones web y en la ciencia de datos. Fue diseñado para ser resistente a fallos y con la habilidad para manejar enormes cantidades de información de manera efectiva. con un rendimiento excepcional.

Tabla 10. Beneficios y consecuencias de PostgreSQL

Ventajas	Desventajas
Instalación y uso gratuito	Elaborado para ambientes con una considerable cantidad de información.
Sistema multiplataforma	Es posible que el sistema exhiba lentitud en sus implementaciones para bases de datos.
Estabilidad	La manipulación de la sintaxis o los comandos puede no ser sencilla de ejecutar.
Escalabilidad y configuración	Puede volverse complicado utilizarlo si no se cuenta con el conocimiento necesario.

2.2.8.2. MySQL

De la misma manera se tiene a MySQL el cual figura como uno de los sistemas de gestión de database más reconocidos y ampliamente empleados a nivel global. Según Robledano (2019), menciona que MySQL se utiliza para guardar, organizar y recuperar información en diversos contextos, que van desde sitios web y aplicaciones móviles hasta sistemas corporativos. Este sistema tiene la capacidad de gestionar volúmenes importantes de información y puede ajustarse según las demandas en constante evolución de una aplicación. Ofrece una serie de características avanzadas y transacciones. También es compatible con diversos lenguajes de desarrollo, tales como PHP, Java, C#, Python, entre otros.

Tabla 11. MySQL

Características	Ventajas	Desventajas
Ofrece la opción de elegir entre varios motores de almacenamiento para cada tabla.	Es de uso libre y gratuito.	Muchas soluciones no se encuentran documentadas o carecen documentación respaldada por fuentes oficiales.
Combina transacciones de diversas formas mediante conexiones para aumentar la velocidad de procesamiento por segundo, al tiempo que asegura una conexión segura.	Software que cuenta con la licencia GPL	La documentación no está disponible para algunas de sus funcionalidades.
Realiza operaciones de compra y venta y utiliza relaciones con claves externas.	Requisitos de bajo coste.	En la actualidad, no se caracteriza por ser el programa más fácil de usar.
Proporciona una amplia gama de funcionalidades dentro del lenguaje SQL.	No requiere de hardware o software de alta potencia. Cuenta con una extensa comunidad de usuarios y desarrolladores que participan activamente con el desarrollo y soporte.	Presenta una deficiencia cuando requieren de una modificación. Configurar y optimizar MySQL para un rendimiento específico puede ser complejo y requerir conocimientos técnicos avanzados.
Amplia capacidad para buscar e indexar información en forma de texto.	Rendimiento eficiente, especialmente en	Puede tener limitaciones en la gestión de transacciones y en

Características	Ventajas	Desventajas
	operaciones de lectura.	de la implementación de ciertas funciones avanzadas.
Utiliza una variedad de herramientas para mejorar la portabilidad.	Cumple con los estándares SQL y ofrece compatibilidad con varios lenguajes de programación.	No es la mejor opción para almacenar datos no estructurados.
Tiene la capacidad de gestionar cantidades masivas de datos, incluso cuando superan los 50 millones de registros.	Herramienta especializada en la administración de bases de datos estable y confiable.	Algunas características avanzadas y herramientas de administración pueden requerir la versión comercial de MySQL.
Utilización de múltiples hilos a través de hilos del núcleo del sistema operativo.	Escalable, lo que significa que se maneja grandes cantidades de datos y un alto volumen de transacciones.	La configuración incorrecta o la falta de actualizaciones pueden poner en riesgo la seguridad del sistema.

2.2.8.3. SQL server

SQL Server se refiere a una plataforma para la gestión de database que admite distintos lenguajes de programación. De acuerdo con Según Parada (2019), SQL Server, desarrollado por Microsoft, es una herramienta para gestionar database relacionales. Reconocido como uno de los sistemas más conocidos y utilizados en el mercado, se utiliza para almacenar, gestionar y recuperar información en diversos contextos empresariales, sitios web y aplicaciones móviles. Además, permite la conexión con diferentes tipos de datos, es decir textos números, imágenes, fechas, ofrece diversas herramientas para facilitar tanto la administración como el desarrollo de bases de datos. Incluye funciones avanzadas como Business Intelligence, inteligencia artificial, análisis de datos brindando seguridad de la información con una escalabilidad alta.

Tabla 12. SQLserver

Ventajas	Desventajas
Optimo rendimiento en entornos Windows.	Los costos de licencias y hardware son elevados.
Fuertes medidas de autenticación.	Configuración complicada en políticas de seguridad.
Escalabilidad vertical y horizontal.	Es necesario tener precaución al escalar horizontalmente.
Amplia variedad de herramientas y compatibilidad.	Algunas herramientas son exclusivas de Microsoft.
Administración sencilla a través de SQL Server Management Studio.	La interfaz de administración puede resultar abrumadora.

Tabla 13. Bases de Datos en aplicaciones móviles.

Base de datos		Características	Ventajas	Desventajas
SQLite		Es una plataforma enfocada a la gestión de bases de datos relacionales	Se destaca por su ligereza y fácil integración.	No es la opción más adecuada para aplicaciones de gran envergadura.
Firestore	Realtime	Constituye una database en tiempo real basada en la nube.	Permite actualizaciones en tiempo real.	Presenta limitaciones en consultas y filtros avanzados.
Firestore		Se trata de una database que opera en tiempo real y utiliza un modelo NoSQL.	Ofrece escalabilidad y permite consultas más complejas.	Su costo está asociado al volumen de operaciones.
Realm		Es una base de datos especialmente creada para su uso en aplicaciones móviles.	Destaca por su rapidez y eficiencia en operaciones de lectura y escritura.	Puede requerir ajustes para adaptarse a ciertos casos de uso específicos.
Couchbase Mobile		Es una base de datos NoSQL diseñada para aplicaciones móviles	Gestiona eficientemente grandes conjuntos de datos.	Su implementación demanda familiaridad con el enfoque NoSQL.

2.2.9. Metodologías de desarrollo

Las estrategias de desarrollo de software son componentes fundamentales en el procedimiento de creación y actualización de software con un buen estándar de calidad, cuyo objetivo primordial es satisfacer los requisitos de los usuarios. Además, considera que estas metodologías son una parte fundamental de la ingeniería de software y las define como un conjunto coherente de métodos interrelacionados, guiados por principios comunes (Rivas et al., 2019).

Tabla 14. Metodologías de Desarrollo.

Metodología	Definición	Características	Ventajas	Desventajas
XP	Representa un enfoque ágil de desarrollo que se enfoca en la mejora constante del software mediante la colaboración estrecha entre desarrolladores y clientes (Raeburn, 2022).	Sus características incluyen desarrollo iterativo, pruebas frecuentes, programación en parejas e integración continua.	Sus beneficios son la adaptabilidad a cambios, entrega rápida y la alta calidad del código.	Sin embargo, puede ser intensiva en recursos y requerir una cultura organizacional específica.
Kanban	Es una metodología de gestión visual centrada en la mejora continua y la entrega constante de trabajo Mesh (2020),	Sus rasgos distintivos son un tablero visual, límites de trabajo en progreso y un flujo continuo.	Ofrece flexibilidad, una clara visualización del trabajo y optimización del flujo de trabajo como ventajas.	No obstante, es menos estructurada que otras metodologías y puede requerir disciplina para su implementación
RAD	Se enfoca en el desarrollo que prioriza la	Características clave incluyen	Entre sus ventajas se	Sin embargo, puede

Metodología	Definición	Características	Ventajas	Desventajas
	rapidez y adaptabilidad, utilizando prototipos y ciclos cortos de desarrollo Catro (2019),	desarrollo iterativo, prototipos y la participación del usuario.	encuentran entregas rápidas, adaptabilidad a cambios y la participación del cliente.	comprometer la calidad en favor de la velocidad y no es adecuada para todos los proyectos.
Scrum	Se centra en la entrega incremental y equipos autoorganizados(Estrada et al., 2021).	Elementos distintivos incluyen sprints, roles definidos, y reuniones regulares.	Proporciona transparencia en el progreso, adaptabilidad y entregas frecuentes como ventajas.	No obstante, requiere roles y responsabilidades bien definidos, lo que puede ser desafiante para equipos nuevos.

Tal y como se evidencia en la tabla 14, muestra la Metodología XP se ha seleccionado para el marco metodológico principal para esta investigación debido a varios motivos clave. En primer lugar, se destaca su capacidad para fomentar una interacción continua y fluida entre el cliente y desarrollador. Esta comunicación personalizada es fundamental para entender lo que el cliente necesita y espera, lo que a su vez facilita la identificación temprana de posibles problemas y su pronta resolución. El entorno en donde se lleva a cabo el desarrollo de software los requisitos pueden evolucionar rápidamente, esta capacidad de adaptación es invaluable. Otro aspecto primordial de esta es su flexibilidad en los plazos de desarrollo de la aplicación. A diferencia de enfoques más rígidos como Kanban, XP permite la inclusión de nuevas tareas en cualquier fase del ciclo de desarrollo. Esta flexibilidad es esencial cuando surgen nuevas necesidades o requisitos durante el proceso de desarrollo, lo que permite al equipo ajustarse rápidamente y priorizar de manera efectiva las tareas adicionales.

En comparación con Kanban, XP ofrece una ventaja significativa al permitir la adición de tareas adicionales en cualquier etapa del proceso. En Kanban, la falta de flexibilidad puede resultar restrictiva, puesto que se requiere que una tarea sea completada antes de que se pueda agregar una nueva. Esta rigidez puede ralentizar el proceso y dificultar la capacidad del equipo para adaptarse a modificaciones en los requerimientos del proyecto.

Además de su versatilidad y capacidad de adaptación, la Metodología XP fomenta un ambiente de trabajo colaborativo y participativo. Todos los miembros del equipo son alentados a contribuir con ideas y soluciones, lo que no solo enriquece el proceso de desarrollo, sino que también aumenta el sentido de propiedad y compromiso con

el proyecto. Esta cultura de colaboración fue un factor determinante en la elección de la Metodología XP para esta investigación, por ende, se reconoció su capacidad para potenciar el talento y la creatividad de todos los involucrados.

III. METODOLOGÍA

3.1. ENFOQUE METODOLÓGICO

3.1.1. Enfoque

Según Salas (2019), se destaca al enfoque mixto por su capacidad para integrar tanto datos cualitativos como cuantitativos en un solo estudio. Este enfoque reconoce la complementariedad y la sinergia entre ambos tipos de datos, lo que permite un entendimiento más exhaustivo y profundo de los fenómenos que están siendo investigados.

En el caso específico del estudio sobre el cultivo de arvejas, el enfoque mixto ofrece una metodología sólida para abordar la complejidad de los procedimientos agrícolas y los desafíos asociados con la salud de las plantas. Al emplear métodos cualitativos, los investigadores pueden explorar en profundidad los procesos y prácticas utilizadas en el cultivo de arvejas, así como comprender las perspectivas y experiencias de los agricultores, expertos en agronomía y otros actores relevantes en el campo.

Por otro lado, el enfoque cuantitativo permite recopilar datos numéricos objetivos sobre variables específicas relacionadas con la salud de los cultivos de arvejas, como la temperatura, el crecimiento de las plantas y la humedad del suelo. Estos datos cuantitativos proporcionan un sólido punto fundamental para realizar un estudio y la identificación de patrones o tendencias en el comportamiento de los cultivos.

Al combinar ambos enfoques, se obtiene una imagen más completa y detallada de la salud y el rendimiento en los cultivos de arvejas. Esta integración de datos cualitativos y cuantitativos facilita una comprensión holística de la realidad agrícola, al mismo tiempo puede realizar estrategias más efectivas para mejorar la productividad y calidad de sembríos de arvejas.

3.1.2. Tipo de Investigación

Dentro del enfoque mixto, se han considerado varios tipos, como la investigación descriptiva, explicativa, documental o bibliográfica, de campo y exploratoria.

3.1.2.1. Investigación descriptiva

Según Gevara et al. (2020), menciona que esta investigación tiene como finalidad explorar el qué y el porqué de un estudio, buscando describir y explicar a fondo el objeto de investigación con el propósito de proporcionar una solución clara a la cuestión tratada. Además, se destaca por la recopilación exhaustiva de información y por el reconocimiento meticuloso de variables, así como la realización de estudios transversales.

Este proyecto se enmarca con una investigación descriptiva, detalla las características de la visión artificial juntamente con la aplicación en la agricultura. Asimismo, proporciona información detallada sobre los requisitos edafoclimáticos en donde se desarrollan las enfermedades dentro del cultivo del guisante. Así mismo la investigación implica la recopilación de datos, resultado de un extenso estudio, que abordará las técnicas utilizadas en el cultivo de arvejas.

3.1.2.2. Investigación explicativa

Según Bastis (2020), afirma que la investigación explicativa investiga un fenómeno específico que es nuevo o que no ha sido investigado a fondo. El propósito de dicha investigación es proporcionar conocimiento relevante sobre ellos.

Se emplea la investigación explicativa, en vista de que se analizan y explican los fenómenos que tienen lugar en los cultivos de arveja. Se exponen las circunstancias propicias para la revelación de las enfermedades más comunes en los sembríos de arveja, así como las circunstancias de crecimiento de la leguminosa. Además, se enfatiza que la visión artificial como una herramienta fundamental en el sector agrícola. Del mismo modo, se ofrece una explicación sobre la utilización de recursos y se realiza un análisis de todos los aspectos como algoritmos, metodologías, database, motores.

3.1.2.3. Investigación bibliográfica

Según Zorrilla (2021), describe que este estudio se centra en examinar la información disponible sobre el tema de interés. Esta revisión constituye uno de los semblantes primordiales en diferentes investigaciones y sus respectivas fuentes. Es un proceso crucial que involucra una serie de fases, que incluyen la observación, investigación, interpretación y síntesis, con el fin de reunir información relevante para el tema de interés.

Además, se enfoca en la búsqueda de fuentes bibliográficas, mediante la exploración de tesis y artículos científicos disponibles en formato digital se realiza una conexión entre investigaciones auténticas que respaldan el trabajo. Además, esta práctica facilita la orientación adecuada de la investigación.

3.1.2.4. Investigación exploratoria

El propósito es obtener datos, recopilar información que, a su vez, posibilite una comprensión precisa. Asimismo, se orienta hacia la exploración de temas específicos, buscando abordarlos desde nuevos enfoques o descubrir aspectos novedosos que merecen ser estudiados (Vizcaíno et al., 2023).

Se emplea la investigación exploratoria, puesto que se realiza análisis preliminar para reconocer áreas de estudio, lo que implica la definición precisa de los problemas que enfrentan los cultivos de arveja en la comunidad de La Delicia. Se reúne la información variada para orientar el desarrollo de este trabajo. De igual manera, se utiliza para la identificación de la problemática específica en la detección temprana de enfermedades en los sembríos de arveja, lo que contribuye a mejorar la comprensión de esta situación sin llegar a conclusiones definitivas.

3.1.2.5. Investigación de campo

Se emplea cuando involucra la interacción con el entorno, lo que facilita la identificación del procedimiento más idóneo para el sector agrícola mediante la utilización de los avances tecnológicos disponibles. Es decir que, se basa en la realidad objetiva en lugar de percepciones subjetivas. Asimismo, los resultados obtenidos orientan a los desarrolladores hacia las necesidades reales de los agricultores. Esto se logra a través de la ejecución de entrevistas, encuestas y análisis de los datos recopilados durante el proceso.

3.2. IDEA A DEFENDER

El uso de visión artificial en la agricultura de precisión facilitará la identificación de enfermedades en las primeras etapas del cultivo de arveja variedad Quantum en la comunidad de La Delicia, ciudad de San Gabriel.

3.3. DEFINICIÓN Y OPERACIONALIZACIÓN DE LAS VARIABLES

Tabla 15. Operacionalización de variables

Variable	Definición	Dimensión	Indicadores	Técnica	Instrumento
Independiente	Visión artificial en la agricultura de precisión: Según Garcés (2022), es una tecnología que respalda significativamente a la agricultura de precisión al ofrecer diversas herramientas diseñadas para reducir las dificultades que enfrenta el agricultor durante el proceso de siembra.	Aprendizaje del algoritmo	Tiempo de aprendizaje del modelo	Entrevista Encuesta	Cuestionario de preguntas abiertas
			Cantidad de información		Cuestionario de preguntas cerradas
			Calidad de la información		
		Algoritmo a estudiar	Costes de ejecución		
			Precisión de algoritmo		
			Procesamiento de imágenes	de Interpretación de resultado	
			Etapas de procesamiento		
Dependiente	Identificación de enfermedades en cultivos de arveja Quantum: Según Cordova (2022), destaca la importancia de estrategias para prevenir enfermedades en el cultivo de arvejas, recomendando control biológico, manual y físico, así como control químico específico. Estas estrategias ofrecen un enfoque integral para garantizar un cultivo saludable y productivo.	Características de la enfermedad	Tamaño	Entrevista Encuesta	Cuestionario de preguntas abiertas
			Color		
			Lesiones en la hoja		Cuestionario de preguntas cerradas
			Textura		
			Ciclo de enfermedad		

Requerimientos
edafoclimáticos

PH del suelo

Clima óptimo

Tipo de suelo

Iluminación

3.4. MÉTODOS UTILIZADOS

Se utilizó la metodología de investigación de acción en la elaboración del proyecto, según Salas (2019), esta investigación se concibe como un estudio del contexto social, caracterizado por un proceso de investigación en espiral, donde la investigación y la intervención se llevan a cabo simultáneamente. Por lo tanto, esta investigación se considera apropiada para la elaboración del proyecto. Se ejecutó una investigación exhaustiva sobre el uso de la visión artificial enfocada en la agricultura de precisión para detectar de forma temprana enfermedades en sembríos de arveja Quantum. Esto permitió desarrollar una solución informática para detectar, tomar acciones correctivas y proteger el cultivo. Es importante ser práctico, realizando tareas que impliquen autorreflexión e investigación, así como interpretar y comprender las señales emitidas por el prototipo y realizar las modificaciones necesarias para mejorarlo.

3.4.1. Método deductivo

Para Villafuerte y Tóala (2020), es un proceso de investigación que emplea diversos métodos de razonamiento, que van desde la aplicación de la lógica general hasta la consideración de hechos particulares fundamentados en leyes o principios. En esencia, se trata de un enfoque lógico destinado a extraer conclusiones a partir de una serie de principios establecidos.

Este método en la investigación se centró en la aplicación de principios teóricos y marcos conceptuales existentes en la visión artificial y la identificación temprana de enfermedades en la agricultura de precisión. Basándose en teorías previas sobre el procesamiento de ilustraciones y algoritmos de detección de objetos, se formuló predicciones específicas sobre cómo estos principios pueden aplicarse a la detección de enfermedades en plantaciones de arveja Quantum, a partir de teorías sobre la discriminación de patrones en imágenes, se pueden derivar predicciones sobre la eficacia de ciertos algoritmos para distinguir entre plantas sanas y afectadas por enfermedades.

3.4.2. Método inductivo

Según Quesada y Medina (2020), afirma que el enfoque inductivo implica identificar las características comunes o generales presentes en diversas situaciones para luego establecer relaciones causales y formular proposiciones de validez universal o leyes científicas. Este método se basa en un razonamiento que parte de lo específico y

concreto para llegar a lo general y abstracto. Se sostiene comúnmente que la inferencia inductiva es un tipo de razonamiento dirigido hacia un objetivo específico. El enfoque inductivo en esta investigación implicó la observación directa de los síntomas de enfermedades presentes en los cultivos de arveja Quantum, así como la recopilación de datos sobre sus patrones de propagación y factores ambientales asociados. A través de este proceso de observación detallada, se buscó patrones o irregularidades que puedan sugerir relaciones significativas entre los síntomas y otros factores. Al estudiar la aparición de manchas foliares en los cultivos, se puede identificar patrones estacionales o condiciones climáticas específicas que podrían influir en la prevalencia de la enfermedad.

3.4.3. Técnicas

Los métodos empleados en este proyecto ayudaron a obtener información relevante de la detección y las diferentes enfermedades dentro del proceso del sembrío de arveja Quantum, por lo cual se sustentó los datos obtenidos mediante la recolección de información con los instrumentos utilizados.

3.4.3.1. Observación participativa

En esta observación se eligió un lugar, el mismo que se va a estudiar para observar de manera sistémica el objeto en cuestión y además se tendrá una reunión con el señor José Rosalino Tutachá Paillacho, agricultor del sector la delicia.

3.4.3.2. Encuesta

Según García (2020), menciona que una encuesta consiste en una serie de preguntas estandarizadas dirigidas a un grupo representativo de la sociedad, con el propósito de indagar sobre opiniones o aspectos que puedan influir en ellos.

Como estrategia, se decidió llevar a cabo una encuesta dirigida a diversos conocidos especializados en el cultivo de arvejas en las distintas comunidades de San Gabriel. El objetivo principal era obtener un entendimiento más profundo sobre el cultivo, incluyendo posibles enfermedades que podrían surgir durante el proceso de siembra. Este enfoque tenía como finalidad recopilar información detallada y precisa que sería de gran utilidad para nuestra investigación, los datos obtenidos de cada agricultor nos brindarían información valiosa.

3.4.3.3. Entrevista estructurada

Según Lázaro (2021), Se caracteriza por seguir un conjunto de preguntas preestablecidas, que suelen ser abiertas en su mayoría, a diferencia de las preguntas cerradas o de opción múltiple que se utilizan en un cuestionario. En este tipo de entrevista, todos los entrevistados responden a las mismas preguntas, que se formulan de la misma manera y en un orden predeterminado.

Se empleó esta técnica con el propósito de adquirir un entendimiento adecuado de los diversos procesos aplicados en el cultivo de la arveja, para ello se implementa un lugar de información sobre cuáles son las técnicas, el proceso y el tratamiento en los sembríos de arveja.

3.5. ANÁLISIS ESTADÍSTICO

3.5.1. Población y muestra

La muestra de la presente investigación consiste en 384 individuos que trabajan en el sector agrícola, según los registros de la cuadrilla de trabajadores de la comunidad de la Delicia. Para obtener el tamaño muestral, se procedió hacer la utilización de la fórmula propuesta por Velázquez (2019), con 95% de confianza, una desviación estándar de 1.96 y una probabilidad de éxito del 50%. Se estableció un margen de error de estimación del 0.05%.

$$n = \frac{N * Z_{\alpha}^2 * p * q}{e^2 * (N - 1) + Z_{\alpha}^2 * p * q}$$

Donde:

N = población

$Z_{\alpha}^2 = 1.96^2$ nivel de confianza

p = probabilidad de acierto % = 0.5

q = probabilidad de fracaso $1 - p$. 0.95

e = error de estimación 0.05%

$$n = \frac{384 * 1.96^2 * 0.5 * 0.5}{0.05^2 * (384 - 1) + 1.96^2 * 0.5 * 0.5}$$

$$n = 192.2 = 192$$

Utilizando la fórmula mencionada, se determinó que 192 agricultores son escogidos para la recopilación de datos de vital importancia usando encuestas vía internet con preguntas adaptadas para una mejor comprensión de estos.

3.6. RECURSOS

3.6.1. Recursos materiales

Esta investigación se basa en una variedad de fuentes, incluyendo tesis tanto físicas como digitales, libros como referencias de investigación, artículos, papers y la guía metodológica para el plan de TIC de la UPEC. Como también se emplean portátiles para crear el prototipo. Los recursos utilizados en la elaboración son los siguientes:

- Computadoras para investigar y desarrollar el documento.
- Internet para recopilar y buscar información necesaria para el proyecto.
- Se emplearon memorias USB y discos duros externos como dispositivos de almacenamiento.

3.6.2. Recursos humanos

Los estudiantes llevarán a cabo este proyecto gracias a la asistencia del personal docente, incluyendo al Msc. Carlitos Guano, quien proporcionará orientación para su desarrollo y realizará correcciones pertinentes.

Tabla 16. Recursos Humanos

Nombres	Ocupación	Rol
Msc. Carlitos Guano	Director de Carrera y docente de la UPEC	Tutor
Kimberlhy Pérez Wilmer Tutachá Sr. José Tutachá	Estudiantes de la Carrera de Computación Agricultor	Desarrolladores Especialista en identificación de enfermedades presentes en cultivos de arveja.

3.6.3. Recursos financieros

Para llevar a cabo el presente trabajo, se ha designado un monto presupuestario que posibilitará su desarrollo. A continuación, se presenta la tabla 17, detallando dichos recursos financieros.

Tabla 17. Recursos Financieros

Recursos	Cantidad	Tiempo	Precio unitario	Total
Internet	20	mensual	\$30	\$600
Energía eléctrica	20	Mensual	\$35	\$700
Transporte	20	Mensual	\$20	\$200
Vivienda (arriendo)	20	Mensual	\$80	\$1600
Total				\$3100

3.6.4. Recursos tecnológicos

A continuación, se detallarán las herramientas de hardware y software empleadas en el desarrollo y ejecución de este proyecto.

Hardware

Para la realización de este trabajo, se hacen necesarias dos herramientas tecnológicas. Se empleará un dispositivo móvil (celular) para la captura de imágenes de las hojas del cultivo, las cuales serán posteriormente sometidas a análisis.

- Sistema operativo: Android 13
- Cámara: 50 megapíxeles
- Resolución: 8160x6120 píxeles
- Distancia focal: 26mm (lente)
- Memoria RAM: 4GB LPDDR4X
- Memoria interna: 128GB (101GB disponible) eMMC 5.1

Para el proceso de programación, pruebas de corrección y análisis, se utilizó una laptop con las siguientes especificaciones:

- Marca: VivoBook_ASUS
- Procesador: AMD Ryzen 7 4700U with Radeon Graphics 2.00 GHz
- Memoria Ram: 8,00 GB
- Sistema operativo: Windows 10 Pro

Software

- Visual Studio Code 1.86.1
- Android Studio 2022.3
- Python 3.10.11
- Flutter 3.16.5
- Yolov8
- Dart

IV. RESULTADOS Y DISCUSIÓN

4.1. RESULTADOS

4.1.1. Análisis de la encuesta

Gracias a las 192 encuestas realizadas a los agricultores a continuación se presenta el análisis respectivo

1. ¿Cuántos años de experiencia tiene usted como agricultor?

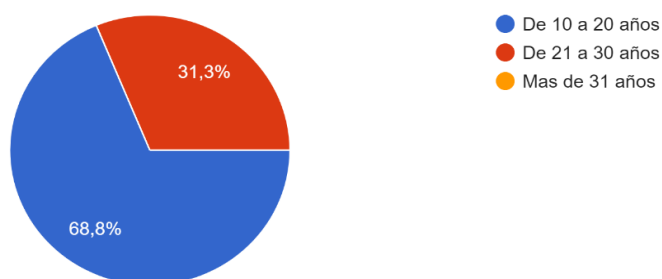


Figura 1. Experiencia del agricultor

Análisis e interpretación

La experiencia de un agricultor desempeña un papel fundamental en el proceso agrícola. Su conocimiento sobre la preparación del suelo, la selección de semillas y la aplicación de fungicidas es crucial para enfrentar las enfermedades que puedan surgir. En el sector de La Delicia, la mayoría de los trabajadores están dedicados a la agricultura, por lo que esta acción genera ingresos económicos altos. Según lo reflejado en la figura 1, el 31% de los agricultores tienen entre 21 y 30 años de experiencia, mientras que un 68,8% cuentan con una experiencia de 10 a 20 años en el cultivo. Estos datos evidencian que la comunidad agrícola local está compuesta mayormente por individuos con un sólido conocimiento y experiencia en el sector, por lo que contribuye significativamente al desarrollo de la agricultura en la zona.

2. ¿Cuántas hectáreas destina usted para el cultivo de arveja?

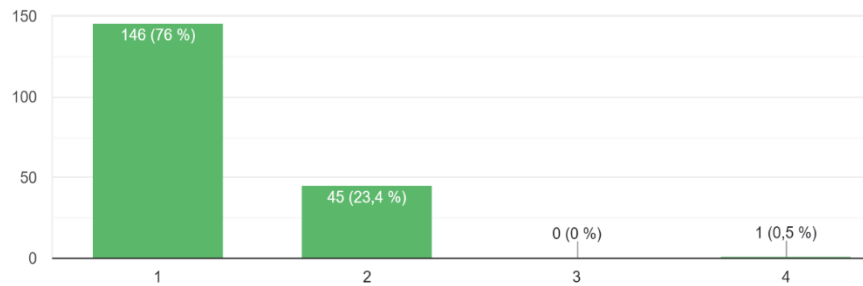


Figura 2. Hectáreas destinadas al cultivo

Análisis e interpretación

Antes de proceder con la siembra, los agricultores evalúan si disponen de la cantidad adecuada de semillas para cubrir la extensión de tierra requerida, ya sea una hectárea completa o menos. En ocasiones, descubren que no es necesario sembrar toda una hectárea para lograr una buena producción. Gracias a los datos recopilados por la encuesta, en la figura 2, el 76% de los agricultores, tienen la capacidad de sembrar una hectárea. Mientras tanto, el 23.4% restante, cuentan con dos hectáreas de tierra. Esto sugiere una distribución adecuada de tierras para la práctica agrícola en el sector.

3. ¿Cuáles son las variedades de arveja que usted siembra en la zona?

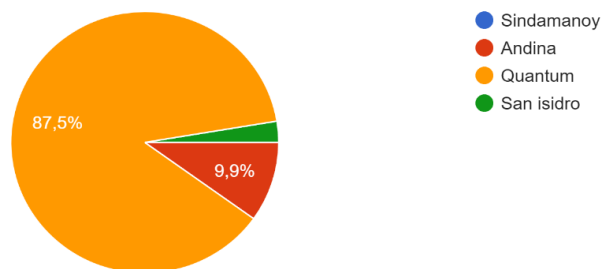


Figura 3. Variedades de arveja más comunes en la zona

Análisis e interpretación

Los agricultores toman decisiones sobre qué variedad de arveja sembrar y cuál evitar basándose en su experiencia previa y conocimiento del rendimiento del producto. Esto incluye evaluar si una variedad ha demostrado ser productiva o no. Según se muestra en la figura 3, el 87.5% de los agricultores eligen sembrar la variedad de arveja

Quantum, mientras que el 9.9% optan por la variedad Andina y el 2.6% prefieren la variedad San isidro. Estos datos indican claramente que la arveja Quantum es la más cultivada en el sector de La Delicia.

4. ¿Cuántas veces en el año siembra ALVERJA?

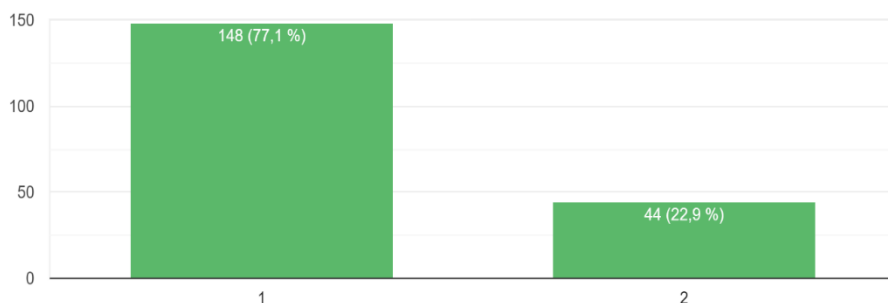


Figura 4. Siembras al año

Análisis e interpretación.

La decisión de cuándo sembrar a lo largo del año está condicionada por factores climáticos que influyen considerablemente en el cultivo de arveja. Según se muestra en la figura 4, el 77.1% de los agricultores elige sembrar una vez al año, mientras que el 22.9% restante prefiere realizar dos siembras anuales. Esto sugiere que la mayoría de los agricultores optan por realizar una sola siembra al año.

5. ¿Seleccione la o las enfermedades más frecuentes que se presentan en el cultivo de arveja?

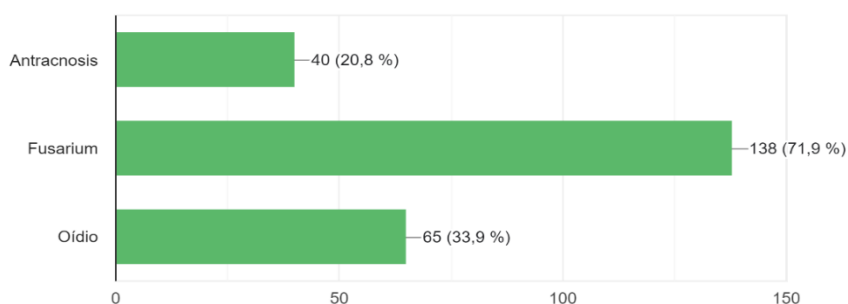


Figura 5. Enfermedades en el cultivo

Análisis e interpretación

Las enfermedades en los cultivos, particularmente en los de arveja, han sido una preocupación persistente a lo largo del tiempo, influidas tanto por factores climáticos como por prácticas agrícolas. La adopción de medidas de control es crucial para garantizar una producción saludable. En la figura 5, se observa que el 71.9% de los

agricultores identifican al fusarium como la enfermedad más común, seguido por el odio con un 33.9% y la antracnosis con un 20.8%.

6. ¿Seleccione la o las plagas más frecuentes que se presentan en el cultivo de arveja?

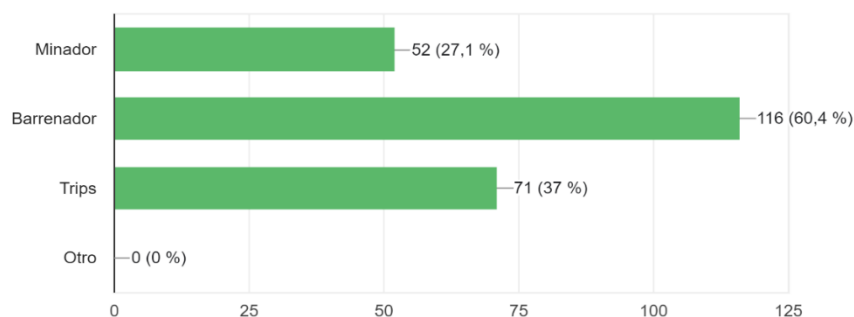


Figura 6. Plagas más frecuentes en el cultivo

Análisis e interpretación

Las plagas representan otro aspecto crucial en el proceso de cultivo, dado que pueden surgir en etapas tempranas y, de no ser controladas a tiempo, pueden ocasionar pérdidas económicas significativas y afectar el desarrollo óptimo de la producción. Tal como se puede visualizar en la figura 6, muestran que el 60.4% de los agricultores son afectados por el barrenador, seguido por el trips con un 37% y el minador con un 27.1%. Esto destaca que el barrenador es la plaga que más impacto tiene en el cultivo.

7. ¿Con que frecuencia realiza el monitoreo de plagas y enfermedades en su cultivo de arveja?

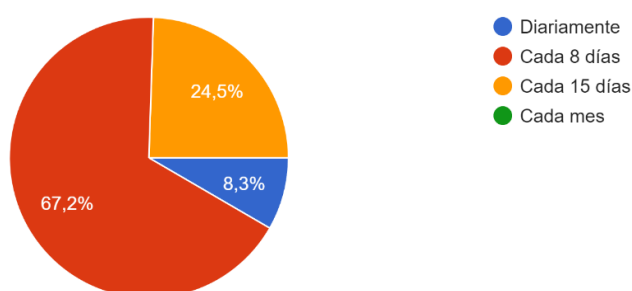


Figura 7. Monitoreo en el cultivo

Análisis e interpretación.

Es fundamental llevar a cabo un monitoreo adecuado del cultivo, puesto que el exceso de maleza puede propiciar la aparición de enfermedades o plagas.

Mantener el cultivo limpio es crucial para garantizar su salud. Según se puede apreciar en la figura 7, el 67.2% de los agricultores realiza un monitoreo cada 8 días, mientras que el 24.5% lo hace cada 15 días y el 8.3% diariamente. Esto resalta la preferencia por un monitoreo más frecuente, cada 8 días, como la práctica más efectiva.

8. ¿Considera usted necesario ayuda técnica de algún agrónomo para el manejo de plagas y enfermedades en el cultivo de arveja?

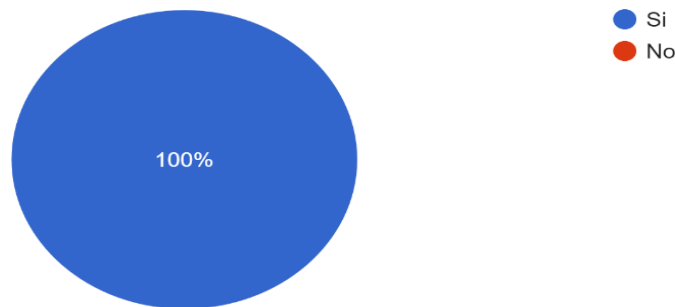


Figura 8. Indicaciones técnicas

Análisis e interpretación

Es fundamental contar con la ayuda técnica de un agrónomo para combatir tanto plagas como enfermedades en el cultivo. Una vez que el cultivo se ve afectado, las pérdidas para el agricultor pueden ser considerablemente altas. De acuerdo con lo representado en la figura 8, el total de agricultores están de acuerdo en la necesidad del asesoramiento técnico para el control de estas enfermedades.

9. ¿En base a su experiencia como identifica una enfermedad en el cultivo de arveja?

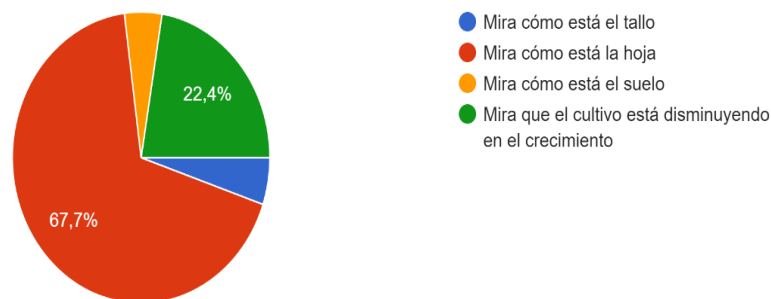


Figura 9. Formas para detectar una enfermedad en el cultivo

Análisis e interpretación

El proceso de identificación temprana de enfermedades es crucial para los agricultores, quienes buscan la asesoría de expertos para prevenir la propagación de enfermedades. Si se detecta una enfermedad en sus etapas iniciales, el agricultor puede intervenir a tiempo y evitar pérdidas en la producción. Sin embargo, si la enfermedad avanza sin control, las pérdidas pueden ser significativas. De acuerdo con la figura 9, el 67.7% de los agricultores se dedica a identificar indicios de enfermedades ubicadas en diferentes partes de la planta específicamente en las hojas, mientras que el 22.4% observa cualquier disminución en el crecimiento del cultivo. Además, un 5.2% presta atención al tallo y un 4.7% monitorea el estado del suelo. Esto demuestra que la mayoría de los agricultores se enfocan en la observación de las hojas para identificar las enfermedades presentes en el cultivo.

10. ¿Qué recomendaría hacer para obtener un cultivo libre de plagas y enfermedades?

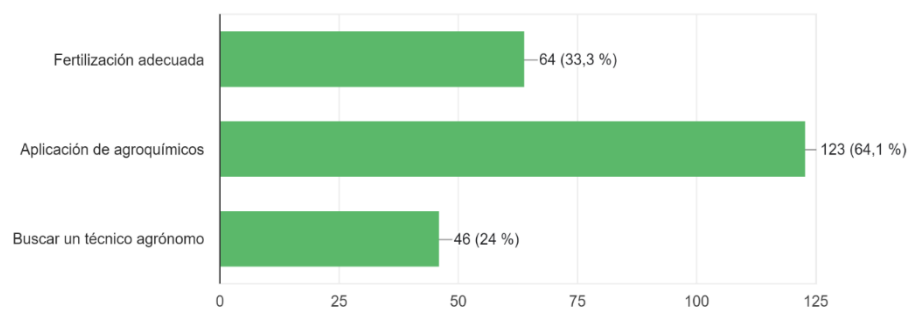


Figura 10. Recomendaciones para un cultivo libre de enfermedades

Análisis e interpretación

Es crucial tomar medidas preventivas al cultivar este tipo de vegetales para prevenir la transmisión y aparición de enfermedades. Esto implica realizar trabajos de desinfección de las semillas antes de realizar la siembra con el objetivo de mitigar cualquier anomalía. Según se muestra en la figura 10, el 64.1% de los agricultores opta por aplicar agroquímicos, el 33.3% prefiere una fertilización adecuada, y el 24% busca el asesoramiento de un agrónomo. Esto refleja la preferencia mayoritaria por el uso de agroquímicos.

11. Usted como agricultor ¿cree que el uso de fungicidas funciona para controlar enfermedades en el cultivo?

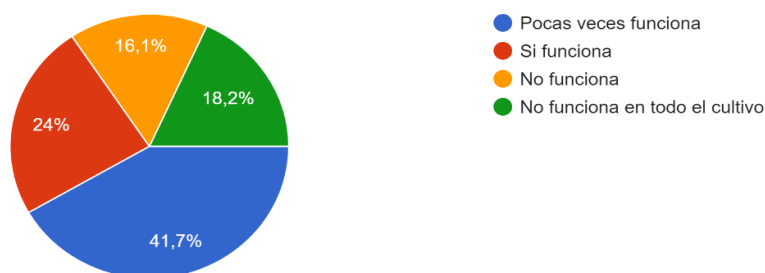


Figura 11. Uso de fungicidas para reducir enfermedades en el cultivo

Análisis e interpretación

La utilización de fungicidas constituye un recurso importante para los agricultores, puesto que su experiencia les proporciona conocimientos sobre qué fungicidas aplicar y cómo dosificarlos para controlar enfermedades. Sin embargo, en casos extremos, puede resultar difícil controlar la enfermedad. Según se muestra en la figura 11, el 41,7% de los agricultores menciona que los fungicidas funcionan solo pocas veces, el 24% indica que sí funcionan, el 18,2% señala que no funcionan en todo el cultivo y el 16,1% indica que directamente no funcionan.

12. ¿En caso de existir una aplicación para su teléfono celular que identifique enfermedades, lo utilizaría?

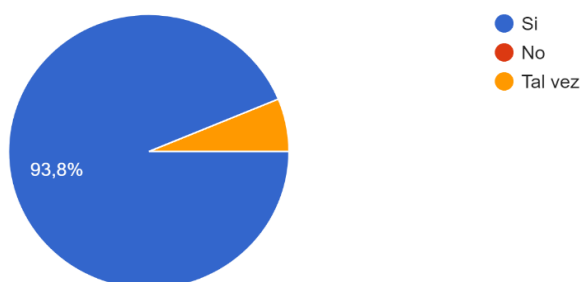


Figura 12. Aplicativo para la detección de enfermedades

Análisis e interpretación

Para los agricultores, contar con asistencia técnica, y preferiblemente tecnológica, es fundamental para reducir pérdidas y aumentar la producción. Esto se refleja en la figura 12, donde el 93,8% de los encuestados afirma que utilizaría una aplicación

móvil, mientras que solo el 6.3% indicó que no lo haría. Es evidente que la mayoría está de acuerdo en utilizar herramientas tecnológicas para mejorar su producción.

13. ¿Cree usted que el uso de herramientas tecnológicas ayudaría a detectar enfermedades en el cultivo de arveja?

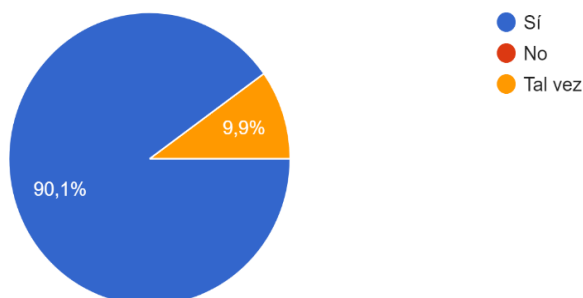


Figura 13. Uso de herramientas tecnológicas

Análisis e interpretación

Según lo expresado por los agricultores, llevan a cabo la detección de enfermedades de manera tradicional, examinando el tallo, las hojas o el suelo. Sin embargo, reconocen la utilidad de contar con ayuda tecnológica para una detección más precisa de las enfermedades que afectan a sus cultivos. Como se observa en la figura 13, el 90.1% está de acuerdo con el uso de herramientas tecnológicas, mientras que el 9.9% restante no lo está.

14. ¿Cuál de las siguientes opciones debería tener la aplicación para identificar enfermedades de su interés?

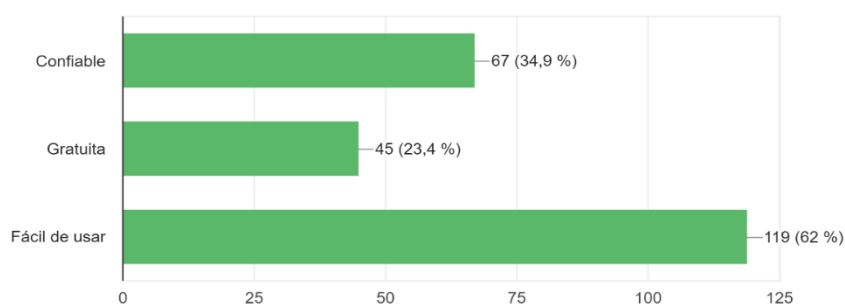


Figura 14. Opciones del aplicativo

Análisis e interpretación

Según lo conversado con los agricultores, consideran que la aplicación debe ser fácil de utilizar, puesto que no tienen un amplio conocimiento en tecnología. En la figura 14, se puede notar que el 62% de los participantes en la encuesta prefiere una opción

fácil de usar, seguido por el 34% que prioriza la confiabilidad y el 23.4% que busca una alternativa gratuita. Esto indica que la mayoría se inclina por una aplicación que sea sencilla y de manejo accesible.

15. ¿Con el uso de la visión artificial usted cree que se puede mejorar la calidad y rendimiento de nuestros cultivos?

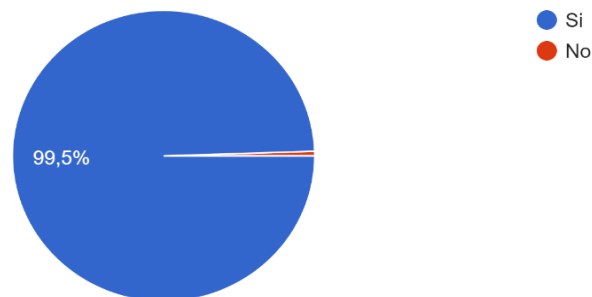


Figura 15. Uso de la visión artificial

Análisis e interpretación

Actualmente, la visión artificial enfocada en la agricultura es un tema de gran relevancia, por lo que ayuda a los agricultores a obtener un producto de calidad y a ser seguros en la intervención de enfermedades. Como se evidencia en la figura 15, un 99.5%, de los participantes en la encuesta están de acuerdo en que el uso de la visión artificial en la agricultura para obtener calidad en cada uno de los productos que ofrecen, específicamente en arveja, mientras que solo el 0.5% señala lo contrario.

4.1.2. Propuesta

La propuesta se originó a partir de conversaciones directas con los agricultores de la comunidad La Delicia en el Cantón Montúfar de la ciudad de San Gabriel. La investigación se enfocó en la detección de patologías que afectan los cultivos de guisantes, destacando aquellas que tienen un impacto significativo. Según los resultados arrojados de las encuestas, las tres principales enfermedades que afectan estos cultivos son: antracnosis (*Colletotrichum* spp.), fusarium (*Oxysporum* f. sp. Pisi) y oídio (*Erysiphe polygoni*). Estas enfermedades generan inconvenientes para los agricultores, incluyendo la pérdida total de sus cultivos.

Dado que actualmente el sector agrícola carece de herramientas tecnológicas para la detección temprana de estas enfermedades, se ha concluido que es viable desarrollar un prototipo capaz de identificar estas enfermedades en sus etapas iniciales en los cultivos de arveja. Además, se optó por la Programación Extrema (XP)

para realizar el desarrollo del prototipo, con el objetivo de gestionar el proyecto de manera efectiva, llevar a cabo el desarrollo en etapas iterativas y definir roles específicos dentro del equipo de investigación.

4.1.2.1. Metodología XP

4.1.2.1.1. Fase de planificación

Durante esta etapa, se definen roles de los desarrolladores en el proyecto, así como los plazos asignados para cada uno de los instrumentos de la metodología.

a) Roles

Tabla 18. Roles

Nombre	Descripción	Rol XP
MSc. Carlitos Guano	Tutor	Consultor
Kimberlhy Pérez	Investigador	Programador
Wilmer Tutachá	Investigador	Programador
Sr. José Tutachá	Agricultor	Consultor

b) Estimación de tiempo

La valoración del tiempo implica medir la importancia de las horas y los días durante la semana de trabajo. Esta evaluación será fundamental para establecer los parámetros en las etapas siguientes del proceso metodológico.

Tabla 19. Tiempo

Semanas	Días	Horas
0,2	1	7
0,4	2	14
1	5	35
1,6	8	56
2	10	70
2,4	12	84
2,6	13	91
3	15	105
4	20	140

c) Módulos

Tabla 20. Módulos de la aplicación

Módulo	Contenido
1.- Inicio de sesión	Registro con datos de usuario Registro mediante Google directamente Restablecimiento de contraseña
2.- Modelo de visión artificial	Preparación del modelo Experimentos de modelo

Módulo	Contenido
	Ejecución de modelo
3.- API REST	Vínculo con el modelo de visión artificial. Captura de datos Captura de imágenes
4.- Consulta de datos	Aceptación de datos API REST Exposición de resultados

Como se observa en la tabla 20, se da a conocer el número de módulos existentes además del contenido de cada uno de estos.

d) Historias de usuario

Módulo 1: Inicio de sesión

Tabla 21. H_usuario 1

Historia de usuario	
Nº: 1	Usuario: administrador
Nombre: Registro con datos de usuario	
Prioridad: Alta	Riesgo en desarrollo: Alto
Tiempo: 1 semanas	Iteración: 1
Encargado: Wilmer Tutachá	
Detalle: El administrador deberá acceder a la aplicación ingresando sus datos personales para iniciar sesión.	

Tabla 22. H_usuario 2

Historia de usuario	
Nº: 2	Usuario: administrador
Nombre: Registro mediante Google directamente	
Prioridad: Alta	Riesgo en desarrollo: Alto
Tiempo: 2 semanas	Iteración: 1
Encargado: Wilmer Tutachá	
Detalle: El administrador deberá registrarse directamente mediante su cuenta de Google para ingresar a la app.	

Tabla 23. H_usuario 3

Historia de usuario	
Nº: 3	Usuario: administrador
Nombre: Restablecimiento de contraseña	
Prioridad: Alta	Riesgo en desarrollo: Alto
Tiempo: 2 semanas	Iteración: 1
Encargado: Wilmer Tutachá	
Detalle: En caso de olvidar la contraseña, podrá restablecerla utilizando dirección de correo electrónico que está registrada y enlazada con su cuenta.	

Módulo 2. Modelo de visión artificial

Tabla 24. H_usuario 4

Historia de usuario	
Nº: 4	Usuario: administrador
Nombre: Entrenamiento del modelo de visión artificial.	
Prioridad: Alta	Riesgo en desarrollo: Alto

Tiempo: 4 semanas	Iteración: 1
Encargado: Wilmer Tutachá	
Detalle: El administrador tendrá el conocimiento para realizar actualizaciones y realizar el entrenamiento en el modelo de visión artificial de acuerdo a su necesidad.	

Tabla 25. H_usuario 5

Historia de usuario	
Nº: 5	Usuario: Administrador
Nombre: Pruebas de modelo	
Prioridad: Alta	Riesgo en desarrollo: Bajo
Tiempo: 4 semanas	Iteración: 2
Encargado: Wilmer Tutachá	
Detalle: El administrador podrá realizar pruebas detalladas de modelo de visión artificial evaluando así su funcionalidad.	

Tabla 26. H_usuario 6

Historia de usuario	
Nº: 6	Usuario: Administrador
Nombre: Implementación del modelo	
Prioridad: Alta	Riesgo en desarrollo: Alto
Tiempo: 4 semanas	Iteración: 1
Encargado: Kimberlhy Pérez, Wilmer Tutachá	
Detalle: El administrador logrará vincular el modelo de visión artificial recién creado y, al mismo tiempo, integrarlo plenamente en la aplicación.	

Modelo 3. API REST

Tabla 27. H_usuario 7

Historia de usuario	
Nº: 7	Usuario: Administrador
Nombre: Conexión del modelo de visión artificial a través de la API.	
Prioridad: Alta	Riesgo en desarrollo: alto
Tiempo: 6 semanas	Iteración: 1
Encargado: Kimberlhy Pérez, Wilmer Tutachá	
Detalle: Conexión de datos a través de la API REST	

Tabla 28. H_usuario 8

Historia de usuario	
Nº: 8	Usuario: Administrador
Nombre: Captura de imágenes	
Prioridad: Alta	Riesgo en desarrollo: Bajo
Tiempo: 3 semanas	Iteración: 1
Encargado: Kimberlhy Pérez, Wilmer Tutachá	
Detalle: Tomar fotografías en el momento o elegir las desde la galería, todo directamente desde la aplicación móvil con la finalidad de hacer el análisis mediante el modelo de visión artificial.	

Tabla 29. H_usuario 9

Historia de usuario	
Nº: 9	Usuario: Administrador
Nombre: Envío de imágenes	
Prioridad: Alta	Riesgo en desarrollo: Medio
Tiempo: 2 semana	Iteración: 2
Encargado: Wilmer Tutachá	
Detalle: Enviar imágenes para su análisis mediante el modelo de visión artificial.	

Modulo 4. Consulta de datos

Tabla 30. H_usuario 10

Historia de usuario	
Nº: 10	Usuario: Agricultor
Nombre: Recepción de datos	
Prioridad: Alta	Riesgo en desarrollo: Alto
Tiempo: 1 semana	Iteración: 2
Encargado: Wilmer Tutachá	
Detalle: El modelo cumple la función de recibir la fotografía y la procesa para su análisis.	

Tabla 31. H_usuario 11

Historia de usuario	
Nº: 11	Usuario: Agricultor
Nombre: Presentación de resultados	
Prioridad: Alta	Riesgo en desarrollo: Alto
Tiempo: 2 semana	Iteración: 2
Encargado: Wilmer Tutachá	
Detalle: El modelo envía los resultados y muestra la predicción de la enfermedad basada en la imagen.	

e) Tareas de usuario

Tabla 32. T_usuario 1

Tarea de usuario	
Número de tarea: 1	Número de historia: 1
Nombre: Información sobre Firebase	
Tipo: Investigación	
Inicio: 05/02/2023	Fin: 15/02/2023
Encargado: Kimberlhy Pérez, Wilmer Tutachá	
Detalle: Análisis del funcionamiento con aplicaciones móviles en relación a Firebase.	

Tabla 33. T_usuario 2

Tarea de usuario	
Número de tarea: 2	Número de historia: 2
Nombre: Conexión con la plataforma Firebase	
Tipo: Investigación	
Inicio: 17/02/2023	Fin: 04/03/2023
Encargado: Kimberlhy Pérez, Wilmer Tutachá	
Detalle: Implementación de código en Visual Studio Code.	

Tabla 34. T_usuario 3

Tarea de usuario	
Número de tarea: 3	Número de historia: 3
Nombre: Autenticación de login	
Tipo: Investigación	
Inicio: 06/03/2023	Fin: 29/03/2023
Encargado: Kimberlhy Pérez, Wilmer Tutachá	
Detalle: Utiliza cuenta de Google como método de inicio de sesión.	

Tabla 35. T_usuario 4

Tarea de usuario	
Número de tarea: 4	Número de historia: 4
Nombre: Selección de algoritmos de visión artificial	
Tipo: Investigación	
Inicio: 02/04/2023	Fin: 12/04/2023
Encargado: Kimberlhy Pérez, Wilmer Tutachá	
Detalle: Selección de algoritmos de visión artificial enfocados a la detección de imágenes.	

Tabla 36. T_usuario 5

Tarea de usuario	
Número de tarea: 5	Número de historia: 5
Nombre: Pruebas de algoritmos de visión artificial	
Tipo: Investigación	
Inicio: 18/04/2023	Fin: 30/04/2023
Encargado: kimberlhy Pérez, Wilmer Tutachá	
Detalle: Pruebas de algoritmos para implementar modelos de visión artificial; Yolo v8, MobileNetV2.	

Tabla 37. T_usuario 6

Tarea de usuario	
Número de tarea: 6	Número de historia: 6
Nombre: Análisis de algoritmos de visión artificial	
Tipo: Desarrollo	
Inicio: 12/05/2023	Fin: 19/05/2023
Encargado: Kimberlhy Pérez, Wilmer Tutachá	
Detalle: Se realizará un análisis de cada algoritmo, considerando aspectos como el tiempo de ejecución, el tiempo de entrenamiento y la eficiencia de estos.	

Tabla 38. T_usuario 7

Tarea de usuario	
Número de tarea: 7	Número de historia: 6
Nombre: Implementación del modelo de visión artificial	
Tipo: Desarrollo	
Inicio: 27/05/2023	Fin: 10/06/2023
Encargado: Kimberlhy Pérez, Wilmer Tutachá	
Detalle: Elaboración y aplicación del modelo de visión artificial en el prototipo, que incluye la capacidad de enviar y recibir datos, para realizar las debidas detecciones.	

Tabla 39. T_usuario 8

Tarea de usuario	
Número de tarea: 8	Número de historia: 7
Nombre: Diseño API REST	
Tipo: Desarrollo	
Inicio: 15/06/2023	Fin: 29/06/2023
Encargado: Kimberlhy Pérez, Wilmer Tutachá	
Detalle: Elaboración de una API REST diseñada para recibir imágenes destinadas a análisis, además de facilitar el envío de imágenes y las correspondientes predicciones.	

Tabla 40. T_usuario 9

Tarea de usuario	
Número de tarea: 9	Número de historia: 8
Nombre: Diseño de prototipado	
Tipo: Diseño	
Inicio: 12/07/2023	Fin: 14/08/2023
Encargado: Wilmer Tutachá	
Detalle: Elaboración de prototipos de interfaces para la app.	

Tabla 41. T_usuario 10

Tarea de usuario	
Número de tarea: 10	Número de historia: 8
Nombre: Desarrollo de interfaz de aplicación móvil	
Tipo: Desarrollo	
Inicio: 15/08/2023	Fin: 02/09/2023
Encargado: Wilmer Tutachá	
Detalle: El diseño y desarrollo de las distintas interfaces y menús de la aplicación destinada a la detección de patologías.	

Tabla 42. T_usuario 11

Tarea de usuario	
Número de tarea: 11	Número de historia: 8
Nombre: Funcionalidades de la aplicación móvil	
Tipo: Desarrollo	
Inicio: 03/09/2023	Fin: 11/10/2023
Encargado: Kimberlhy Pérez, Wilmer Tutachá	
Detalle: Implementación de funcionalidades para captura de imágenes del dispositivo móvil.	

Tabla 43. T_usuario 12

Tarea de usuario	
Número de tarea: 12	Número de historia: 9
Nombre: Envío de imágenes para el análisis de la aplicación móvil con API	
Tipo: Desarrollo	
Inicio: 12/10/2023	Fin: 13/11/2023
Encargado: Kimberlhy Pérez, Wilmer Tutachá	
Detalle: Desarrollo de un script con la capacidad de enviar imágenes de manera eficiente a la aplicación móvil y la API.	

Tabla 44. T_usuario 13

Tarea de usuario	
Número de tarea: 13	Número de historia: 10
Nombre: Recepción de imágenes y generación de predicciones en la aplicación móvil	
Tipo: Desarrollo	
Inicio: 16/11/2023	Fin: 04/12/2023
Encargado: Kimberlhy Pérez, Wilmer Tutachá	
Detalle: Desarrollo de un script para recibir imágenes y realizar las predicciones correspondientes.	

Tabla 45. T_usuario 14

Tarea de usuario	
Número de tarea: 14	Número de historia: 11
Nombre: Presentación de los resultados que incluyen las predicciones generadas en las imágenes	
Tipo: Desarrollo	
Inicio: 08/01/2024	Fin: 12/02/2024
Encargado: Kimberlhy Pérez, Wilmer Tutachá	
Detalle: Desarrollo de un script que permita realizar predicciones basadas en la imagen visualizada.	

f) Estimación de tareas de usuario

Tabla 46. Estimación de tiempo en tareas de usuario

Nombre historia	Nº, Tarea	Tarea	Semana	Días	Horas	Porcentaje de semanas
Inicio de sesión	1	Registro con datos de usuario	1	5	35	4,35
	2	Registro mediante Google directamente	2	10	70	8,70
	3	Restablecimiento de contraseña	2	10	70	8,70
Entrenamiento del modelo de visión artificial	4	Selección de algoritmos de visión artificial.	1	5	35	4,35
	5	Pruebas de Algoritmos de visión artificial.	2	10	70	8,70
Pruebas del modelo	6	Elección de algoritmo de visión artificial	2	10	70	8,70

Nombre historia	Nº, Tarea	Tarea	Semana	Días	Horas	Porcentaje de semanas
Implementación del modelo	7	Implementación de modelo de visión artificial.	2	10	70	8,70
Conexión con el modelo de visión artificial.	8	Diseño API REST	3	15	105	13,04
Captura de imágenes	9	Diseño de prototipado	2	10	70	8,70
	10	Desarrollo de interfaz de aplicación móvil	1	5	35	4,35
Envió de imágenes	11	Funcionalidades de aplicación móvil	1	5	35	4,35
	12	Envió de imágenes para el análisis de la aplicación móvil	2	10	70	8,70
Recepción de datos API	13	Recepción de imágenes y generación de predicciones en la aplicación móvil	1	5	35	4,35
Presentación de resultados	14	Presentación de resultados con imagen y predicción en aplicación móvil	1	5	35	4,35
Total, de tiempo estimado			23	115	805	100

g) Plan de entrega proyecto

La tabla 47, describe el calendario estimado con cada uno de los pasos esenciales para progresar en el proyecto y proporciona una estimación del tiempo requerido para cada fase, basada en las historias narradas en etapas anteriores.

Tabla 47. Plan de entrega del proyecto

Módulo	Nº.	Nombre de historia	Semanas	Días	Horas
Inicio de sesión	1	Registro con datos de usuario	1	5	35
	2	Registro mediante Google directamente	2	10	70
	3	Restablecimiento de contraseña	2	10	70
Modelo de visión artificial	4	Entrenamiento del modelo de visión artificial.	4	20	140
	5	Pruebas de modelo.	4	20	140
	6	Implementación del modelo.	4	20	140
Api Rest	7	Enlace con el modelo de visión artificial.	6	30	210
Captura de datos	8	Captura de imágenes.	3	15	105
	9	Envió de imágenes.	2	10	70
Consulta de datos	10	Recepción de datos API	1	5	35
	11	Presentación de resultados.	2	10	70

4.1.2.1.2. Fase de diseño

El prototipo emplea la arquitectura cliente-servidor con el objetivo de lograr resultados óptimos y se presenta de la siguiente manera.

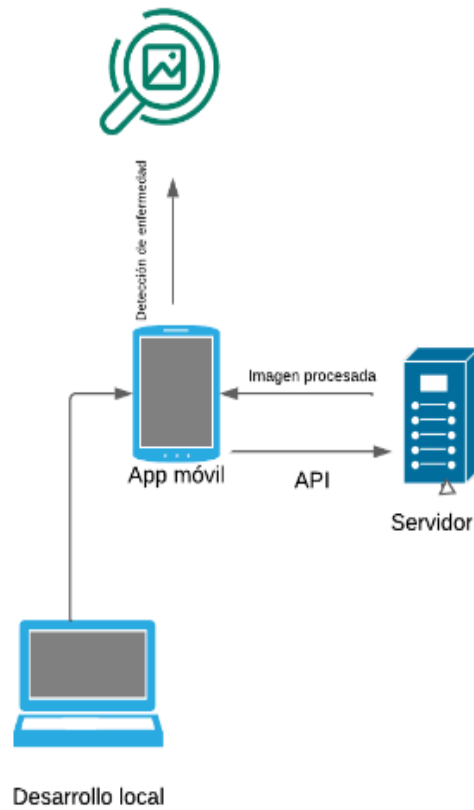


Figura 16. Arquitectura cliente - servidor

a) Prototipo de interfaces

La herramienta que se utilizó para el perfeccionamiento del prototipo de la app se denomina Marvel que permite la creación rápida de prototipos además proporciona un entorno colaborativo por lo que es una buena alternativa en este proyecto, de tal manera se muestra las diferentes escenas del aplicativo, gracias a esta herramienta se detalla cada interfaz del aplicativo las cuales se muestran a continuación:

Interfaz 1: Login

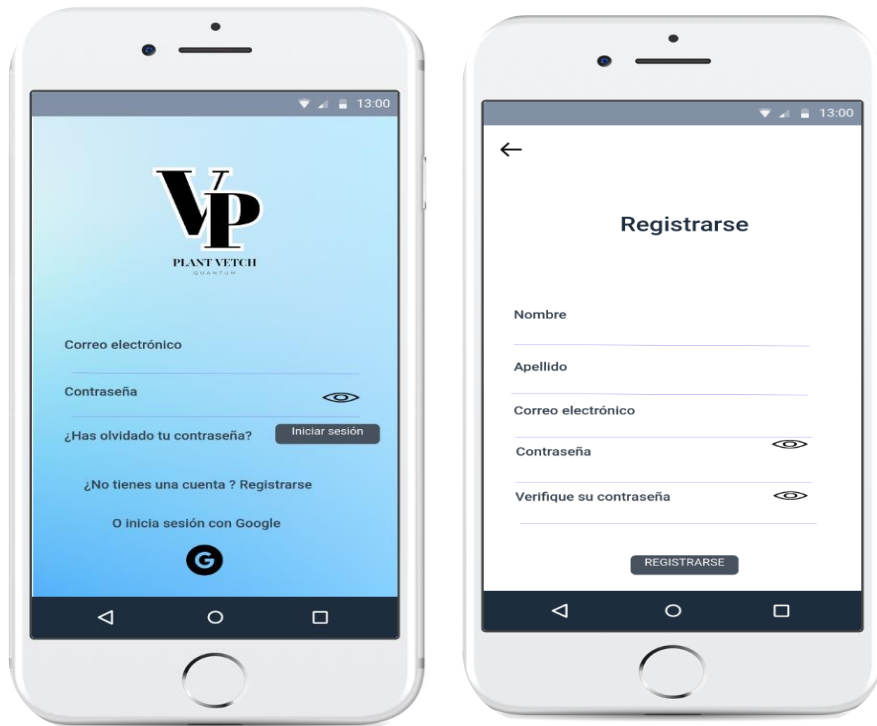


Figura 17. Prototipo login

Interfaz 2: Menú de inicio



Figura 18. Prototipo menú inicio

Interfaz 3: Manejo Agrónomo



Figura 19. Prototipo Manejo agrónómico

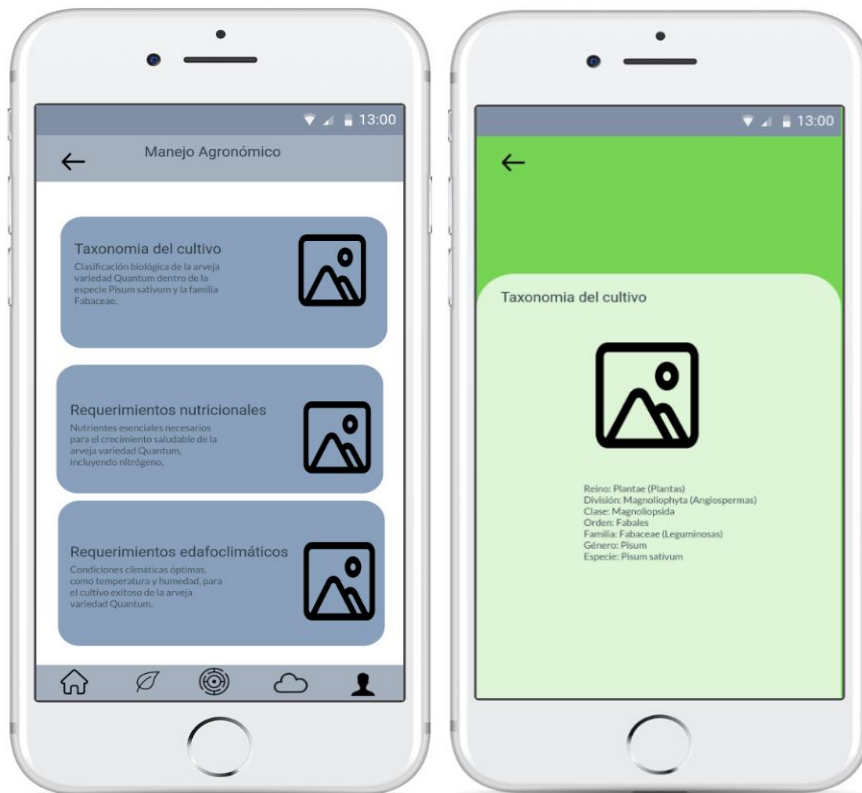


Figura 20. Prototipo Taxonomía del cultivo

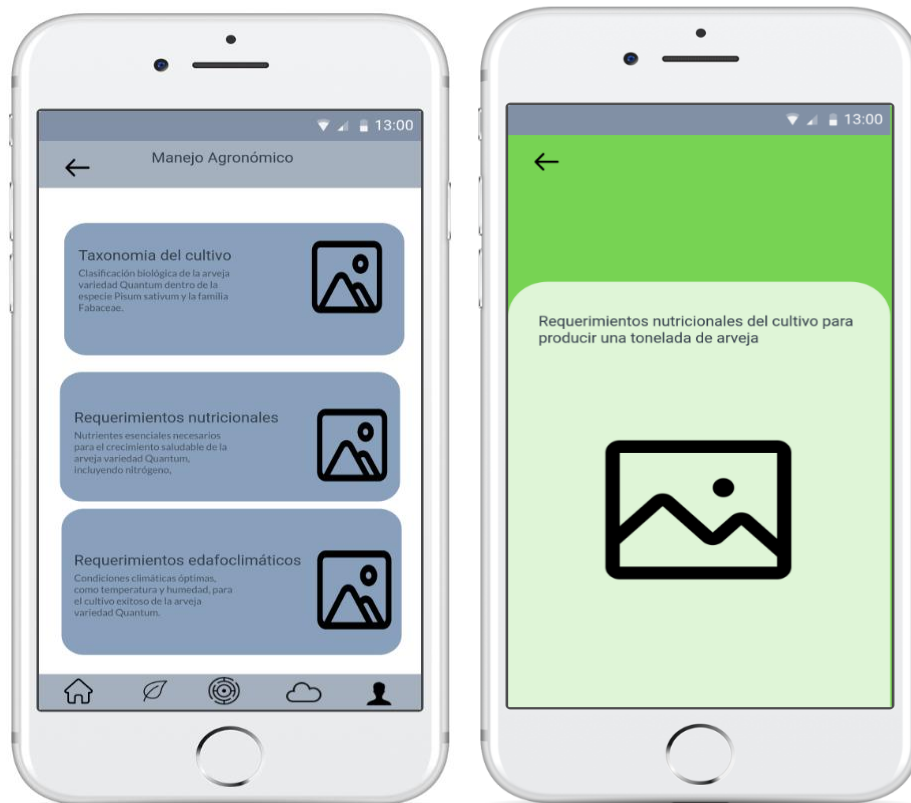


Figura 21. Prototipo requerimientos nutricionales

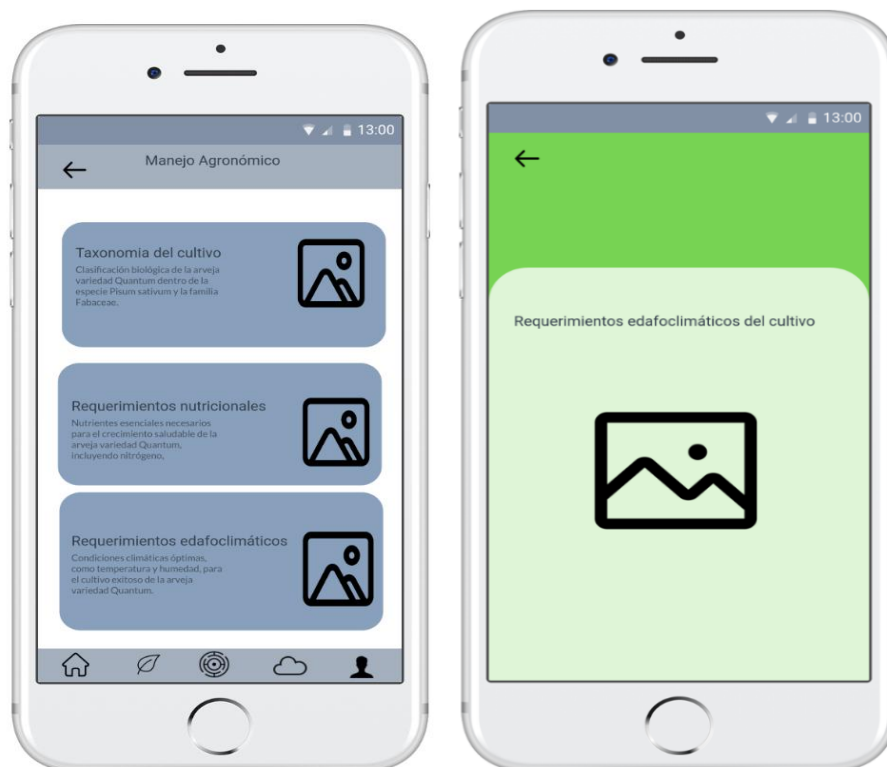


Figura 22. Prototipo requerimientos edafoclimáticos



Figura 23. Prototipo plagas

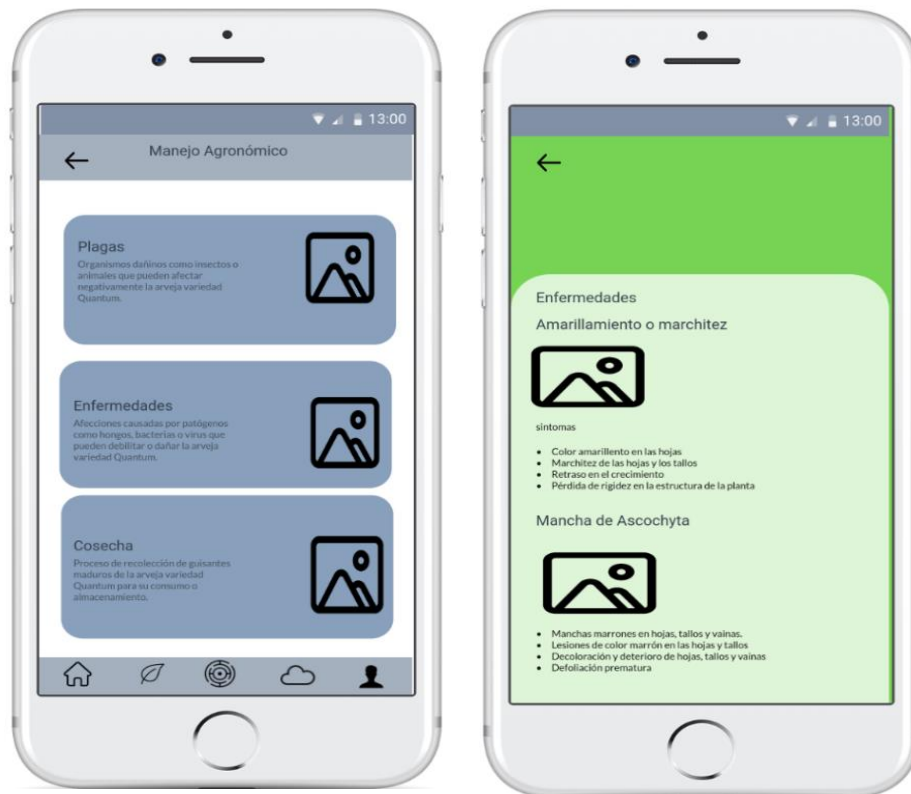


Figura 24. Prototipo de enfermedades

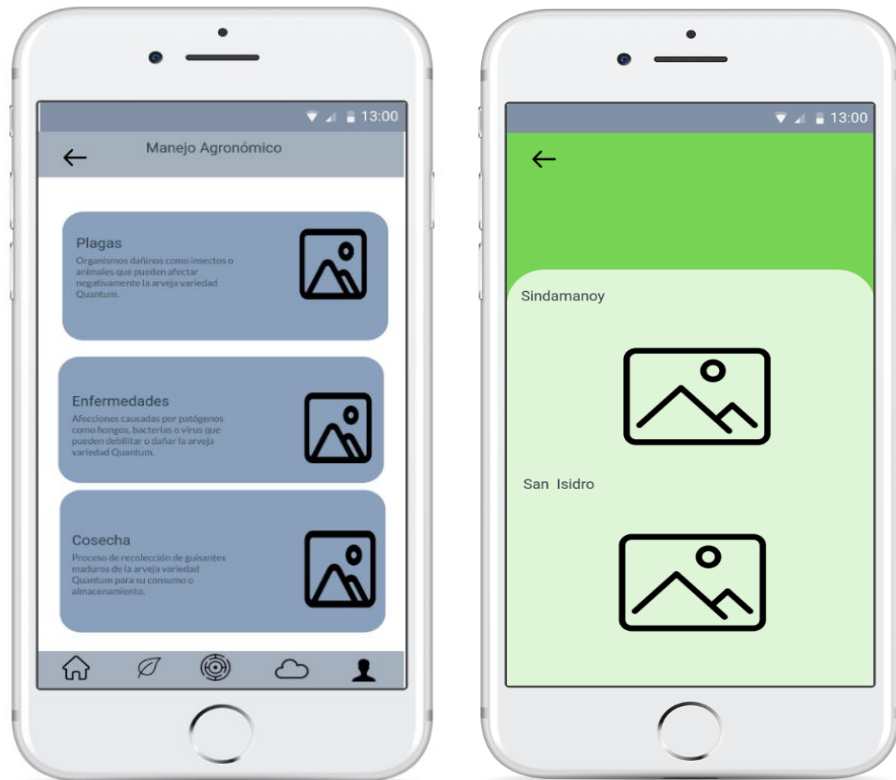


Figura 25. Prototipo cosecha

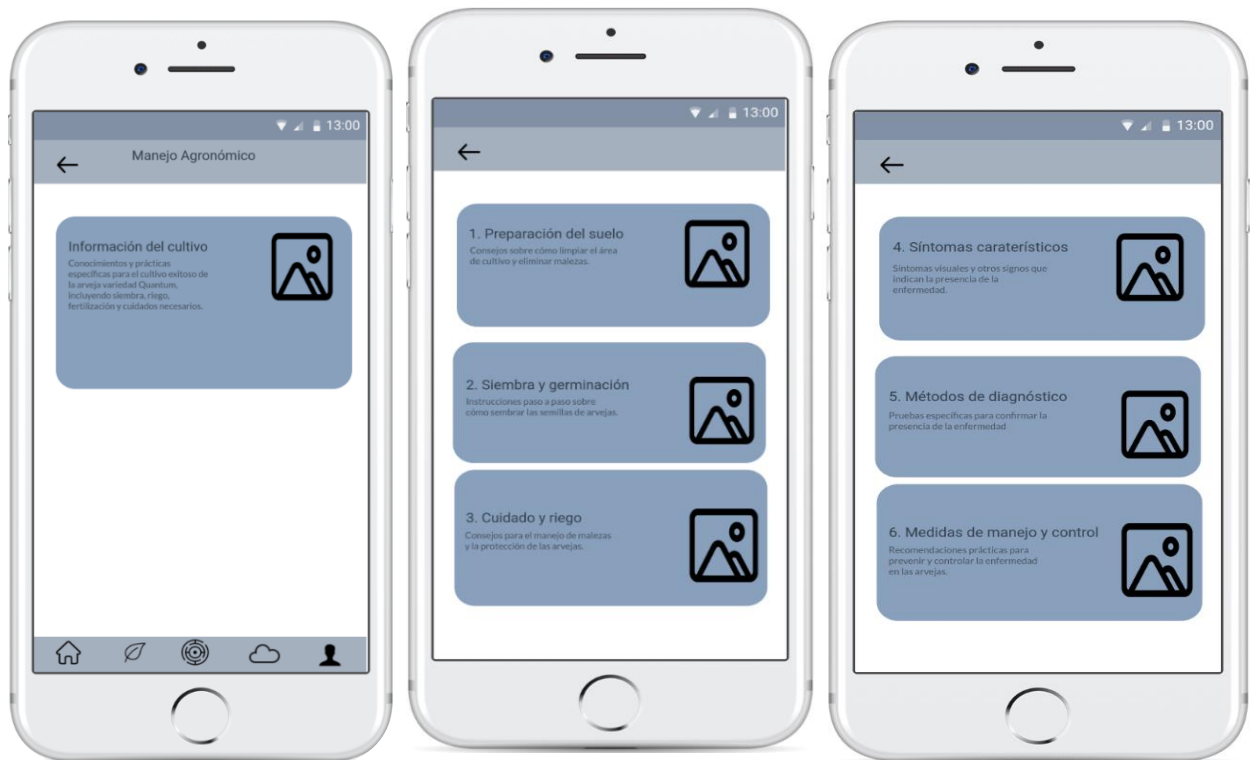




Figura 26. Prototipo información del cultivo



Figura 27. Prototipo preparación suelo



Figura 28. Prototipo siembra y germinación



Figura 29. Prototipo cuidado y riego



Figura 30. Prototipo síntomas característicos



Figura 31. Prototipo métodos de diagnóstico



Figura 32. Prototipo medidas de manejo y control

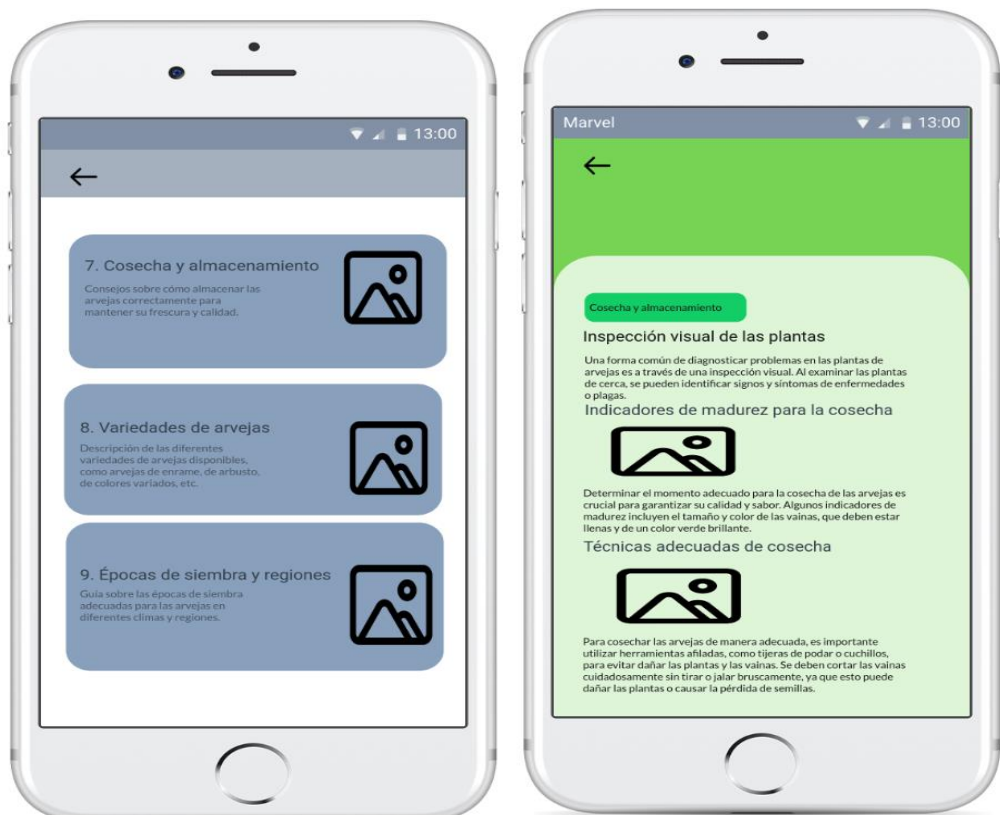


Figura 33. Prototipo cosecha y almacenamiento

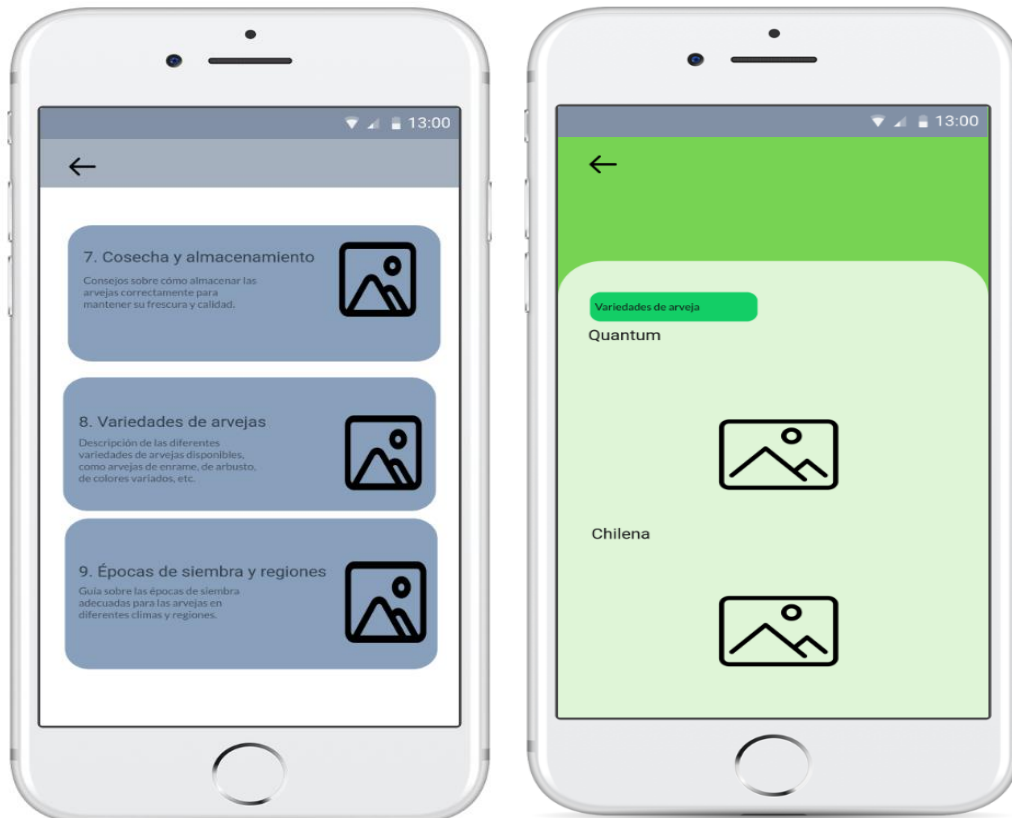


Figura 34. Prototipo variedad de arvejas



Figura 35. Prototipo épocas de siembra y regiones



Figura 36. Prototipo buenas prácticas de cultivo

Interfaz 4: Detección de enfermedades

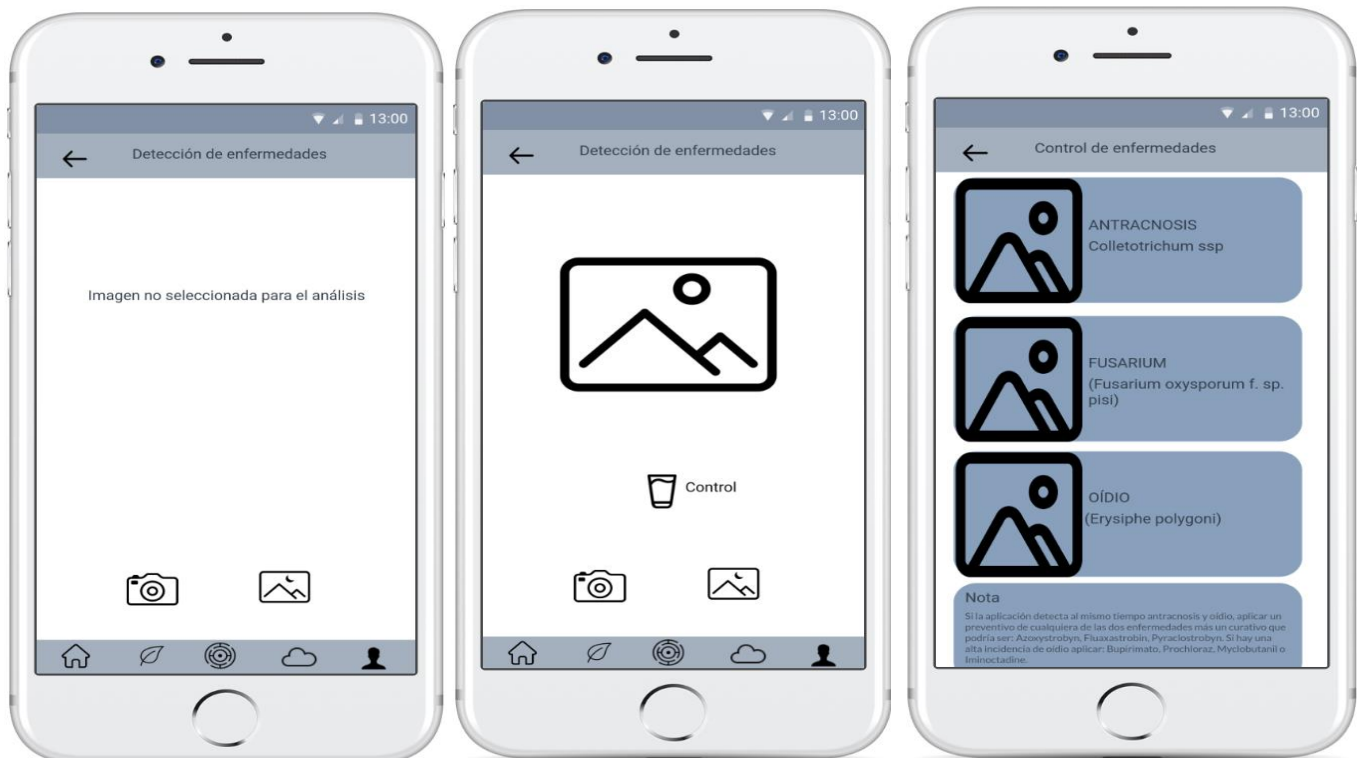


Figura 37. Prototipo detección enfermedades

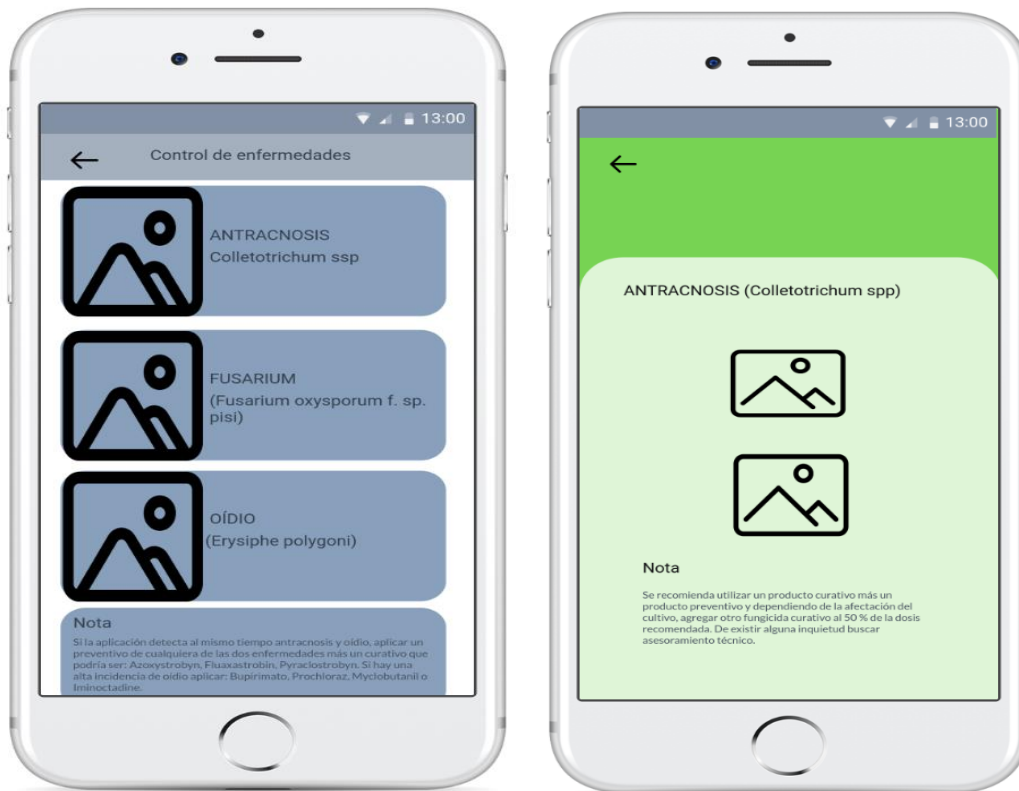


Figura 38. Prototipo control antracnosis

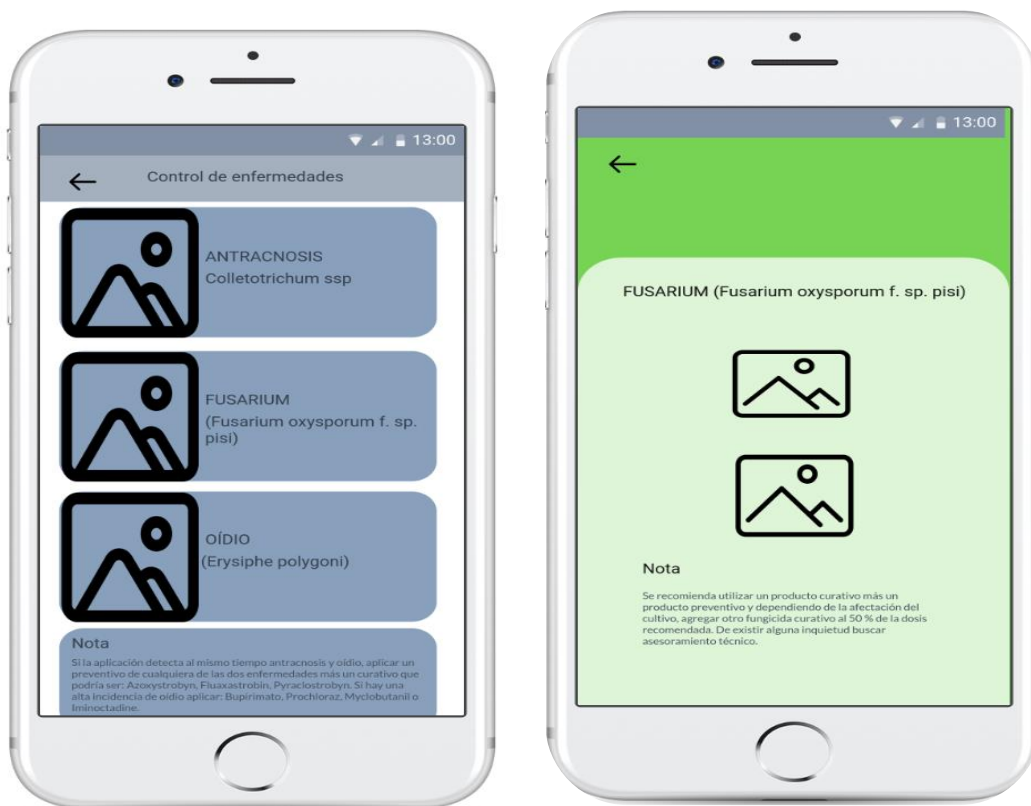


Figura 39. Prototipo control fusarium

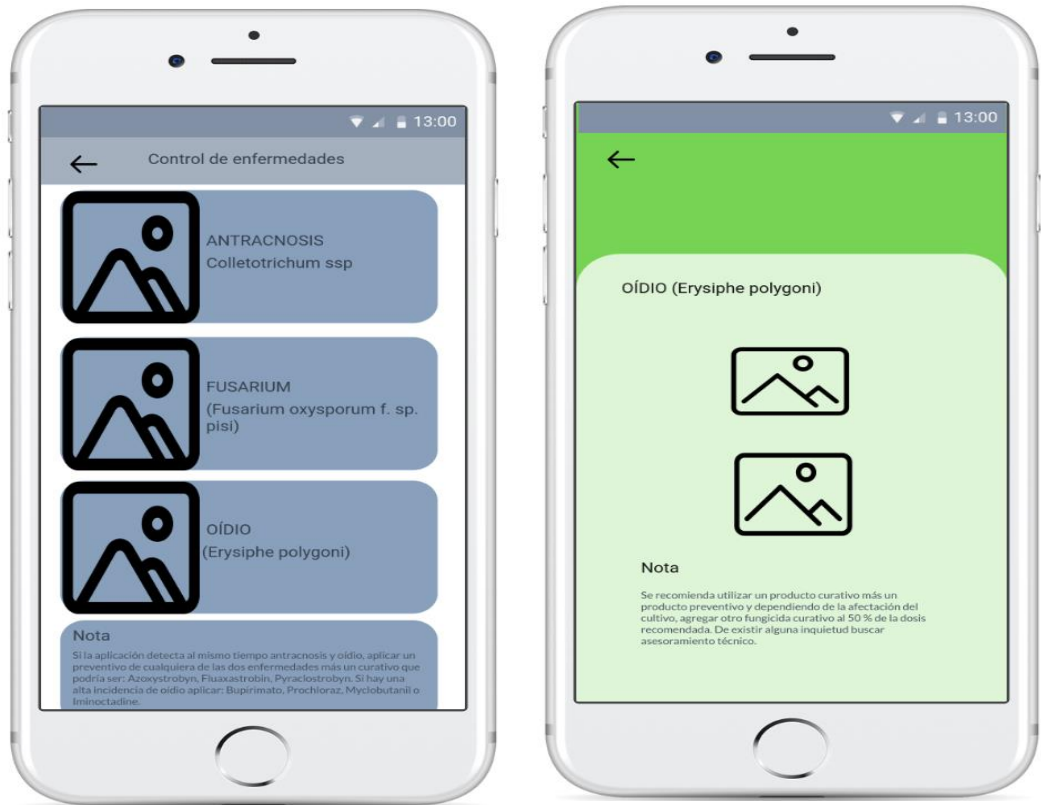


Figura 40. Prototipo control óidio

Interfaz 5: Clima



Figura 41. Prototipo clima

Interfaz 6: Perfil

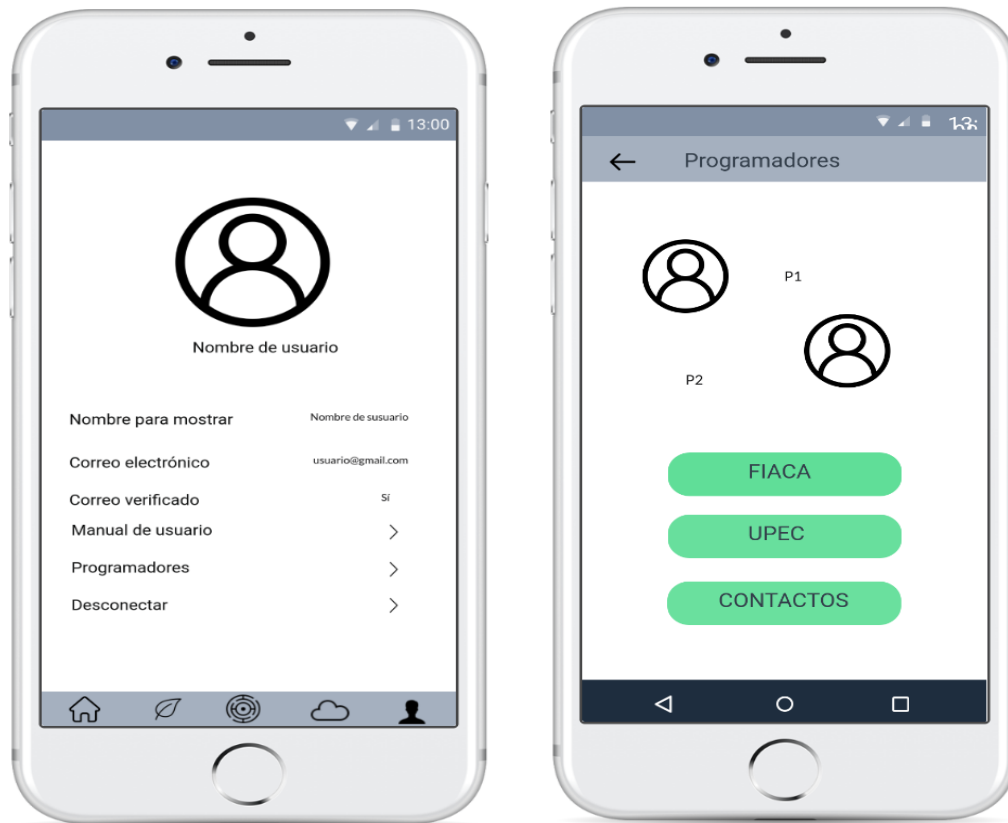


Figura 42. Prototipo perfil de usuario

b) Diagrama de caso de uso

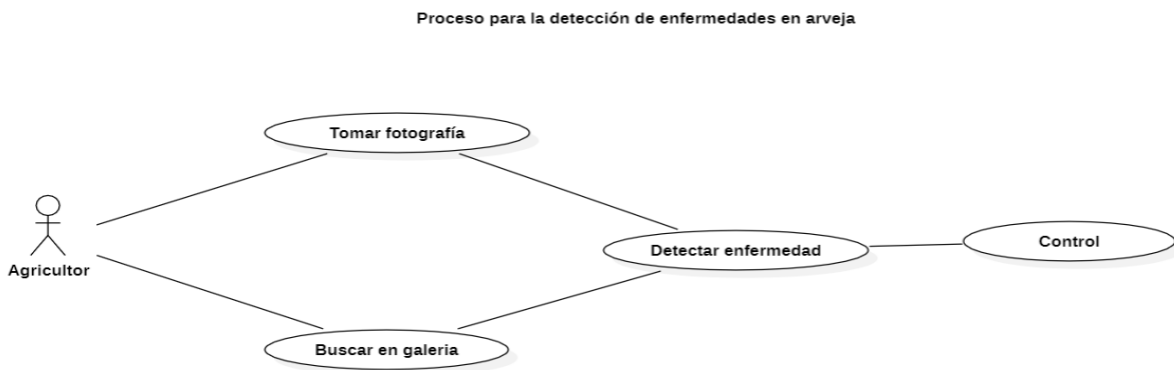


Figura 43. Caso de uso del agricultor

4.1.2.1.3. Fase de codificación

a) Diseño del entorno de desarrollo

Con el fin de mejorar el desarrollo del software, se presenta la figura 44, por lo que se optó por emplear esta herramienta de desarrollo que en este caso es Android Studio,

brinda la capacidad de crear aplicaciones específicas para Android, su eficaz sistema de depuración y la disponibilidad de emuladores para realizar las pruebas en diversos dispositivos lo convierten en la elección ideal para este proyecto de investigación.

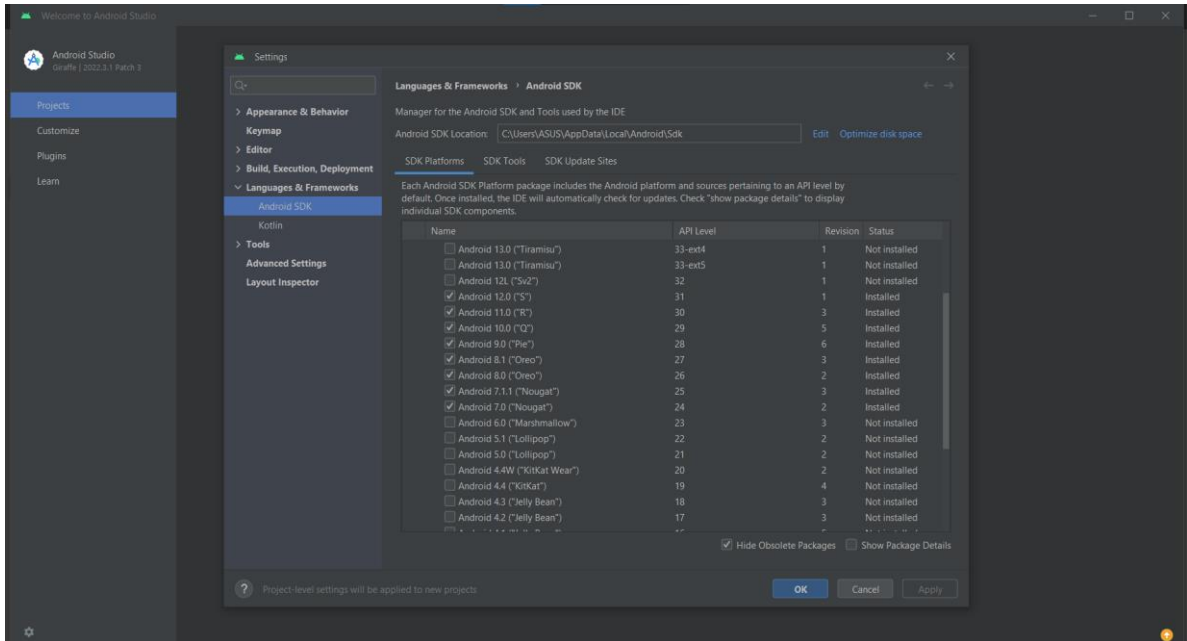


Figura 44. Herramienta SDK

La figura 45, muestra la autenticación de los usuarios mediante Google en la plataforma Firebase la cual permite el registro de los usuarios directamente vía email o haciendo uso del campo de registro.

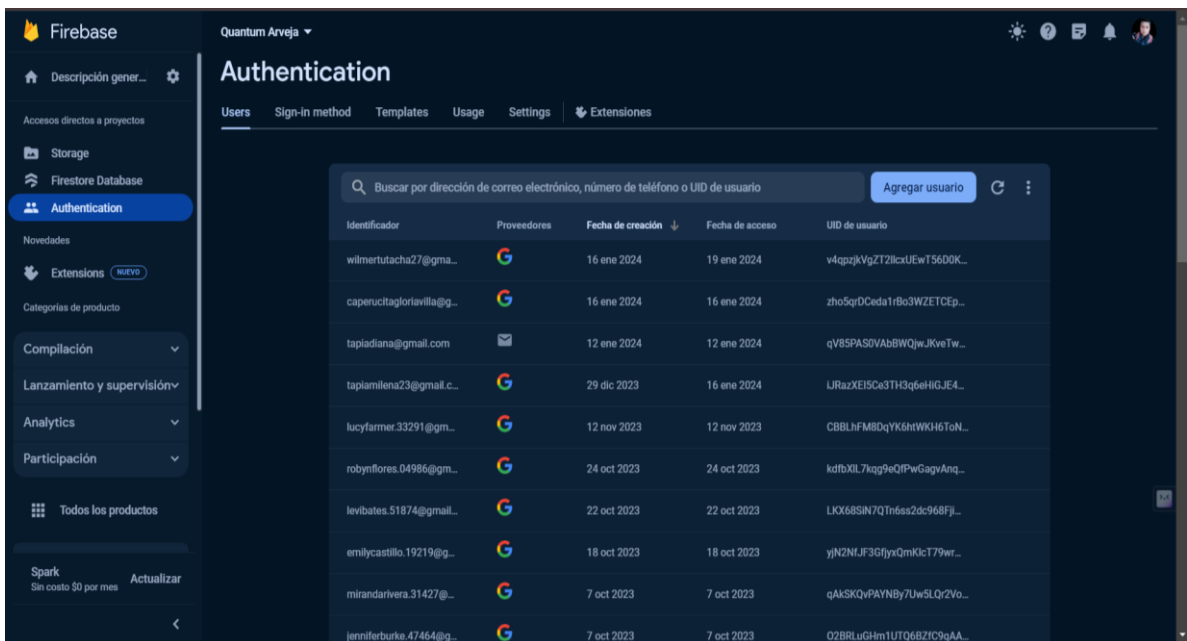


Figura 45. Autenticación de usuarios a través de Firebase

Para efectos de la investigación se realizó dos modelos enfocados a la problemática existente en la comunidad de la Delicia, a continuación, se presentan los mismos.

b) MobileNetV2

El prototipo hace el requerimiento del algoritmo MobileNetV2 para ejecutar la identificación temprana de enfermedades de arveja por ende a continuación se presentan algunos beneficios:

Según Martínez (2020), manifiesta que MobileNetV2 integra una serie de atributos fundamentales que contribuyen a su eficiencia y rendimiento en la tarea de clasificación de imágenes. Estas particularidades abarcan desde la utilización de convoluciones separables en profundidad hasta la implementación de residuos invertidos, el diseño de cuellos de botella y cuellos de botella lineales, así como la inclusión de bloques de compresión y excitación (SE). Cada una de estas características desempeña una función crucial en la simplificación de la carga computacional del modelo, sin comprometer su precisión. Brinda las siguientes ventajas:

- Gracias a su diseño ligero, MobileNetV2 puede implementarse de manera eficiente en dispositivos móviles y sistemas integrados que tienen recursos de computación limitados, permitiendo así un uso efectivo en tales entornos.
- MobileNetV2 posee una precisión alta que se encuentra a la altura de modelos de gran tamaño, es decir complejos y costosos desde perspectiva computacional
- El tamaño reducido del modelo admite lapsos de deducción más rápidos, lo que permite ejecutar apps en tiempo real.

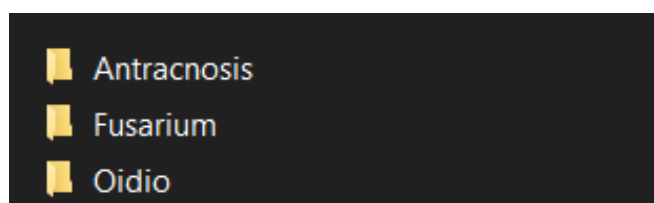


Figura 46. Dataset de imágenes

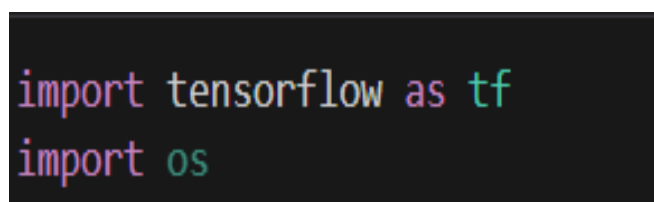


Figura 47. Instalación de la librería de TensorFlow

```
base_dir = './dataset'
```

Figura 48. Asignamos la dirección del directorio de la dataset

En la figura 49, se presenta la cantidad de imágenes que se van a entrenar y validar con el modelo.

```
IMAGE_SIZE = 224
BATCH_SIZE = 1320

datagen = tf.keras.preprocessing.image.ImageDataGenerator(
    rescale=1./255,
    validation_split=0.2
)

train_generator = datagen.flow_from_directory(
    base_dir,
    target_size=(IMAGE_SIZE, IMAGE_SIZE),
    batch_size=BATCH_SIZE,
    subset='training'
)

val_generator = datagen.flow_from_directory(
    base_dir,
    target_size=(IMAGE_SIZE, IMAGE_SIZE),
    batch_size=BATCH_SIZE,
    subset='validation'
)
```

```
... Found 1057 images belonging to 3 classes.
     Found 263 images belonging to 3 classes.
```

Figura 49. Cantidad de imágenes

Se genera el archivo de texto con las respectivas etiquetas de cada patología.

```
print(train_generator.class_indices)
labels = '\n'.join(sorted(train_generator.class_indices.keys()))
with open('labels.txt', 'w') as f:
    f.write(labels)
```

```
{'antracnosis': 0, 'fusarium': 1, 'oidio': 2}
```

Figura 50. Etiquetas de cada patología.

```

base_model.trainable=False
model = tf.keras.Sequential([
    base_model,
    tf.keras.layers.Conv2D(32,3, activation = 'relu'),
    tf.keras.layers.Dropout(0.2),
    tf.keras.layers.GlobalAveragePooling2D(),
    tf.keras.layers.Dense(4, activation='softmax')
])

```

Figura 51. Modelo de red neuronal

```

model.compile(
    optimizer=tf.keras.optimizers.Adamax(),
    loss='categorical_crossentropy',
    metrics=['accuracy']
)

```

Figura 52. Proceso de entrenamiento del modelo

```

epochs = 100
history = model.fit(
    train_generator,
    epochs=epochs,
    validation_data=val_generator
)

```

Figura 53. Número de épocas

```

... Epoch 1/100
WARNING:tensorflow:From C:\Users\ASUS\AppData\Roaming\Python\Python310\site-packages\keras\src\utils\tf_utils.py:492: The name tf.ragged.RaggedTensorVa

WARNING:tensorflow:From C:\Users\ASUS\AppData\Roaming\Python\Python310\site-packages\keras\src\engine\base_layer_utils.py:384: The name tf.executing_ea

1/1 [=====] - 91s 91s/step - loss: 1.8357 - accuracy: 0.3117 - val_loss: 4.0513 - val_accuracy: 0.4286
Epoch 2/100
1/1 [=====] - 75s 75s/step - loss: 3.5539 - accuracy: 0.5501 - val_loss: 3.0275 - val_accuracy: 0.6593
Epoch 3/100
1/1 [=====] - 76s 76s/step - loss: 3.5179 - accuracy: 0.6179 - val_loss: 1.7822 - val_accuracy: 0.6593
Epoch 4/100
1/1 [=====] - 84s 84s/step - loss: 2.0385 - accuracy: 0.6585 - val_loss: 0.9459 - val_accuracy: 0.6374
Epoch 5/100
1/1 [=====] - 72s 72s/step - loss: 0.9794 - accuracy: 0.6667 - val_loss: 0.6569 - val_accuracy: 0.5824
Epoch 6/100
1/1 [=====] - 76s 76s/step - loss: 0.5836 - accuracy: 0.6612 - val_loss: 0.6328 - val_accuracy: 0.5714
Epoch 7/100
1/1 [=====] - 68s 68s/step - loss: 0.5366 - accuracy: 0.6558 - val_loss: 0.6821 - val_accuracy: 0.6484
Epoch 8/100
1/1 [=====] - 77s 77s/step - loss: 0.5813 - accuracy: 0.6802 - val_loss: 0.6740 - val_accuracy: 0.7253
Epoch 9/100
1/1 [=====] - 71s 71s/step - loss: 0.5980 - accuracy: 0.7453 - val_loss: 0.6161 - val_accuracy: 0.8022
Epoch 10/100
1/1 [=====] - 62s 62s/step - loss: 0.5866 - accuracy: 0.7913 - val_loss: 0.5543 - val_accuracy: 0.8462
Epoch 11/100
...
Epoch 99/100
1/1 [=====] - 89s 89s/step - loss: 0.0137 - accuracy: 1.0000 - val_loss: 0.1772 - val_accuracy: 0.9451
Epoch 100/100
1/1 [=====] - 60s 60s/step - loss: 0.0117 - accuracy: 1.0000 - val_loss: 0.1773 - val_accuracy: 0.9451

```

Figura 54. Progreso de entrenamiento

A continuación, en la figura 55 muestra el resultado del modelo de clasificación.

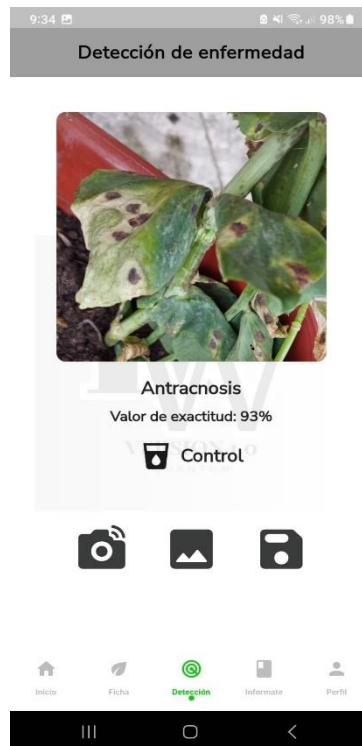


Figura 55. App de clasificación

Para la identificación de las patologías que se ocasionan en la arveja, se ha tomado en cuenta el modelo de clasificación movilnetv2, que permite clasificar imágenes de acuerdo con cada etiqueta asignada es decir al nombre de cada patología: antracnosis, fusarium, oídio. En ese aspecto se realizó el entrenamiento y análisis de este modelo en donde se utilizó una dataset de imágenes con su respectiva categoría y se procedió a realizar el entrenamiento, sin embargo con las diferentes pruebas que se ejecutó se tomó en cuenta que para este tipo de enfermedades se necesita más precisión en la detección, es decir que muestre la zona directamente afectada en la leguminosa, para ello se optó por elegir un modelo más preciso como es el caso de Yolo que permite la detección de objetos en imágenes en tiempo real e incluso utiliza bounding boxes es decir cuadros delimitadores en donde señala directamente las partes de la planta afectada ya sea hojas y fruto, cabe destacar que cada patología muestra características únicas. A continuación, se presenta el modelo de detección de objetos.

c) Yolov8

El prototipo hace el requerimiento del algoritmo yolov8 para ejecutar la identificación temprana de enfermedades de arveja por ende a continuación se presentan algunos beneficios:

Según Polo (2024), afirma que la combinación de velocidad y precisión de YOLOv8 lo convierte como la opción más idónea para diversas apps que necesitan procesamiento en tiempo real, como la identificación de objetos en videos en vivo o la supervisión de flujos de datos en tiempo real.

Adoptar YOLOv8 para su implementación brinda diversos beneficios:

- Precisión sobresaliente: YOLOv8 destaca por su excepcional nivel de precisión en la detección de objetos, siendo especialmente crucial en sectores como la medicina y la automoción, donde la seguridad ocupa un lugar prioritario.
- Impresionante velocidad: A pesar de su elevada precisión, YOLOv8 exhibe una notable rapidez. Puede procesar transmisiones de video en tiempo real, abriendo nuevas oportunidades de aplicación.
- Facilidad de uso: YOLOv8 ha sido diseñado para integrarse de manera sencilla en aplicaciones existentes. Respaldo por una comunidad activa, ofrece documentación detallada que facilita su implementación.
- Flexibilidad: YOLOv8 es altamente personalizable, permitiendo ajustar el modelo para detectar categorías específicas de objetos u optimizar su rendimiento según las necesidades particulares del usuario.

Etiquetado de Imágenes: Antracnosis, Fusarium y Oídio en arveja

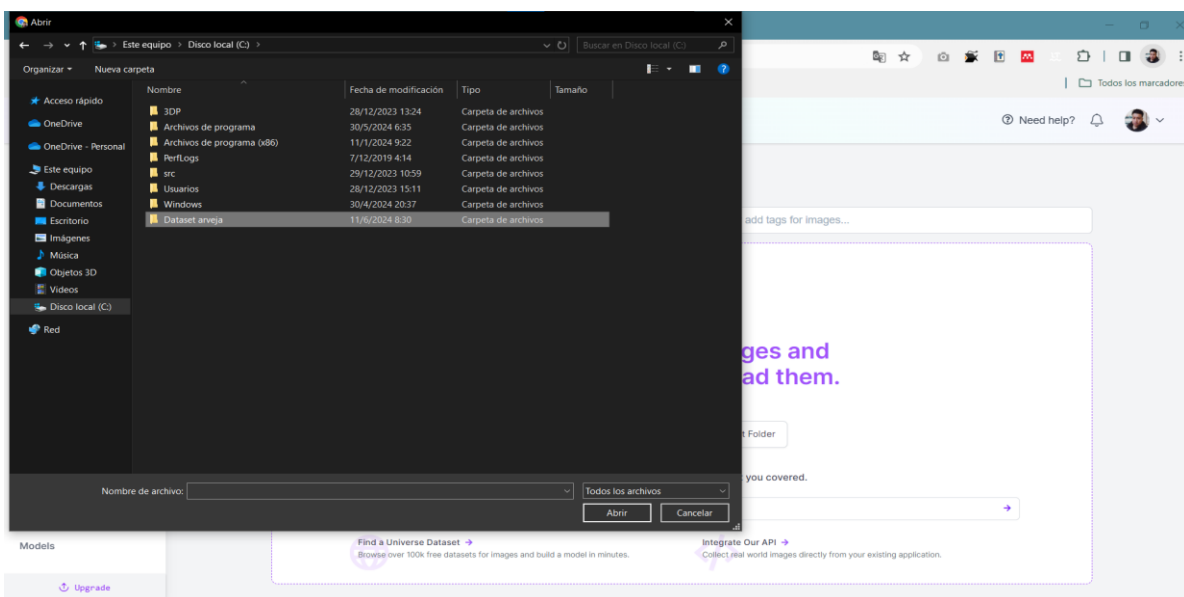


Figura 56. Dataset de imágenes

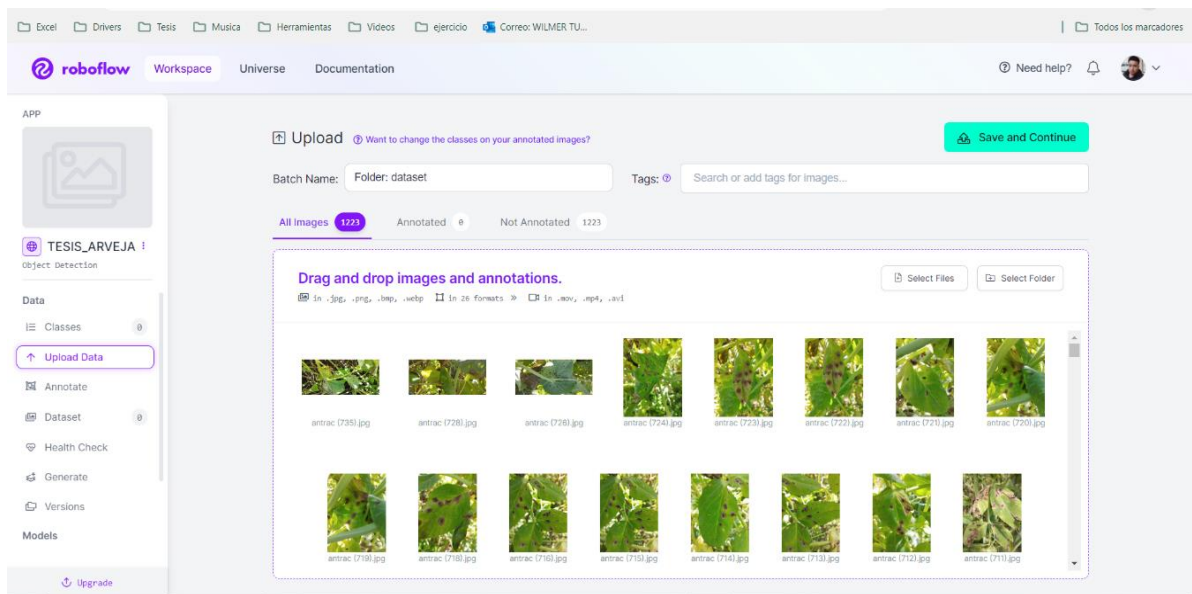


Figura 57. Preparación para etiquetar imágenes de enfermedades

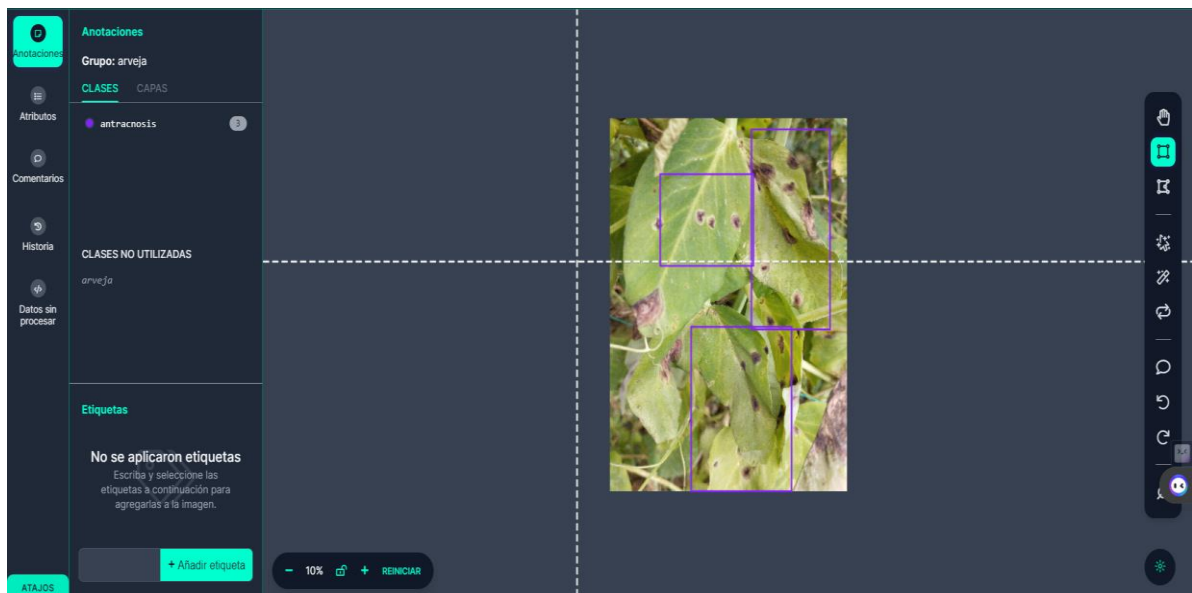


Figura 58. Etiquetado de imágenes

d) Codificación de entrenamiento del modelo de Yolo V8

Para iniciar el proceso de entrenamiento del modelo, es crucial contar con un conjunto de datos de imágenes que permita asignar a cada una su etiqueta correspondiente en este caso las etiquetas son: antracosis, fusarium y oídio. Para llevar a cabo esta tarea, se empleó un notebook que utiliza el modelo YOLOv8 en Google Colab, para optimizar el consumo de recursos computacionales. Este enfoque se implementa mediante la configuración de un archivo con extensión yaml, con la estructura detallada a continuación.

Para llevar a cabo el entrenamiento, hacemos uso de diversas bibliotecas integradas en el notebook del modelo YOLOv8 específicamente para esta investigación.

```
data.yaml X
C: > Users > ASUS > Downloads > carpeta yolo > data.yaml
1  train: ../train/images # training images
2  val: ../valid/images # validation images
3  test: ../test/images # test images
4
5  nc: 3 # classes
6  names: ['Antracnosis', 'Fusarium', 'Oidio']
7
8
```

Figura 59. Archivo data.yaml

```
!nvidia-smi
import os
HOME = os.getcwd()
print(HOME)
!pip install ultralytics==8.0.196

from IPython import display
display.clear_output()

import ultralytics
ultralytics.checks()
from ultralytics import YOLO

from IPython.display import display, Image
```

Figura 60. Instalación y uso de la biblioteca Ultralytics

```
!mkdir {HOME}/datasets
%cd {HOME}/datasets

!pip install roboflow

from roboflow import Roboflow
rf = Roboflow(api_key="tDW8s9t2HqATf0V54t0E")
project = rf.workspace("app-zlb9m").project("tesis_arveja")
version = project.version(1)
dataset = version.download("yolov8")
```

Figura 61. Entorno para entrenar un modelo con Roboflow

e) Número de épocas del algoritmo

En este apartado se visualiza el entrenamiento y el número de épocas es decir al número de veces que el algoritmo va a analizar los datos.

```
%cd {HOME}
!yolo task=detect mode=train model=yolov8s.pt data={dataset.location}/data.yaml epochs=100 imgsz=800 plots=True
```

Class	Images	Instances	Box(P)	R	mAP50	mAP50-95)
all	92	359	0.34	0.427	0.29	0.0969

```
Epoch GPU_mem box_loss cls_loss dfl_loss Instances Size
90/100 6.776 1.321 1.23 1.495 11 800: 100% 21/21 [00:00:00:00, 2.34it/s]
Class Images Instances Box(P) R mAP50 mAP50-95): 100% 3/3 [00:01:00:00, 1.62it/s]
all 92 359 0.375 0.378 0.28 0.0975
```

Closing dataloader mosaic
 augmentations: Blur(p=0.01, blur_limit=(3, 7)), MedianBlur(p=0.01, blur_limit=(3, 7)), ToGray(p=0.01), CLAHE(p=0.01, clip_limit=(1, 4.0), tile_grid_size=(8, 8))

Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size
91/100	6.786	1.507	1.421	1.666	7	800: 100% 21/21 [00:11:00:00, 1.84it/s]

Class	Images	Instances	Box(P)	R	mAP50	mAP50-95)
all	92	359	0.375	0.373	0.283	0.0972

Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size
92/100	6.856	1.453	1.322	1.635	3	800: 100% 21/21 [00:08:00:00, 2.49it/s]

Class	Images	Instances	Box(P)	R	mAP50	mAP50-95)
all	92	359	0.378	0.359	0.268	0.0974

Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size
93/100	6.546	1.398	1.256	1.687	11	800: 100% 21/21 [00:08:00:00, 2.47it/s]

Class	Images	Instances	Box(P)	R	mAP50	mAP50-95)
all	92	359	0.356	0.36	0.266	0.09

Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size
94/100	6.816	1.321	1.223	1.583	2	800: 100% 21/21 [00:08:00:00, 2.51it/s]

Class	Images	Instances	Box(P)	R	mAP50	mAP50-95)
all	92	359	0.333	0.38	0.27	0.0918

Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size
95/100	6.816	1.348	1.26	1.574	9	800: 100% 21/21 [00:08:00:00, 2.35it/s]

Class	Images	Instances	Box(P)	R	mAP50	mAP50-95)
all	92	359	0.354	0.372	0.264	0.0915

Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size
96/100	6.786	1.32	1.199	1.538	8	800: 100% 21/21 [00:09:00:00, 2.29it/s]

Class	Images	Instances	Box(P)	R	mAP50	mAP50-95)
all	92	359	0.356	0.363	0.261	0.0874

Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size
97/100	6.786	1.347	1.195	1.552	6	800: 100% 21/21 [00:09:00:00, 2.28it/s]

Class	Images	Instances	Box(P)	R	mAP50	mAP50-95)
all	92	359	0.34	0.377	0.266	0.0878

Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size
98/100	6.786	1.334	1.177	1.525	13	800: 100% 21/21 [00:09:00:00, 2.31it/s]

Class	Images	Instances	Box(P)	R	mAP50	mAP50-95)
all	92	359	0.345	0.405	0.272	0.0902

Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size
99/100	6.816	1.385	1.234	1.592	7	800: 100% 21/21 [00:08:00:00, 2.34it/s]

Class	Images	Instances	Box(P)	R	mAP50	mAP50-95)
all	92	359	0.369	0.366	0.273	0.0913

Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size
100/100	6.536	1.281	1.137	1.516	3	800: 100% 21/21 [00:09:00:00, 2.28it/s]

Class	Images	Instances	Box(P)	R	mAP50	mAP50-95)
all	92	359	0.415	0.351	0.28	0.0917

Figura 62. Número de épocas del algoritmo

```
Validating runs/detect/train/weights/best.pt...
Ultralytics YOLOv8.0.196 Python-3.10.12 torch-2.1.0+cu121 CUDA:0 (Tesla T4, 15102MiB)
Model summary (fused): 168 layers, 11126745 parameters, 0 gradients, 28.4 GFLOPs
```

Class	Images	Instances	Box(P)	R	mAP50	mAP50-95)
all	92	359	0.446	0.371	0.328	0.112
Antracnosis	92	174	0.417	0.494	0.395	0.128
Fusarium	92	96	0.463	0.281	0.327	0.12
Oidio	92	89	0.459	0.337	0.262	0.0865

```
Speed: 0.3ms preprocess, 8.0ms inference, 0.0ms loss, 4.7ms postprocess per image
Results saved to runs/detect/train
```

Figura 63. Rendimiento de la detección en diferentes clases

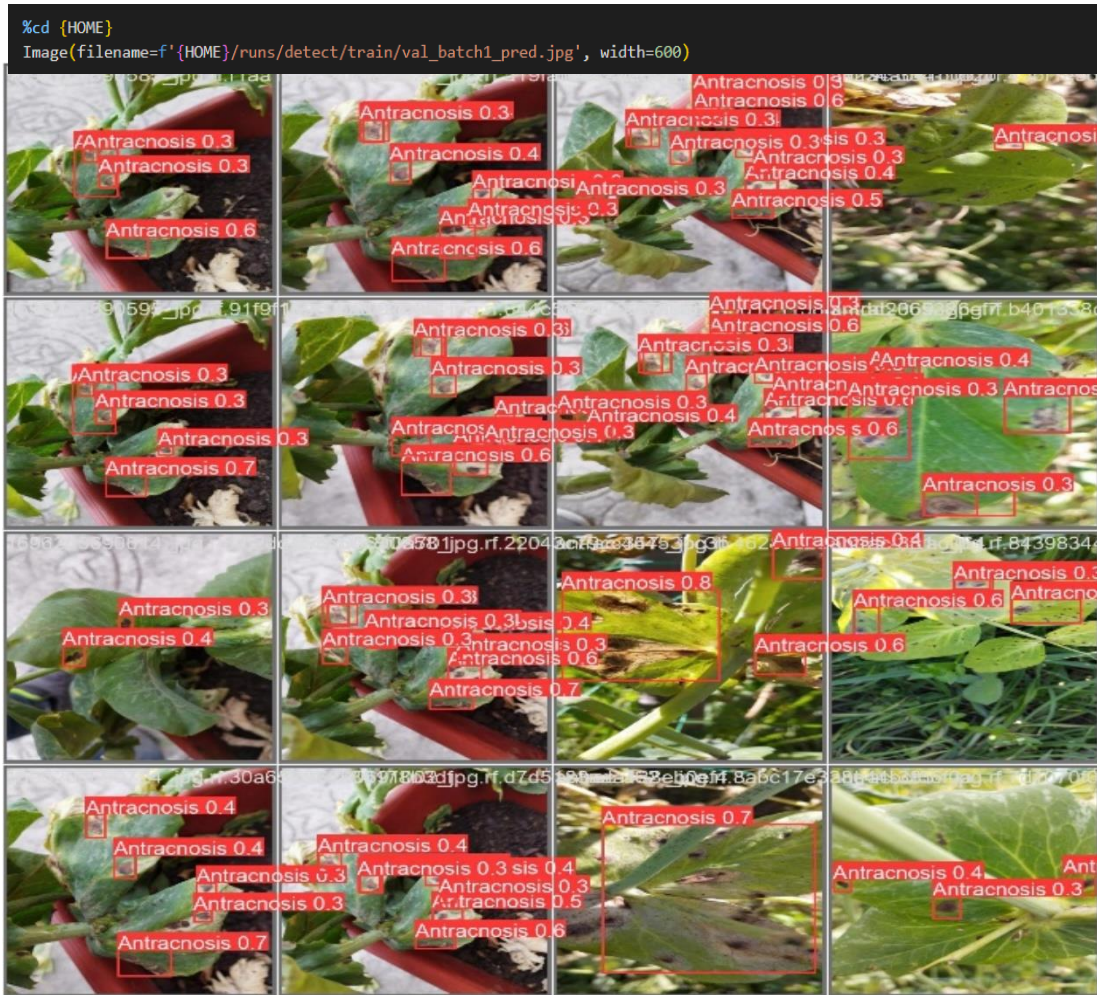


Figura 64. Test del modelo para la detección de antracnosis en arveja

```

%cd {HOME}
!yolo task=detect mode=predict model={HOME}/runs/detect/train/weights/best.pt conf=0 source={dataset.location}/test/images save=True

/content
Ultralytics YOLOv8.0.196 Python-3.10.12 torch-2.2.1+cu121 CUDA:0 (Tesla T4, 15102MiB)
Model summary (fused): 168 layers, 11126745 parameters, 0 gradients, 28.4 GFLOPs

WARNING ⚠ NMS time limit 0.550s exceeded
image 1/46 /content/Dataset-wlm-1/test/images/1696199614614_jpg.rf.4d9401892923002cfe9ca420a58a9ee6.jpg: 800x800 3 Antracnosis, 285 Fusariums, 12 Oidios, 22.5ms
image 2/46 /content/Dataset-wlm-1/test/images/1696199614643_jpg.rf.880275bf38a047dc91099cfa759b9de4.jpg: 800x800 300 Fusariums, 22.5ms
image 3/46 /content/Dataset-wlm-1/test/images/1696199614652_jpg.rf.f96969646972e255513e1112d9423ca7.jpg: 800x800 300 Fusariums, 22.5ms
image 4/46 /content/Dataset-wlm-1/test/images/1696199614688_jpg.rf.712fefb94c00abe8e5cf50df6b0275c3.jpg: 800x800 8 Antracnosis, 291 Fusariums, 1 Oidio, 22.4ms
image 5/46 /content/Dataset-wlm-1/test/images/1696199614793_jpg.rf.3726cb2ac93ab76b6a3cca61a43d7c97.jpg: 800x800 300 Fusariums, 22.4ms
image 6/46 /content/Dataset-wlm-1/test/images/1696200490407_jpg.rf.9752f03f2e3101a604def0ad4136f6ff.jpg: 800x800 30 Antracnosis, 210 Fusariums, 60 Oidios, 22.4ms
image 7/46 /content/Dataset-wlm-1/test/images/1696200490452_jpg.rf.c26df539b64845431bb6b644e5c9534c.jpg: 800x800 3 Antracnosis, 295 Fusariums, 2 Oidios, 22.4ms
image 8/46 /content/Dataset-wlm-1/test/images/1696200490537_jpg.rf.f062dbe362c4c4e49da71637bc0c20a9.jpg: 800x800 65 Antracnosis, 234 Fusariums, 1 Oidio, 22.4ms
image 9/46 /content/Dataset-wlm-1/test/images/1696200490599_jpg.rf.dea9cab5e16658bb694607d6fbc8d16.jpg: 800x800 3 Antracnosis, 295 Fusariums, 2 Oidios, 22.4ms
image 10/46 /content/Dataset-wlm-1/test/images/1696200689817_jpg.rf.c3fa07f1604f1147aef2d18f60b86abe.jpg: 800x800 194 Antracnosis, 77 Fusariums, 29 Oidios, 22.4ms
image 11/46 /content/Dataset-wlm-1/test/images/1696200689941_jpg.rf.80136994f369e89e874a29833dab4e29.jpg: 800x800 167 Antracnosis, 125 Fusariums, 8 Oidios, 22.4ms
image 12/46 /content/Dataset-wlm-1/test/images/1696200690023_jpg.rf.76e7425ff7a6416e986c9c34ad4a37a1.jpg: 800x800 256 Antracnosis, 37 Fusariums, 7 Oidios, 22.4ms
image 13/46 /content/Dataset-wlm-1/test/images/1696200690034_jpg.rf.4ec15b6a0ef663238a0e9055f74d78f.jpg: 800x800 233 Antracnosis, 67 Fusariums, 22.4ms
image 14/46 /content/Dataset-wlm-1/test/images/1696200690203_jpg.rf.4b7b3e572e246f98537f62168058e7e5.jpg: 800x800 233 Antracnosis, 42 Fusariums, 25 Oidios, 22.5ms
image 15/46 /content/Dataset-wlm-1/test/images/1696200690239_jpg.rf.d6765a210f57b3ad0ff0cbb38539208.jpg: 800x800 217 Antracnosis, 44 Fusariums, 39 Oidios, 22.5ms
image 16/46 /content/Dataset-wlm-1/test/images/1696200690272_jpg.rf.39e7043e8f6ca9eeda734b1da04cfb444.jpg: 800x800 240 Antracnosis, 19 Fusariums, 41 Oidios, 22.4ms
image 17/46 /content/Dataset-wlm-1/test/images/1696200690367_jpg.rf.2ecc63f484b5c9a67833df010911a1ef.jpg: 800x800 258 Antracnosis, 1 Fusarium, 41 Oidios, 22.4ms
image 18/46 /content/Dataset-wlm-1/test/images/1696200690477_jpg.rf.14f6e84c0a86e78856502d57e4649a4.jpg: 800x800 143 Antracnosis, 64 Fusariums, 93 Oidios, 22.4ms
image 19/46 /content/Dataset-wlm-1/test/images/1696200690487_jpg.rf.41791bec6ee1f32d1a5fce2b9984add.jpg: 800x800 116 Antracnosis, 75 Fusariums, 109 Oidios, 22.4ms

```

Figura 65. Rendimiento del modelo YOLOv8 en imágenes de prueba

```
import glob
from IPython.display import Image, display

for image_path in glob.glob(f'{HOME}/content/runs/detect/predict/1696200490537_jpg.rf.f062dbe362c4c4e49da71637bc0c20a9.jpg')[1:3]:
    display(Image(filename=image_path, width=600))
    print("\n")
```

Figura 66. Búsqueda de imágenes según un patrón en un directorio específico

```
#Run inference on your model on a persistent, auto-scaling, cloud API

#load model
model = project.version(dataset.version).model

#choose random test set image
import os, random
test_set_loc = dataset.location + "/test/images/"
random_test_image = random.choice(os.listdir(test_set_loc))
print("running inference on " + random_test_image)

pred = model.predict(test_set_loc + random_test_image, confidence=40, overlap=30).json()
pred
```

Figura 67. Inferencias en un modelo utilizando una API

```
running inference on antrac-258-_jpg.rf.87989809843759e1e11cda2b28e77e48.jpg
{'predictions': [{'x': 430.5,
  'y': 168.0,
  'width': 341.0,
  'height': 170.0,
  'confidence': 0.6725567579269409,
  'class': 'Antracnosis',
  'class_id': 0,
  'detection_id': 'bdcb5fd6-dd4c-44e1-842f-f4066e142802',
  'image_path': '/content/Dataset-wlm-1/test/images/antrac-258-_jpg.rf.87989809843759e1e11cda2b28e77e48.jpg',
  'prediction_type': 'ObjectDetectionModel'}],
  'image': {'width': '640', 'height': '640'}}
```

Figura 68. ID de detección y la ruta de la imagen procesada

f) Codificación de la aplicación en Visual Studio Code

En el proceso de codificación del proyecto Plant Vetch, se incorporaron dependencias externas esenciales para garantizar su correcto funcionamiento. Estas dependencias fueron especificadas y configuradas en el archivo pubspec.yaml, lo que facilita la ejecución fluida del proyecto.

```
1  ependencies:
2    flutter:
3      sdk: flutter
4
5    cupertino_icons: ^1.0.2
6    http: ^1.1.2
7    flutter_meedu: ^5.2.1
8    freezed: ^1.1.1
9    shared_preferences: ^2.0.12
10   google_fonts: ^4.0.3
11   flutter_svg: ^1.0.3
12   google_sign_in: ^5.2.3
13   flutter_facebook_auth: ^4.0.1
14   crypto: ^3.0.1
15   jwt_decoder: ^2.0.1
16   flutter_launcher_icons: ^0.9.2
17   image_picker: ^1.0.1
18   flutter_bloc: ^8.1.3
19   url_launcher: ^6.2.1
20   cloud_firestore: ^4.2.0
21   firebase_auth: ^4.2.1
22   firebase_core: ^2.4.0
23   firebase_storage: ^11.0.8
24   google_ml_kit: ^0.7.3
25   camera: ^0.8.1
26   carousel_slider: ^4.2.1
27   path_provider: ^2.0.5
28   cached_network_image: ^3.1.0
29   flutter_pdfview: 1.2.9
30   #tflite: ^1.1.2
31   tflite_flutter: ^0.10.4
32   youtube_player_flutter: ^8.1.2
33   flutter_inappwebview: ^5.3.2
34   chewie: ^1.5.0
35   image_editor: ^1.3.0
36   image_cropper: ^1.3.0
37   pdf_renderer: ^1.4.0
38   photo_view: ^0.14.0
39   gesture_zoom_box: ^0.0.4
40   permission_handler: ^11.0.0
41   audioplayers: ^5.2.0
```

Figura 69. Dependencias en el archivo pubspec.yaml

g) Codificación para desarrollo de las interfaces del aplicativo

La aplicación se construyó haciendo el uso de framework Flutter juntamente con el lenguaje de programación Dart, lo que posibilita la elaboración de aplicativos móviles. La estructura del software se organiza de manera intuitiva y funcional, brindando una experiencia de usuario amigable. En la interfaz principal, se encuentra una barra de pestañas en la parte inferior para lograr una navegación mas amigable entre las distintas secciones clave.

```
1 import 'package:flutter/material.dart';
2 import 'package:firebase_core/firebase_core.dart';
3 import 'package:quantum_app/app/inject_dependencies.dart';
4 import 'package:flutter_meedu/router.dart' as router;
5 import 'app/my_app.dart';
6
7 void main() async {
8   WidgetsFlutterBinding.ensureInitialized();
9   await Firebase.initializeApp();
10  await injectDependencies();
11  router.setDefaultTransition(
12    router.Transition.fadeIn,
13    duration: const Duration(milliseconds: 200),
14  );
15  runApp(
16    const MyApp(),
17  );
18 }
19
```

Figura 70. Función main

```

1 class RegisterPage extends StatelessWidget {
2   const RegisterPage({Key? key}) : super(key: key);
3
4
5   @override
6   Widget build(BuildContext context) {
7     final appBar = AppBar(
8       elevation: 0,
9     );
10    final padding = context.mediaQueryPadding;
11    final height = context.height - padding.top - padding.bottom - appBar.preferredSize.height;
12
13    return ProviderListener<RegisterController>(
14      provider: registerProvider,
15      builder: (_, controller) {
16        return Scaffold(
17          appBar: appBar,
18          body: ListView(
19            children: [
20              SizedBox(
21                height: height,
22                child: GestureDetector(
23                  onTap: () => FocusScope.of(context).unfocus(),
24                  child: Container(
25                    width: double.infinity,
26                    height: double.infinity,
27                    color: Colors.transparent,
28                    padding: const EdgeInsets.all(15),
29                    child: Align(
30                      child: Form(
31                        key: controller.formKey,
32                        child: ConstrainedBox(
33                          constraints: const BoxConstraints(
34                            maxWidth: 360,
35                          ),
36                          child: Column(
37                            mainAxisAlignment:
38                              context.isTablet ? MainAxisAlignment.center
39                              : MainAxisAlignment.end,
40                            children: [
41                              const Text(
42                                "Registrarse",
43                                textAlign: TextAlign.center,
44                                style: TextStyle(
45                                  fontSize: 28,
46                                  fontWeight: FontWeight.bold,
47                                ),
48                              ),
49                              const SizedBox(height: 15),
50                              CustomInputField(
51                                label: "Nombre",
52                                onChanged: controller.onNameChanged,
53
54                                validator: (text) {
55                                  return isValidName(text!) ? null : "Nombre inválido";
56                                },
57                              ),
58                              const SizedBox(height: 15),
59                              CustomInputField(
60                                label: "Apellido",
61                                onChanged: controller.onLastNameChanged,
62                                validator: (text) {
63                                  return isValidName(text!) ? null : "Apellido inválido";
64                                },
65                              ),
66                              const SizedBox(height: 15),
67                              CustomInputField(
68                                label: "Correo electrónico",
69                                inputType: TextInputType.emailAddress,
70                                onChanged: controller.onEmailChanged,
71                                validator: (text) {
72                                  return isValidEmail(text!) ? null :
73                                  "Correo electrónico inválido";
74                                }
75                              )
76                            ],
77                          ),
78                        ),
79                      ),
80                    ),
81                  ),
82                ),
83              ),
84            ],
85          ),
86        );
87      },
88    );
89  }
90 }

```

Figura 71. Script registro usuario

```

1
2
3 @override
4 Widget build(BuildContext context) {
5   final appBar = AppBar(
6     elevation: 0,
7   );
8   final padding = context.mediaQueryPadding;
9   final height = context.height - padding.top - padding.bottom - appBar.preferredSize.height;
10
11   return ProviderListener<RegisterController>(
12     provider: registerProvider,
13     builder: (_, controller) {
14       return Scaffold(
15         appBar: appBar,
16         body: ListView(
17           children: [
18             SizedBox(
19               height: height,
20               child: GestureDetector(
21                 onTap: () => FocusScope.of(context).unfocus(),
22                 child: Container(
23                   width: double.infinity,
24                   height: double.infinity,
25                   color: Colors.transparent,
26                   padding: const EdgeInsets.all(15),
27                   child: Align(
28                     child: Form(
29                       key: controller.formKey,
30                       child: ConstrainedBox(
31                         constraints: const BoxConstraints(
32                           maxWidth: 360,
33                         ),
34                       child: Column(
35                         mainAxisAlignment:
36                           context.isTablet ? MainAxisAlignment.center
37                           : MainAxisAlignment.end,
38                         children: [
39                           const Text(
40                             "Registrarse",
41                             textAlign: TextAlign.center,
42                             style: TextStyle(
43                               fontSize: 28,
44                               fontWeight: FontWeight.bold,
45                             ),
46                           ),
47                           const SizedBox(height: 15),
48                           CustomInputField(
49                             label: "Nombre",
50                             onChanged: controller.onNameChanged,
51
52                             validator: (text) {
53                               return isValidName(text!) ? null : "Nombre inválido";
54                             },
55                           ),
56                           const SizedBox(height: 15),
57                           CustomInputField(
58                             label: "Apellido",
59                             onChanged: controller.onLastNameChanged,
60                             validator: (text) {
61                               return isValidName(text!) ? null : "Apellido inválido";
62                             },
63                           ),
64                           const SizedBox(height: 15),
65                           CustomInputField(
66                             label: "Correo electrónico",
67                             inputType: TextInputType.emailAddress,
68                             onChanged: controller.onEmailChanged,
69                             validator: (text) {
70                               return isValidEmail(text!) ? null :
71                                 "Correo electrónico inválido";
72                             }
73                           )
74                         ]

```

Figura 72. Script inicio de sesión

```

1 class ResetPasswordPage extends StatelessWidget {
2   const ResetPasswordPage({Key? key}) : super(key: key);
3
4   @override
5   Widget build(BuildContext context) {
6     final isDarkMode = context.isDarkMode;
7     final padding = context.mediaQueryPadding;
8     final height = context.height - padding.top - padding.bottom;
9     return ProviderListener<ResetPasswordController>(
10      provider: resetPasswordProvider,
11      builder: (_, controller) => Scaffold(
12        appBar: AppBar(),
13        body: ListView(
14          children: [
15            SizedBox(
16              height: height,
17              child: GestureDetector(
18                onTap: () => FocusScope.of(context).unfocus(),
19                child: Container(
20                  padding: const EdgeInsets.all(15),
21                  width: double.infinity,
22                  color: Colors.transparent,
23                  child: Align(
24                    child: ConstrainedBox(
25                      constraints: const BoxConstraints(
26                        maxWidth: 360,
27                      ),
28                      child: Column(
29                        mainAxisAlignment: MainAxisAlignment.center,
30                        crossAxisAlignment: CrossAxisAlignment.start,
31                        children: [
32                          AspectRatio(
33                            aspectRatio: 16 / 9,
34                            child: SvgPicture.asset(
35                              'assets/images/${isDarkMode ? 'dark' : 'light'}
36                              /forgot_password.svg',
37                            ),
38                          const SizedBox(height: 10),
39                          const Center(
40                            child: Text(
41                              "Restablecer la contraseña",
42                              textAlign: TextAlign.center,
43                              style: TextStyle(
44                                fontSize: 28,
45                                fontWeight: FontWeight.bold,
46                              ),
47                            ),
48                          const SizedBox(height: 20),
49                          CustomInputField(
50                            label: "Correo electrónico",
51                            onChanged: controller.onEmailChanged,
52                            inputType: TextInputType.emailAddress,
53                          ),
54                          const SizedBox(height: 5),
55                          const Text(
56                            "Ingrese su correo electrónico para recibir un enlace para cambiar su contraseña",
57                          ),
58                          const SizedBox(height: 20),
59                          SizedBox(
60                            width: double.infinity,
61                            child: RoundedButton(
62                              onPressed: () => _submit(context),
63                              text: "Enviar",
64                            ),
65                          ),
66                          const SizedBox(height: 30),
67                        ],
68                      ),
69                    ),
70                  ),
71                ),
72              ),
73            ),
74          ],
75        ),
76      ),
77    );
78  }
79 }

```

Figura 73. Script reestablecer contraseña

```
1 class HomePage extends StatelessWidget {
2   const HomePage({Key? key}) : super(key: key);
3
4   @override
5   Widget build(BuildContext context) {
6     return ProviderListener<HomeController>(
7       provider: homeProvider,
8       builder: (_, controller) {
9         return Scaffold(
10          bottomNavigationBar: HomeTabBar(),
11          body: SafeArea(
12            child: TabBarView(
13              children: const [
14                HomeTab(),
15                InformacionTab(),
16                Home(),
17                LibrosTab(),
18                ProfileTab(),
19              ],
20              controller: controller.tabController,
21            ),
22          ),
23        );
24      },
25    );
26  }
27 }
28
```

Figura 74. Script Homepage

```
1 import 'package:flutter/material.dart';
2
3 void main() {
4   runApp(const MaterialApp(
5     home: HomeTab(),
6   ));
7 }
8
9 class HomeTab extends StatelessWidget {
10  const HomeTab({Key? key}) : super(key: key);
11
12  @override
13  Widget build(BuildContext context) {
14    return Scaffold(
15      body: Stack(
16        children: <Widget>[
17          Image.asset(
18            "assets/images/kimf.jpg",
19            fit: BoxFit.cover,
20            height: MediaQuery.of(context).size.height,
21            width: MediaQuery.of(context).size.width,
22          ),
23        ],
24      ),
25    );
26  }
27 }
28
```

Figura 75. Script hometab

```

1 import 'package:flutter/material.dart';
2 import 'package:quantum_app/app/ui/pages/home/tabs/informacion/taxonomia.dart';
3 import 'package:quantum_app/app/ui/pages/home/tabs/informacion/cosecha.dart';
4 import 'package:quantum_app/app/ui/pages/home/tabs/informacion/enfermedades.dart';
5 import 'package:quantum_app/app/ui/pages/home/tabs/informacion/plagas.dart';
6 import 'package:quantum_app/app/ui/pages/home/tabs/informacion/requerimientos.dart';
7 import 'package:quantum_app/app/ui/pages/home/tabs/informacion/requerimientos2.dart';
8 import 'package:quantum_app/app/ui/pages/home/tabs/informacion/botones.dart';
9
10 class InformacionTab extends StatefulWidget {
11   const InformacionTab({Key? key}) : super(key: key);
12
13   @override
14   _InformacionTabState createState() => _InformacionTabState();
15 }
16
17 class _InformacionTabState extends State<InformacionTab> {
18   final List<String> _pdfTitles = [
19     "Consejos útiles",
20     "Consejos útiles",
21     "Consejos útiles",
22     "Consejos útiles",
23     "Consejos útiles",
24     "Consejos útiles",
25     "Consejos útiles",
26   ];
27
28   final List<String> _pdfSubtitles = [
29     "1. Taxonomía del cultivo",
30     "2. Requerimientos nutricionales",
31     "3. Requerimientos edafoclimáticos",
32     "4. Plagas",
33     "5. Enfermedades",
34     "6. Cosecha",
35     "7. Información del cultivo ",
36   ];
37
38   final List<String> _pdfDescriptions = [
39     "Clasificación biológica de la arveja variedad Quantum dentro de la especie Pisum sativum y la familia Fabaceae.",
40     "Nutrientes esenciales necesarios para el crecimiento saludable de la arveja variedad Quantum, incluyendo nitrógeno, fósforo y potasio.",
41     "Condiciones climáticas óptimas, como temperatura y humedad, para el cultivo exitoso de la arveja variedad Quantum.",
42     "Organismos dañinos como insectos o animales que pueden afectar negativamente la arveja variedad Quantum.",
43     "Afecciones causadas por patógenos como hongos, bacterias o virus que pueden debilitar o dañar la arveja variedad Quantum.",
44     "Proceso de recolección de guisantes maduros de la arveja variedad Quantum para su consumo o almacenamiento.",
45     "Conocimientos y prácticas específicas para el cultivo exitoso de la arveja variedad Quantum, incluyendo siembra, riego, fertilización y cuidados necesarios.",
46   ];
47
48   final List<String> _imagePaths = [
49     "assets/images/taxonomia2.jpg",
50     "assets/images/re.jpg",
51     "assets/images/re2.jpg",
52     "assets/images/arañas.jpg",
53     "assets/images/mlldu.jpg",
54     "assets/images/cosecha2.jpg",
55     "assets/images/como.jpg",
56   ];
57
58   final String _mainTitle = "MANEJO AGRONÓMICO";
59
60   bool _isDarkMode = false;
61
62   @override
63   void initState() {
64     super.initState();
65   }
66

```

Figura 76. Script informacióntab

```

1 Widget _galleryBody() {
2   return ListView(shrinkWrap: true, children: [
3     _image != null
4       ? SizedBox(
5         height: 400,
6         width: 400,
7         child: Stack(
8           fit: StackFit.expand,
9           children: <Widget>[
10            Image.file(_image!),
11            if (widget.customPaint != null) widget.customPaint!,
12          ],
13        ),
14      )
15      : const Icon(
16        Icons.image,
17        size: 200,
18      ),
19    Padding(
20      padding: const EdgeInsets.symmetric(horizontal: 16),
21      child: ElevatedButton(
22        child: const Text('From Gallery'),
23        onPressed: () => _getImage(ImageSource.gallery),
24      ),
25    ),
26    Padding(
27      padding: const EdgeInsets.symmetric(horizontal: 16),
28      child: ElevatedButton(
29        child: const Text('Take a picture'),
30        onPressed: () => _getImage(ImageSource.camera),
31      ),
32    ),
33  ]);
34 }
35
36 Future _getImage(ImageSource source) async {
37   final pickedFile = await _imagePicker?.pickImage(source: source);
38   if (pickedFile != null) {
39     _processPickedFile(pickedFile);
40   } else {
41     if (kDebugMode) {
42       print('No image selected.');
```

Figura 77. Script cámara

```

1 import 'dart:async';
2 import 'dart:io';
3 import 'package:flutter/material.dart';
4 import 'package:image_picker/image_picker.dart';
5 import 'package:image_cropper/image_cropper.dart';
6 import 'package:quantum_app/app/ui/pages/home/tabs/deteccion/control.dart';
7 import 'package:quantum_app/app/ui/pages/home/tabs/deteccion/image_functions.dart';
8
9 void main() {
10   runApp(const MaterialApp(
11     home: Home(),
12   ));
13 }
14
15 class Home extends StatefulWidget {
16   const Home({Key? key}) : super(key: key);
17
18   @override
19   _HomeState createState() => _HomeState();
20 }
21
22 class _HomeState extends State<Home> {
23   File? imageFile;
24   List<Map<String, dynamic>> highProbabilityPredictions = [];
25
26   @override
27   Widget build(BuildContext context) {
28     return Scaffold(
29       appBar: AppBar(
30         centerTitle: true,
31         title: const Text(
32           'Detección de enfermedades',
33           style: TextStyle(
34             fontSize: 20,
35             fontWeight: FontWeight.normal,
36             color: Colors.black,
37           ),
38         ),
39       backgroundColor: const Color.fromARGB(255, 198, 197, 197),
40       shadowColor: Colors.green,
41       shape: RoundedRectangleBorder(
42         borderRadius: BorderRadius.circular(0),
43       ),
44     ),
45     body: Column(
46       children: [
47         Expanded(
48           child: Stack(
49             alignment: Alignment.center,
50             children: [
51               Positioned(
52                 top: MediaQuery.of(context).size.height * 0.2,
53                 child: Container(
54                   padding: const EdgeInsets.only(top: 16, bottom: 16),
55                   child: Text(
56                     imageFile == null
57                       ? "Imagen no seleccionada para el análisis!"
58                       : "",
59                     style: const TextStyle(
60                       color: Color.fromARGB(255, 27, 27, 27),
61                       fontSize: 16,
62                       fontWeight: FontWeight.normal,
63                     ),
64                   ),
65               ),
66             ],
67           ),
68         ),
69       ],
70     ),
71   ),
72 );

```

Figura 78. Script detección enfermedades

```

1 import 'package:flutter/material.dart';
2 import 'package:quantum_app/app/ui/pages/home/tabs/deteccion/recomendaciones.dart';
3 import 'package:quantum_app/app/ui/pages/home/tabs/deteccion/recomendaciones2.dart';
4 import 'package:quantum_app/app/ui/pages/home/tabs/deteccion/recomendaciones3.dart';
5
6 void main() {
7   runApp(ControlPage());
8 }
9
10 class ControlPage extends StatelessWidget {
11   @override
12   Widget build(BuildContext context) {
13     return MaterialApp(
14       home: Scaffold(
15         appBar: AppBar(
16           leading: IconButton(
17             icon: const Icon(Icons.arrow_back),
18             onPressed: () {
19               Navigator.pop(context);
20             },
21           ),
22           title: const Text('Control de Enfermedades'),
23           backgroundColor: Color.fromARGB(255, 198, 197, 197),
24         ),
25         body: SingleChildScrollView(
26           child: Column(
27             children: [
28               const SizedBox(height: 16.0),
29               MiContenedor(
30                 titulo: 'ANTRACNOSIS',
31                 subtitulo: '(Colletotrichum spp)',
32                 imagenPath: 'assets/images/manchaa.jpg',
33                 color: const Color.fromARGB(255, 199, 204, 196),
34                 onTap: () {
35                   Navigator.push(
36                     context,
37                     MaterialPageRoute(
38                       builder: (context) =>
39                         const RecomendacionesPage(titulo: 'ANTRACNOSIS'),
40                     ),
41                 );
42               },
43             ),
44               const SizedBox(height: 16.0),
45               MiContenedor(
46                 titulo: 'FUSARIUM',
47                 subtitulo: '(Fusarium oxysporum f. sp. pisi)',
48                 imagenPath: 'assets/images/fus2.jpg',
49                 color: const Color.fromARGB(255, 199, 204, 196),
50                 onTap: () {
51                   Navigator.push(
52                     context,
53                     MaterialPageRoute(
54                       builder: (context) =>
55                         const Recomendaciones2Page(titulo: 'FUSARIUM'),
56                     ),
57                 );
58               },
59             ),

```

Figura 79. Script control enfermedades

```
1 import 'package:flutter/material.dart';
2
3 class ButtonInfo {
4   final String title;
5   final String codigo;
6   final String provincia;
7   final String propietario;
8   final String latitud;
9   final String longitud;
10  final String altitud;
11  final String tipo;
12  final String estado;
13
14  ButtonInfo({
15    required this.title,
16    required this.codigo,
17    required this.provincia,
18    required this.propietario,
19    required this.latitud,
20    required this.longitud,
21    required this.altitud,
22    required this.tipo,
23    required this.estado,
24  });
25 }
26
27 class LibrosTab extends StatefulWidget {
28   const LibrosTab({Key? key}) : super(key: key);
29
30   @override
31   _LibrosTabState createState() => _LibrosTabState();
32 }
33
34 class _LibrosTabState extends State<LibrosTab> {
35   final List<String> subtítulos = [
36     'Zona meteorológica 1',
37     'Zona meteorológica 2',
38     'Zona meteorológica 3',
39     'Zona meteorológica 4'
40   ];
41
42   final List<String> imágenes = [
43     'assets/images/zona.jpg',
44     'assets/images/zona.jpg',
45     'assets/images/zona.jpg',
46     'assets/images/zona.jpg',
47   ];
48
49   int? selectedButtonIndex;
50
51   final List<ButtonInfo> buttonInfoList = [
52     ButtonInfo(
53       title: 'Tulcán',
54       codigo: 'UPEC/EMA1',
55       provincia: 'Carchi',
56       propietario: 'UPEC',
57       latitud: '0.8036162',
58       longitud: '-77.734786',
59       altitud: '2989.2 m.s.n.m',
60       tipo: 'Automática',
61       estado: 'Operativa',
62     ),
63     ButtonInfo(
64       title: 'Huaca',
65       codigo: 'UPEC/EMA2',
66       provincia: 'Carchi',
67       propietario: 'Upec',
68       latitud: '0.7987771',
69       longitud: '-77.7275173',
70       altitud: '2959 m.s.n.m',
71       tipo: 'Automática',
72       estado: 'Operativa',
73     ),
74   ],
```

Figura 80. Script clima arveja

```
1 @override
2 Widget build(BuildContext context) {
3   final sessionController = sessionProvider.read;
4   final user = sessionController.user!;
5   final displayName = user.displayName ?? '';
6   final letter = displayName.isNotEmpty ? displayName[0] : '';
7
8   return showAyuda
9     ? const AyudaPage()
10    : Container(
11      decoration: const BoxDecoration(),
12      child: SingleChildScrollView(
13        child: Column(
14          children: [
15            const SizedBox(height: 10),
16            CircleAvatar(
17              radius: 100,
18              child: user.photoURL == null
19                ? Text(
20                  letter,
21                  style: const TextStyle(fontSize: 65),
22                )
23                : null,
24              backgroundImage: user.photoURL != null
25                ? NetworkImage(user.photoURL!)
26                : null,
27            ),
28            const SizedBox(height: 10),
29            Center(
30              child: Text(
31                displayName,
32                style: const TextStyle(
33                  fontSize: 18,
34                  fontWeight: FontWeight.bold,
35                ),
36            ),
37          ],
38        ),
39      ),
40    );
41  }
```

Figura 81. Script perfil de usuario

4.1.2.1.4. Fase de pruebas

Esta fase corresponde a una parte fundamental de la metodología escogida para el progreso del trabajo, y tiene como meta asegurar la integridad y agilidad en el desarrollo del prototipo.

a) Test Lab Firebase

Como se visualiza en la Figura 82, se utilizó la herramienta Firebase Test Lab para llevar a cabo pruebas de rendimiento. Esta herramienta, que es de servicio gratuito, permite realizar pruebas en múltiples dispositivos en la nube para certificar el buen funcionamiento de la app en términos de su interfaz. Además, ofrece diversas

opciones para visualizar los detalles de la aplicación, generando informes en forma de videos e imágenes de las pruebas realizadas.

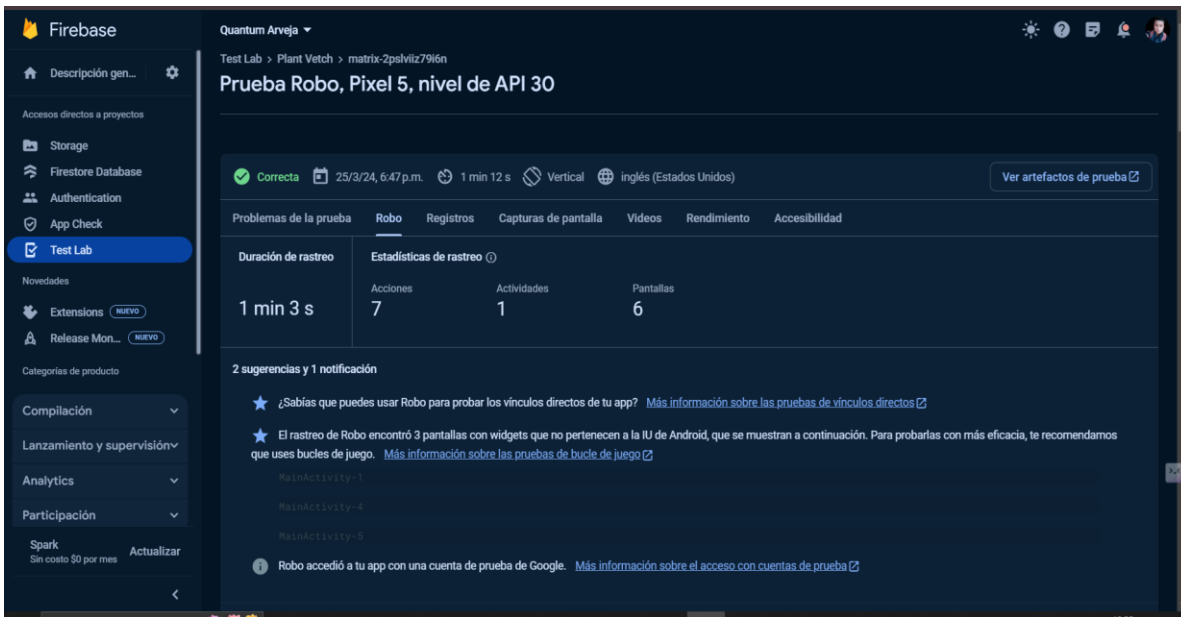


Figura 82. Prueba Robo, Quantum Arveja

Como se visualiza en la Figura 83, la prueba de rendimiento indica que la aplicación Quantum Arveja presenta los siguientes datos:

- Tiempo de inicio de la aplicación: 439 milisegundos.
- Tiempo necesario para la visualización inicial: 439 milisegundos
- Tiempo requerido para la visualización completa: 439 milisegundos.

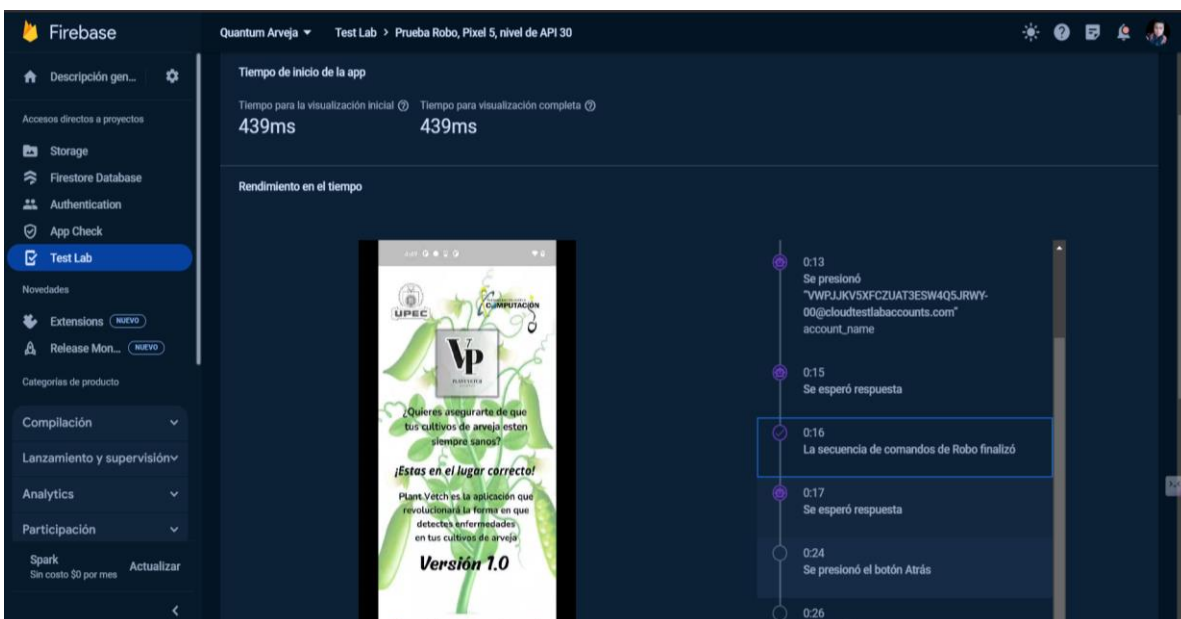


Figura 83. Rendimiento de la aplicación en Test lab

Los parámetros representados son:

Porcentaje de uso de la CPU: Muestra el grado de uso de la Unidad Central de Procesamiento (CPU) por parte del proceso, expresado en porcentaje. En este contexto, se observa que el uso de la CPU varía entre el 20% y el 10% en diferentes instantes (Figura 84).

Memoria (kB): Se exhibe el consumo de memoria del sistema por el proceso, medido en kilobytes (kB). En este caso particular, se registra un uso de memoria de 100 kB (Figura 84).

Red (Bytes/s): Ilustra la tasa de transmisión de datos mediante la red por el proceso, expresada en bytes por segundo. Sin embargo, en esta instancia no se muestran valores de transferencia en el gráfico (Figura 84).

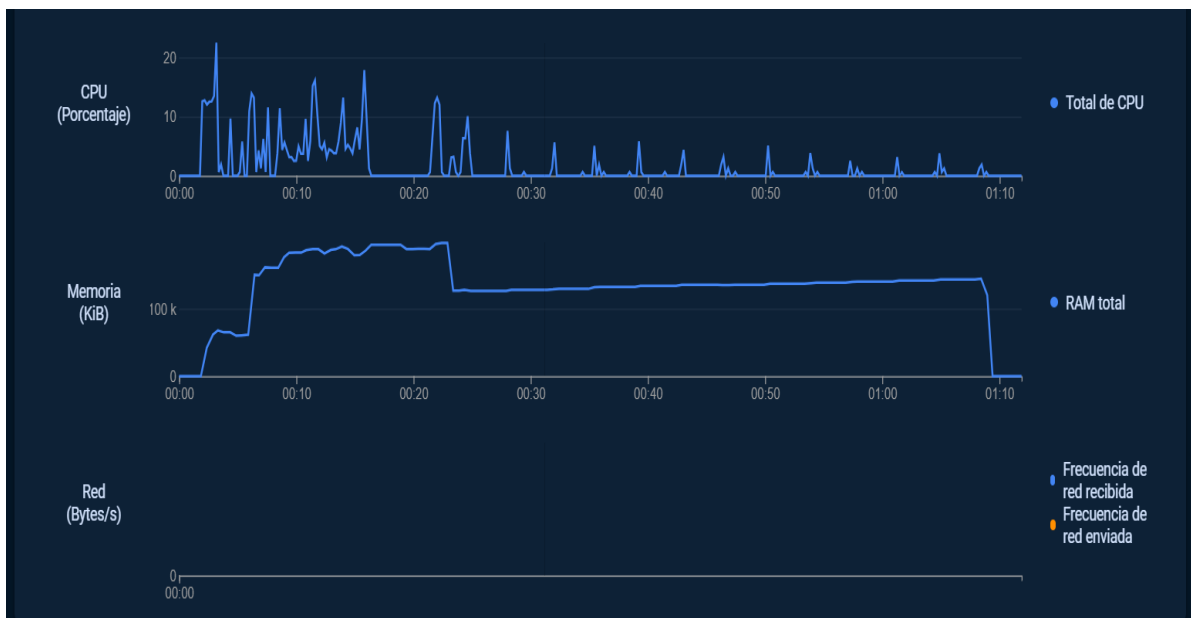


Figura 84. Rendimiento CPU, Memoria

b) Pruebas de rendimiento

Tabla 48. Pruebas de rendimiento

Marca	Nº	Modelo	Versión de Android	Memoria Ram	Tiempo de instalación	Rendimiento (1 al 10)	Tiempo de respuesta (s)	Precisión de detección (1 al 10)	Observaciones
Xiaomi	1	Redmi note 11S	android 13	8Gb	15s	9	3s	9	ninguna
	2	Redmi note 11	android 13	4Gb	18s	9	3s	9	ninguna
	3	Redmi note 9	android 10	4Gb	25s	9	4s	9	ninguna
	4	Redmi 12	android 12	8 GB	10s	9	3s	9	ninguna
	5	Redmi note 12 Pro +	android 11	8GB	10s	9	3s	9	ninguna
	6	Redmi note 9 S	android 12	4GB	18s	9	3s	9	ninguna
	7	Redmi note 10	android 13	4GB	19s	9	3s	9	ninguna
Samsung	1	Galaxy A03	android 13	-	54s	9	3s	9	ninguna
	2	Galaxy A12	android 13	4 GB	10s	9	3s	9	ninguna
	3	Galaxy A10	android 11	4 GB	15s	9	5s	9	ninguna
	4	Galaxy A10	android 10	4 GB	15s	9	5s	9	ninguna
	5	Galaxy A13	android 13	4 GB	8s	9	3s	9	ninguna
	1	Moto G13 SE	android 13	4GB	10s	9	4s	9	ninguna
Motorola	1	Moto G13 SE	android 13	4GB	10s	9	4s	9	ninguna
Huawei	1	Mate X5	android 13	16GB	8s	9	2s	9	ninguna
LG	1	K41S	Android 9	-	1:02s	9	5s	9	ninguna

c) Pruebas de aceptación

Tabla 49. Registro de usuario

Prueba de aceptación	
Nº: 1	N.º Historia de usuario: 1
Historia de usuario: Registro con datos de usuario	
Caso de prueba: Conexión con Firebase	
Detalle: La aplicación móvil ofrece opciones para el registro de datos necesarios para iniciar sesión.	
Condiciones de ejecución: Es importante que el procedimiento de registro en la aplicación móvil esté configurado de manera correcta.	
Entrada: Interacción de usuario.	
Resultado esperado: Inicio de sesión correcto mediante el registro realizado	
Evaluación: La función de registro accede a Firebase y dirige de manera efectiva al usuario al inicio de la aplicación.	

Tabla 50. Registro con la cuenta de Google

Prueba de aceptación	
Nº: 2	N.º Historia de usuario: 2
Historia de usuario: Registro mediante Google directamente	
Caso de prueba: Registro mediante Google	
Detalle: La aplicación móvil proporciona un proceso de registro sencillo y rápido utilizando la cuenta de Google.	
Condiciones de ejecución: Es crucial que la forma de registro en la aplicación móvil esté configurada correctamente.	
Entrada: Interacción de usuario.	
Resultado esperado: Ingreso exitoso a la aplicación mediante el correo electrónico.	
Evaluación: El botón de Google redirige a la cuenta del usuario y lleva a cabo la autenticación de manera eficaz.	

Tabla 51. Restablecimiento de la contraseña

Prueba de aceptación	
Nº: 3	N.º Historia de usuario: 3
Historia de usuario: Restablecimiento de contraseña	
Caso de prueba: Restablecer contraseña	
Detalle: La aplicación móvil proporciona una opción para recuperar la contraseña.	
Condiciones de ejecución: El ingreso de datos debe ser preciso para lograr un inicio de sesión efectivo.	
Entrada: Interacción de usuario.	
Resultado esperado: El proceso de restablecimiento de contraseña se realiza satisfactoriamente para acceder a la aplicación.	
Evaluación: El proceso de restablecimiento de contraseña envía una notificación al correo electrónico, lo que permite modificar la contraseña y así acceder a la aplicación.	

Tabla 52. Funcionamiento de opciones de navegación

Prueba de aceptación	
Nº: 4	N.º Historia de usuario: 7
Historia de usuario: Conexión aplicación móvil	
Caso de prueba: Funcionamiento de las opciones de navegación	
Detalle: La aplicación móvil ofrece opciones de navegación que incluyen botones interactivos, diseñados para ser accesibles y fáciles de usar.	
Condiciones de ejecución: Las rutas de navegación de la aplicación móvil deben estar correctamente configuradas.	
Entrada: Interacción de usuario.	
Resultado esperado: Las rutas de navegación de la aplicación móvil deben estar correctamente configuradas.	
Evaluación: Las opciones de navegación guían a las pantallas correspondientes, mientras que los menús de función ejecutan sus acciones correspondientes.	

Tabla 53. Captura de imágenes

Prueba de aceptación	
Nº: 5	N.º Historia de usuario: 8
Historia de usuario: Captura de imágenes.	
Caso de prueba: Captura de imágenes.	
Detalle: La aplicación móvil incluye la capacidad de capturar imágenes utilizando la cámara del dispositivo móvil.	
Condiciones de ejecución: El teléfono móvil debe contar con una cámara en funcionamiento.	
Entrada: Participación del usuario.	
Resultado esperado: La cámara se activa y el usuario puede capturar imágenes.	
Evaluación: Las imágenes se capturan correctamente y están disponibles para su uso dentro de la aplicación.	

Tabla 54. Selección de imágenes en galería

Prueba de aceptación	
Nº: 6	N.º Historia de usuario: 8
Historia de usuario: Funcionalidad de captura de imágenes	
Caso de prueba: Captura de imágenes galería	
Detalle: La aplicación móvil puede capturar imágenes utilizando la cámara del dispositivo móvil.	
Condiciones de ejecución: El teléfono móvil debe contar con una cámara en funcionamiento.	
Entrada: Participación del usuario.	
Resultado esperado: La capacidad de la cámara para capturar se ejecuta de manera adecuada.	
Evaluación: La cámara funciona correctamente y procesa los datos internamente dentro de la aplicación.	

Tabla 55. Recorte de imágenes

Prueba de aceptación	
Nº: 7	N.º Historia de usuario: 9
Historia de usuario: Funcionalidad de captura de imágenes	
Caso de prueba: Recorte de imágenes.	
Detalle: La función de recorte de imágenes está disponible en la aplicación móvil para ajustar tanto las imágenes capturadas como las seleccionadas desde la galería.	
Condiciones de ejecución: Antes de utilizar esta función, es necesario que el usuario haya capturado o seleccionado una imagen previamente.	
Entrada: imagen	
Resultado esperado: La aplicación móvil ofrece la posibilidad de recortar imágenes de manera conveniente.	
Evaluación: La capacidad de recortar imágenes en la aplicación móvil funciona adecuadamente, ya sea que se trate de una imagen capturada o seleccionada por el usuario.	

Tabla 56. Envío de imagen al análisis

Prueba de aceptación	
Nº: 8	N.º Historia de usuario: 10
Historia de usuario: Envío de imagen para su análisis API	
Caso de prueba: Envío de imagen para su análisis.	
Detalle: La aplicación móvil establece una conexión adecuada para enviar datos.	
Condiciones de ejecución: Antes de emplear esta función, el usuario necesita haber tomado o elegido una imagen con antelación y contar con una conexión a Internet sólida.	
Entrada: La imagen.	
Resultado esperado: La aplicación móvil transmite la imagen de manera adecuada para que pueda ser analizada.	
Evaluación: El proceso de enviar los datos se ha ejecutado correctamente.	

Tabla 57. Recepción de imagen etiquetada con resultados

Prueba de aceptación	
Nº: 9	N.º Historia de usuario: 11
Historia de usuario: Mostrando resultados.	
Caso de prueba: Imágenes etiquetadas con resultados.	
Detalle: La aplicación móvil recibe y analiza datos de imágenes.	
Condiciones de ejecución: El usuario debe enviar una foto.	

Entrada: La imagen.
 Resultado esperado: La aplicación móvil obtiene imágenes con resultados etiquetados.
 Evaluación: La aplicación para dispositivos móviles tiene la capacidad de recibir imágenes que han sido procesadas y mostrar los resultados correspondientes.

Tabla 58. Cuadro de diálogo emergente con resultados

Prueba de aceptación	
Nº: 10	N.º Historia de usuario: 11
Historia de usuario: Mostrar resultados.	
Caso de prueba: Cuadro de dialogo emergente de resultados.	
Detalle: La aplicación móvil exhibe imágenes con descripciones de las enfermedades encontradas.	
Condiciones de ejecución: Es fundamental que los usuarios dispongan de una conexión a Internet sólida y fiable.	
Entrada: Imagen.	
Resultado esperado: La aplicación móvil debe presentar un cuadro de diálogo emergente que muestre los resultados del análisis de manera clara y concisa.	
Evaluación: La aplicación móvil despliega un cuadro de diálogo emergente que contiene imágenes y descripciones exactas de las enfermedades detectadas.	

d) Resumen iteraciones

Tabla 59. Resumen iteraciones

Iteración	Historia de usuario	Fecha inicio	Fecha fin	Estado
	1	05/02/2023	15/02/2023	Cumple
	2	17/02/2023	04/03/2023	Cumple
1	3	06/03/2023	29/03/2023	Cumple
	4	02/04/2023	12/04/2023	Cumple
	5	18/04/2023	30/04/2023	Cumple
	6	12/05/2023	19/05/2023	Cumple
2	7	27/05/2023	10/06/2023	Cumple
	8	15/06/2023	29/06/2023	Cumple
	9	12/07/2023	02/09/2023	Cumple
	10	03/09/2023	11/10/2023	Cumple
	11	12/10/2023	13/11/2023	Cumple

De acuerdo a las pruebas realizadas en las tres patologías se pudo evidenciar que el modelo de detección muestra una mayor precisión que el anterior, tomando en cuenta que cada una de ellas muestran un desafío al momento de la detección siendo un factor importante la iluminación en la captura de imágenes, sin embargo este algoritmo ha desempeñado las características esenciales en este aspecto, cabe señalar que para este proceso la toma de fotografías debe ser lo más cerca posible para una mejor eficiencia y efectividad.

A continuación, se muestra el modelo de detección terminada.

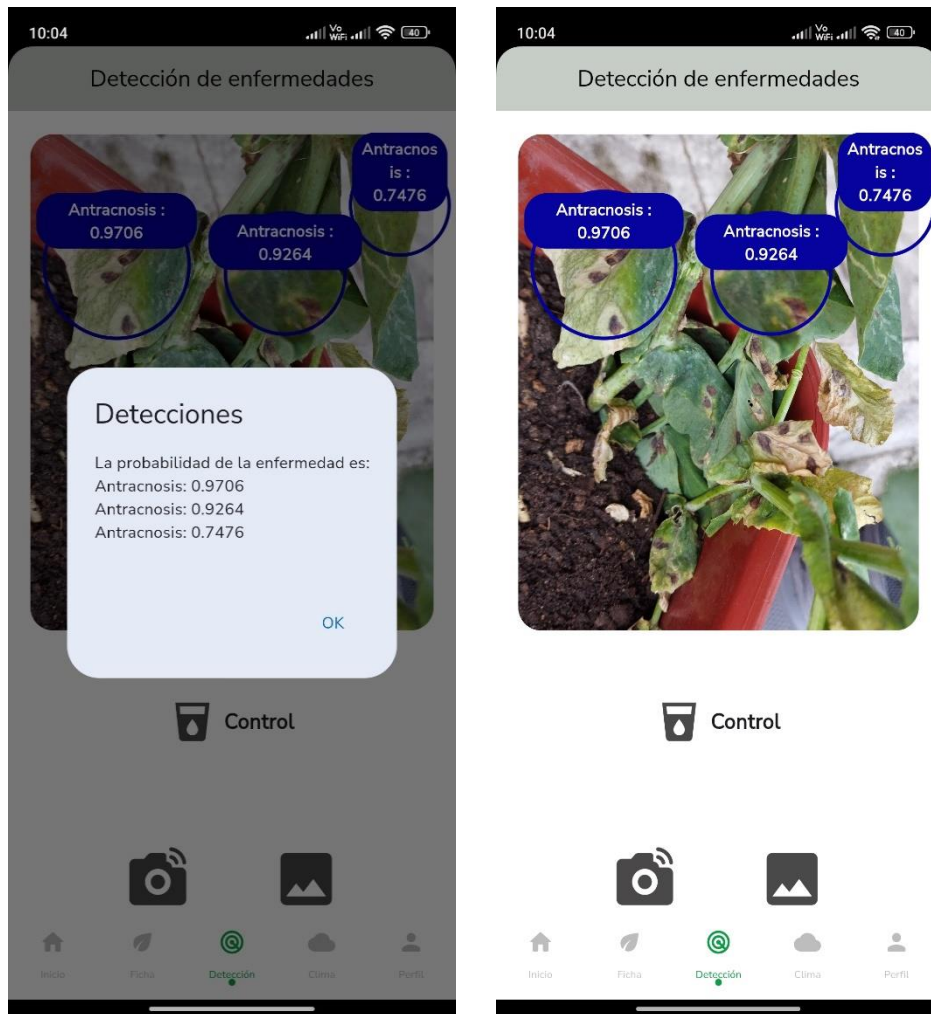


Figura 85. Modelo de detección.

4.2. DISCUSIÓN

Los objetivos generales y específicos desempeñaron un papel fundamental en la orientación para la realización del presente trabajo de investigación tomando en cuenta que, facilitaron el análisis juntamente con la descripción de cada resultado logrado, debido a que estos surgieron a partir de la sólida fundamentación teórica.

La utilización de apps móviles simplifica la detección de patologías, en las plantas al emplear la visión artificial junto con algoritmos como YOLO, lo que ayuda a los agricultores a mantener un control estricto sobre la salud de sus cultivos. En este contexto, Huanca (2022) propuso la elaboración de una app móvil para el análisis preliminar del oídio en plantas de uva. Esta propuesta se basó en una metodología de investigación cuantitativa y experimental, logrando una precisión del 99.00%. Además, Serrano y Torres (2020) presentaron un prototipo de aplicación móvil para identificar mazorcas de cacao afectadas por *Monilia* y/o *Fitóftora*, utilizando

tecnologías de visión por computadora y aprendizaje automático. Los resultados obtenidos mediante el modelo YOLOv4 arrojaron una precisión del 60% en la detección de las mazorcas de cacao. En contraste, con la presente investigación se realizó la elaboración de un aplicativo para la detección temprana de enfermedades en arveja Quantum haciendo el uso de una metodología mixta, explicativa, de campo, descriptiva y exploratoria. Mediante el uso del algoritmo YOLOv8, se alcanzó una precisión del 91.6%, lo que indica un rendimiento óptimo dentro de los rangos conocidos, con lo que se puede afirmar que el uso de la visión artificial ayuda al agricultor de manera inmediata.

Es factible realizar la detección de enfermedades mediante aplicaciones técnicas puesto que una aplicación móvil puede asistir al agricultor en la identificación de enfermedades a edad temprana, de igual manera el escaso uso de tecnologías como la agricultura de precisión la cual lleva de forma empírica la identificación de plagas, es decir se realiza la detección de forma tradicional. En efecto, la información antes mencionada es verificada por (Guerrero, 2021) quien destaca la importancia crucial del seguimiento de cultivos en parcelas en la agricultura de precisión, tomando en cuenta que esta actividad es fundamental para evaluar el rendimiento y prever la cosecha además la utilización de sistemas de visión artificial ayuda a la identificación de enfermedades en sembríos, para desarrollar el sistema de detección de enfermedades en los cultivos, se utilizó un conjunto de datos compuesto por 502 imágenes, que se sometieron a un extenso proceso de entrenamiento de 6000 épocas. Durante este proceso, el sistema logró una precisión media (mAP) del 81.60% en un conjunto de datos densamente poblado. Esto indica que el sistema alcanza una efectividad del 86% en la identificación de la enfermedad específica en cuestión. En contraste con Plant Vetch que utilizó un conjunto de 1320 imágenes con un total de 100 épocas de entrenamiento con una Precisión Media (mAP) de 86% con una efectividad del 91,6% en la detección.

En la realización de este procedimiento se ha tenido en consideración tanto la morfología como las dimensiones de la hoja. Asimismo, en el desarrollo de este proyecto, se han contemplado minuciosamente las distintas fases del proceso de visión artificial y tratamiento de imágenes involucran algoritmos diseñados para alterar la imagen antes de su análisis con la ayuda de YOLOv5, que concuerdan con los trabajos de (Morán y Quiranza, 2022) y (Gonzabay, 2023) quienes concuerdan con lo ya mencionado anteriormente, además menciona que el algoritmo YOLOv5

tiene un rendimiento esencial en el ámbito de la inteligencia artificial debido a su notable eficiencia y precisión. Este algoritmo se distingue por su capacidad para llevar a cabo la detección de objetos en imágenes y videos en tiempo real, por lo tanto, lo convierte en una herramienta idónea para proyectos de investigación. Su capacidad para analizar y reconocer objetos de manera rápida y precisa lo posiciona como una opción valiosa en aplicaciones que requieren un rendimiento óptimo, contribuyendo significativamente al progreso de la tecnología de visión por computadora y a la mejora de diversos campos de estudio.

Es de vital importancia la identificación de campos sanos e infectados por tal motivo se necesita de herramientas tecnológicas que brinden todo lo necesario para la detección, es por eso que el aplicativo propuesto se enfoca en la distinción de estos espacios gracias a la recopilación de imágenes tanto de hojas como de la planta para un mejor procesamiento que concuerda con el trabajo de (Cusme y Loor, 2019) quien obtuvieron un 97% de efectividad en la detección de la enfermedad en las hojas de café robusta. Las iteraciones ejecutadas proporcionaron un promedio del 93% de efectividad en la detección, y (Jácome, 2022). Que demostraron que los algoritmos y la detección innata en espacios sanos e infectados es gracias un buen procesamiento integrado a un aplicativo móvil, que obtuvo una precisión de validación de 96.875 %. A comparación del aplicativo Plant Vetch que cuenta con una precisión del 91,6% en la detección de enfermedades en arvejas Quantum.

Corroborando con (Issuu, 2019), las enfermedades más comunes en la arveja son oídio, fusarium y antracnosis con un porcentaje de 47%, 33% y 20% respectivamente, de igual manera se detectó las mismas enfermedades con el aplicativo en la comunidad La Delicia con un porcentaje de 71%, 56% y 53% lo que afirma que el desconocimiento de los factores climáticos además de la falta de herramientas tecnológicas en la agricultura genera un porcentaje de pérdidas para los agricultores que se dedican a la siembra de este guisante.

Con ayuda de la metodología de desarrollo de software ágil concretamente Extreme Programming (XP) se logró cumplir el objetivo principal del proyecto denominado Plant Vetch como apoyo en el proceso de cultivo para agricultores de la comunidad La Delicia, por lo que se llevó un desarrollo organizado el cual fue planificado mediante historias de usuario de cada una de las actividades para obtener resultados de calidad con recursos mínimos. Cada una de las fases fueron realizadas con éxito por lo tanto se hicieron pruebas de aceptación de cada una de las funcionalidades

del aplicativo, el cual coincide con (Morán & Quiranza, 2022) el cual realizó la elección de esta metodología dado que se basa en su eficacia y adaptabilidad, principalmente gracias a su flexibilidad temporal durante el desarrollo de la aplicación. Además de su eficiencia inherente, fomenta un entorno colaborativo en el cual los miembros del equipo pueden contribuir con criterios para cada tarea realizada. En otras palabras, la metodología se ajusta a los requerimientos específicos del usuario. Este enfoque destaca la interacción continua entre el cliente y el desarrollador, lo que contribuye a un proceso más dinámico y orientado hacia resultados satisfactorios.

V. CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

- La exploración de varias fuentes bibliográficas enfocadas con los distintos algoritmos de visión artificial contribuyó de manera significativa al proceso de elaboración de la aplicación para móviles destinada a la identificación de enfermedades en cultivos de arveja.
- Tras comparar los algoritmos MobileNetV2 y YOLOv8, se determinó que YOLOv8 ofrece una mayor precisión dado que cuenta con cuadros delimitadores en la detección aumentando su eficiencia.
- El entrenamiento del algoritmo en Google Colab demostró ser eficiente, ahorrando considerablemente tiempo en comparación con un equipo local, donde el proceso es notablemente más prolongado.
- El amplio conjunto de imágenes de arveja permitió el entrenamiento y la validación del aplicativo. puesto que a mayor cantidad de imágenes el aplicativo presenta mayor exactitud obteniendo resultados confiables y precisos.
- El desarrollo de la aplicación móvil resultó ser una herramienta altamente eficiente en el procedimiento para detectar enfermedades en los campos de cultivo de arvejas. Además, al ofrecer opciones de control específicas para cada enfermedad, la aplicación contribuye significativamente a la prevención y propagación de enfermedades en las etapas tempranas del cultivo de arvejas.

5.2. RECOMENDACIONES

- Es fundamental considerar la iluminación como uno de los factores influyentes en el momento de la captura de imágenes, dado que la calidad de la imagen depende directamente de ella. Una buena iluminación asegura una mejor precisión en el análisis y la identificación de posibles enfermedades.

- Para mejorar la precisión del algoritmo de detección de enfermedades en aplicaciones móviles, se aconseja tomar las fotografías a distancias cortas de 3 a 5 cm. Esto permite una mayor resolución y detalle, especialmente para enfermedades como el oídio, cuyos síntomas pueden ser sutiles y difíciles de identificar a distancia.
- Es esencial seguir con precisión las instrucciones proporcionadas por la aplicación para el uso de fungicidas, tanto para tratamiento curativo como preventivo. Esto incluye la dosificación adecuada, el momento óptimo de aplicación y la selección apropiada del fungicida según la enfermedad y la planta específica afectada.
- La implementación de arquitecturas de aprendizaje automático más avanzadas puede conducir a mejoras significativas en el rendimiento del modelo. Esto se traduce en una detección más precisa de enfermedades y una eficiencia mejorada en la operación general de la aplicación. Se recomienda explorar técnicas como redes neuronales convolucionales (CNN) o modelos de atención para aprovechar al máximo las capacidades del sistema y mejorar su capacidad de adaptación a diferentes condiciones y tipos de plantas.

VI. REFERENCIAS BIBLIOGRÁFICAS

- Agrositio. (2020, June 11). *El cultivo de arvejas enfrenta un gran desafío a nivel mundial*. <https://www.agrositio.com.ar/noticia/210684-el-cultivo-de-arvejas-enfrenta-un-gran-desafio-a-nivel-mundial.html>
- Almeida, F., Rodrigues, M. L., & Coelho, C. (2019). El problema aún subestimado de las enfermedades fúngicas en todo el mundo. *Frontiers in Microbiology*, 10(FEB). <https://doi.org/10.3389/FMICB.2019.00214>
- Angulo, A. (2019). *Identificación de las principales plagas y enfermedades en el cultivo de arveja (Pisum sativum), parroquia Bolívar, cantón Bolívar, Provincia del Carchi*. 1–45. <http://dspace.utb.edu.ec/bitstream/handle/49000/6395/E-UTB-FACIAGING%20AGRON-000152.pdf?sequence=1&isAllowed=y>
- Anitha, R., & Jayalakshmi, S. (2021). An Efficient Region Based Object Detection method using Deep learning Algorithm. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, 12(11), 5909–5916. <https://doi.org/10.17762/TURCOMAT.V12I11.6876>
- Bastis, C. (2020, June 1). *La Importancia de la Investigación Explicativa*. <https://online-tesis.com/la-importancia-de-la-investigacion-explicativa/>
- Borges, S. (2019, November 19). *¿Qué es PostgreSQL? - Para qué sirve, Características e Instalación*. <https://blog.infranetworking.com/servidor-postgresql/>
- Castellanos, L., Torrado, joe, & Céspedes, N. (2021). Alternativas biológicas para el control de *Fusarium oxysporum* en arveja en pamplona, Norte de Santander. *Revista de Investigación Agraria y Ambiental*, 12(1), 1–16. <https://doi.org/10.22490/21456453.3650>
- Castillo, A. (2019). Efecto de la aplicación de tres dosis de zeolita, en la producción del cultivo de arveja (*Pisum sativum* L.), en el sector de San José de Chaltura, Cantón Antonio Ante, Provincia de Imbabura. 1–60. <http://dspace.utb.edu.ec/bitstream/handle/49000/4378/TE-UTB-FACIAGING%20AGRON-000092.pdf?sequence=1&isAllowed=y>

- Catro, M. (2019, December 25). *Qué es y para qué sirve la metodología RAD*. <https://www.incentro.com/es-ES/blog/metodologia-rad-desarrollo-rapido-aplicaciones>
- Cherlinka, V. (2023, November 21). *Enfermedades De Las Plantas: Tipos, Causas Y Control*. <https://eos.com/es/blog/enfermedades-de-las-plantas/>
- Cordova, M. (2022a). *Precipitación solida simulada como factor meteorológico influyente en el rendimiento de la arveja (Pisum sativum) en Acobamba*. <https://repositorio.unh.edu.pe/handle/unh/5658>
- Cordova, M. (2022b). *Precipitación solida simulada como factor meteorológico influyente en el rendimiento de la arveja (pisum sativum) en acobamba*. <https://apirepositorio.unh.edu.pe/server/api/core/bitstreams/62e6da9f-4baa-4856-92d0-4f44eab44e7a/content>
- Cuauhtemoc, J. (2019, June). *Visión artificial en la agricultura y la ganadería*. https://www.researchgate.net/publication/325617522_Vision_artificial_en_la_agricultura_y_la_ganaderia
- Cusme, K., & Loor, A. (2019). *Aplicación móvil de detección y clasificación de "la roya" en hojas de café robusta mediante aprendizaje automático*. 1–76. <https://repositorio.esпам.edu.ec/bitstream/42000/1104/1/TTC21.pdf>
- Das, T. (2023, August 16). *Google Colab: todo lo que necesita saber*. <https://geekflare.com/es/google-colab/>
- Estrada, M., Núñez, J., Saltos, P., & Cunuhay, W. (2021). *Revisión Sistemática de la Metodología Scrum para el Desarrollo de Software* *Revisión Sistemática de la Metodología Scrum para el Desarrollo de Software* *Systematic review of the SCRUM methodology for software development* *Revisão Sistemática da Metodologia Scrum para Desenvolvimento de Software*. 7, 434–447. <https://doi.org/10.23857/dc.v7i4.2429>
- Garcés, A. (2022). *Sistema cuantificador de calidad de cultivo de manzana para monitoreo de la producción utilizando algoritmos de Aprendizaje Profundo con Visión Artificial y Segmentación de Instancias*. <https://repositorio.uta.edu.ec/bitstream/123456789/35319/1/t2020meat.pdf>
- García, M. (2020). *La encuesta*. <https://es.scribd.com/document/358292348/Garcia-Ferrando-M-La-Encuesta>

- Garzón, W. (2020). *Agricultura de precisión con drones*. <https://repository.ucc.edu.co/server/api/core/bitstreams/eabec72f-9ad4-43a5-90b1-9b8f9a070066/content>
- Gevara, G., Verdesoto, A., & Castro, N. (2020). *Metodologías de investigación educativa (descriptivas, experimentales, participativas, y de investigación-acción)*. 165–167. <https://dialnet.unirioja.es/servlet/articulo?codigo=7591592>
- Gonzabay, J. (2023). *Procesamiento de imágenes para la clasificación de granos de café por forma y color*. <http://repositorio.utmachala.edu.ec/bitstream/48000/21622/1/ESTEBAN%20GONZABAY%20JIMENEZ.pdf>
- González, P., & Pacheco, R. (2022). *Detección de defectos en frutas usando modelos de CNN con datos reales y virtuales*. 14–23. <https://www.dspace.espol.edu.ec/bitstream/123456789/57041/1/T-112972%20Gonz%20a11ez%20-%20Pacheco.pdf>
- Guerrero, C. (2021). *Reconocimiento de lesiones necróticas para la detección temprana de la plaga thrips (kakothrips robustus uzell) en los cultivos del guisante o arveja mediante técnicas deep learning*. 1–161. <http://repositorio.utn.edu.ec/bitstream/123456789/11654/2/PG%20909%20TRABAJO%20GRADO.pdf>
- Guerrero, E. (2023). *Evaluación de dosis de NPK-Ca en la producción de arveja (Pisum sativum var. Quantum)*. <https://repositorio.uta.edu.ec/handle/123456789/38233>
- Guerrero, O. (2023). *Evaluación de dosis de NPK-Ca en la producción de arveja (Pisum sativum var. Quantum)*. <https://repositorio.uta.edu.ec/bitstream/123456789/38233/1/034%20Agronom%20ada%20-%20Guerrero%20Guano%20Ori%20Efr%20a9n.pdf>
- Huanca, A. (2022). *Aplicación móvil para el diagnóstico preliminar del oídio en la uva a través de imágenes digitales*. 8–62. https://repositorio.ucv.edu.pe/bitstream/handle/20.500.12692/81826/Huanca_VAC-SD.pdf?sequence=8&isAllowed=y
- Issuu. (2019, July 17). *Principales plagas y enfermedades en el cultivo de arveja en la provincia del Carchi*. 2019. https://issuu.com/praxisconsulting5/docs/identificaci_n_de_las_principales_p

- Jácome, J. (2022). *Detección temprana de minador, mosca blanca y fusarium en el tomate riñón, aplicando técnicas de visión artificial y machine learning*. <http://dspace.unach.edu.ec/handle/51000/9102>
- Layedra, N., Ramos, M., Salazar, S., & Baldeón, B. (2022). *Análisis de los lenguajes de programación más utilizados en el desarrollo de aplicaciones web y móviles*. 8(3), 1601–1625. <https://doi.org/10.23857/dc.v8i3>
- Lázaro, R. (2021). *ENTREVISTAS ESTRUCTURADAS, SEMIESTRUCTURADAS Y LIBRES. ANÁLISIS DE CONTENIDO*. <https://ruidera.uclm.es/server/api/core/bitstreams/fdf77886-6075-453a-b7cc-731232b56e77/content>
- Marín, R. (2019, April 16). *Los gestores de bases de datos (SGBD) más usados*. <https://www.inesem.es/revistadigital/informatica-y-tics/los-gestores-de-bases-de-datos-mas-usados/>
- Martínez, I. (2020). *Aprendizaje automático aplicado a la regulación inteligente del tráfico en ciudad*. <https://ruidera.uclm.es/server/api/core/bitstreams/15dede1f-bb4a-4677-b3e3-d198021b677d/content>
- Mesh, J. (2020, March 29). *Metodología Kanban: revoluciona tu manera de trabajar más ágil*. <https://blog.trello.com/es/metodologia-kanban>
- Móran, D., & Quiranza, A. (2022). *Visión Artificial para la detección de enfermedades en el cultivo de papa*. <http://repositorio.upec.edu.ec/bitstream/123456789/1705/1/072-%20MOR%c3%81N%20DANIELA%20-%20QUIRANZA%20ANTHONY.pdf>
- Morán, D., & Quiranza, A. (2022). *Visión Artificial para la detección de enfermedades en el cultivo de papa*. <http://repositorio.upec.edu.ec/bitstream/123456789/1705/1/072-%20MOR%c3%81N%20DANIELA%20-%20QUIRANZA%20ANTHONY.pdf>
- Moreno, V. (2019). *Determinación de la rentabilidad del cultivo de arveja (Pisum sativum L.) en la comunidad El Capulí, parroquia San José, cantón Montufar, provincia del Carchi, 2019*. 1–19. <http://dspace.utb.edu.ec/bitstream/handle/49000/6461/E-UTB-FACIAG-ING%20AGRON-000191.pdf?sequence=1&isAllowed=y>
- Palomo, E. (2022, March 10). *Plagas y enfermedades en leguminosas grano, ¿qué podemos hacer?* <https://www.empresaagraria.com/plagas-enfermedades-leguminosas-grano-podemos/>

- Parada, M. (2019, November 23). Qué es SQL Server. <https://openwebinars.net/blog/que-es-sql-server/>
- Polo, J. (2024, February 20). YOLO, así es el líder en detección de objetos y segmentación de imágenes. <https://www.whatsnew.com/2024/02/20/yolo-asi-es-el-lider-en-deteccion-de-objetos-y-segmentacion-de-imagenes/>
- Quesada, A., & Medina, C. (2020). MÉTODOS TEÓRICOS DE INVESTIGACIÓN: ANÁLISIS-SÍNTESIS, INDUCCIÓN-DEDUCCIÓN, ABSTRACTO-CONCRETO E HISTÓRICO-LÓGICO. <http://monografias.umcc.cu/monos/2020/Inglnd/mo2076.pdf>
- Raeburn, A. (2022, November 28). La programación extrema (XP) produce resultados, pero ¿es la metodología adecuada para ti? <https://asana.com/es/resources/extreme-programming-xp>
- Rivas, I., Corona, V., Gutiérrez, J., & Hernández, L. (2019). Metodologías actuales de desarrollo de software. In *Artículo Revista Tecnología e Innovación Diciembre* (Vol. 2, Issue 5). www.ecorfan.org/bolivia
- Robledano, A. (2019, September 24). Qué es MySQL: Características y ventajas. <https://openwebinars.net/blog/que-es-mysql/>
- Roldán, B., Roshan, R., & Sánchez, E. (2019). Detección de enfermedades en el sector agrícola utilizando Inteligencia Artificial Detection of Diseases in the Agricultural Sector using Artificial Intelligence. In *Research in Computing Science* (Vol. 148, Issue 7). https://www.rcs.cic.ipn.mx/2019_148_7/Deteccion%20de%20enfermedades%20en%20el%20sector%20agricola%20utilizando%20Inteligencia%20Artificial.pdf
- Sáenz, K. (2020). Principales enfermedades en el cultivo de arveja (*Pisum sativum* L.) en la Facultad de Ciencias Agropecuarias. <http://dspace.utb.edu.ec/bitstream/handle/49000/7971/E-UTB-FACIAG-ING%20AGRON-000216.pdf?sequence=1&isAllowed=y>
- Salas, D. (2019, August 20). Investigación-acción - Investigalia. <https://investigaliacr.com/investigacion/investigacion-accion/>
- Serrano, J., & Torres, C. (2020). Prototipo de aplicación móvil para la identificación de mazorcas de cacao enfermas haciendo uso de visión por computadora y aprendizaje de máquina. 10-70. https://repository.unab.edu.co/bitstream/handle/20.500.12749/13367/2020_Tesis_Juan_Sebastian_Serrano_Arenas.pdf?sequence=1&isAllowed=y

- Ticona, H. (2022). *Desarrollo de aplicativos webs, beneficios y diferencias al utilizar un framework o un cms (content management system)*. 1–51. <https://repositorio.unam.edu.pe/server/api/core/bitstreams/a2deb9a3-d5e8-4067-b31b-f305c64bf8e0/content>
- Torrado, J. (2019). *Enfermedades causadas por microorganismos del suelo en el cultivo de arveja (Pisum sativum) en las condiciones de la vereda Monteadentro del municipio Pamplona, como alternativas biológicas de control*. 1–90. http://repositoriodspace.unipamplona.edu.co/jspui/bitstream/20.500.12744/1838/1/Torrado_2017_TG.pdf
- Velázquez, A. (2019, September 3). *¿Cuál es la diferencia entre población y muestra?* <https://www.questionpro.com/blog/es/diferencia-entre-poblacion-y-muestra/>
- Villafuerte, J., & Tóala, J. (2020). *El método deductivo para la comprensión de la lectoescritura en los estudiantes del cuarto año de Educación General Básica de la escuela Alejandro Humboldt del cantón Jipijapa*. 1–73. <http://repositorio.utm.edu.ec:3000/server/api/core/bitstreams/df95fc38-80e8-4bcd-9d22-6b7fca0874b9/content>
- Villota, P. (2019). *Implementación de una estación prototipo con visión artificial, aplicado a la agricultura de precisión*. 1–99. <http://dspace.ucuenca.edu.ec/bitstream/123456789/33492/1/Trabajo%20de%20Titulaci%C3%B3n.pdf>
- Vizcaíno, P., Maldonado, I., & Cedeño, R. (2023). Metodología de la investigación científica: guía práctica. *Ciencia Latina Revista Científica Multidisciplinar*, 7(4), 9723–9762. https://doi.org/10.37811/cl_rcm.v7i4.7658
- Waldimar, R. (2022). *Evaluación de trichoderma (trichoderma harzianum) y microorganismos eficientes como antagonistas del fusarium (fusarium spp.) en la fase vegetativa del cultivo de frijol (phaseolus vulgaris l.) variedad canario, distrito de abancay – apurímac*. <https://repositorio.utea.edu.pe/bitstream/utea/361/1/Evaluaci%C3%B3n%20de%20Trichoderma%20%28Trichoderma%20harzianum%29%20y%20microorganismos%20eficientes%20como%20antagonistas%20del%20fusarium%20%28Fusarium%20spp.%29.pdf>
- Zorrila, A. (2021, January 8). *¿Cómo se realiza una investigación documental o bibliográfica?* <https://campusidyd.com/investigacion-documental-o-bibliografica/>

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

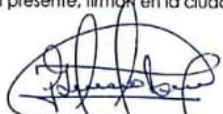
DE LA SUSTENTACIÓN ORAL DE LA PREDENSA DEL TRABAJO DE INTEGRACIÓN CURRICULAR

ESTUDIANTE:	TUTACHA DELGADO WILMER ORLANDO	CÉDULA DE IDENTIDAD:	0401933031
PERIODO ACADÉMICO:	2024A		
PRESIDENTE TRIBUNAL	MSC. GEORGINA GUADALUPE ARCOS PONCE	DOCENTE TUTOR:	MSC. CARLITOS ALBERTO GUANO CÁRDENAS
DOCENTE:	MSC. MARCO ANTONIO YANDÚN VELASTEGUI		
TEMA DEL TIC:	"Visión artificial en la agricultura de precisión para la identificación de enfermedades en cultivos de arveja Quantum"		
No.	CATEGORÍA	Evaluación cuantitativa	OBSERVACIONES Y RECOMENDACIONES
1	PROBLEMA - OBJETIVOS	9,00	Mejorar redacción del problema
2	FUNDAMENTACIÓN TEÓRICA	9,00	Matriz de correlación entre enfermedad y causas
3	METODOLOGÍA	9,00	
4	RESULTADOS	9,00	
5	DISCUSIÓN	9,00	Mejorar la precisión en las enfermedades (aplicativo)
6	CONCLUSIONES Y RECOMENDACIONES	9,00	Mejorar la redacción de conclusiones y recomendaciones, muchas de ellas ya están aplicando
7	DEFENSA, ARGUMENTACIÓN Y VOCABULARIO PROFESIONAL	9,00	
8	FORMATO, ORGANIZACIÓN Y CALIDAD DE LA INFORMACIÓN	9,00	Revisar fechas, revisión normas APA, no dejar títulos sueltos

Obteniendo una nota de: **9,00** Por lo tanto, **APRUEBA** ; debiendo el o los investigadores acatar el siguiente artículo:

Art. 36.- De los estudiantes que aprueban el informe final del TIC con observaciones.- Los estudiantes tendrán el plazo de 10 días para proceder a corregir su informe final del TIC de conformidad a las observaciones y recomendaciones realizadas por los miembros del Tribunal de sustentación de la pre-defensa.

Para constancia del presente, firman en la ciudad de Tulcán el **lunes, 20 de mayo de 2024**


MSC. GEORGINA GUADALUPE ARCOS PONCE
PRESIDENTE TRIBUNAL


MSC. CARLITOS ALBERTO GUANO CÁRDENAS
DOCENTE TUTOR


MSC. MARCO ANTONIO YANDÚN VELASTEGUI
DOCENTE



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


ESTUDIANTE:	PÉREZ ENRÍQUEZ KIMBERLHY NAYELLY	CÉDULA DE IDENTIDAD:	0401677901
PERIODO ACADÉMICO:	2024A		
PRESIDENTE TRIBUNAL	MSC. GEORGINA GUADALUPE ARCOS PONCE	DOCENTE TUTOR:	MSC. CARLITOS ALBERTO GUANO CÁRDENAS
DOCENTE:	MSC. MARCO ANTONIO YANDÚN VELASTEGUI		
TEMA DEL TIC:	"Visión artificial en la agricultura de precisión para la identificación de enfermedades en cultivos de arveja Quantum"		
No.	CATEGORÍA	Evaluación cuantitativa	OBSERVACIONES Y RECOMENDACIONES
1	PROBLEMA - OBJETIVOS	9,00	Mejorar redacción del problema
2	FUNDAMENTACIÓN TEÓRICA	9,00	Matriz de correlación entre enfermedad y causas
3	METODOLOGÍA	9,00	
4	RESULTADOS	9,00	
5	DISCUSIÓN	9,00	Mejorar la precisión en las enfermedades (aplicativo)
6	CONCLUSIONES Y RECOMENDACIONES	9,00	Mejorar la redacción de conclusiones y recomendaciones, muchas de ellas ya están aplicando
7	DEFENSA, ARGUMENTACIÓN Y VOCABULARIO PROFESIONAL	9,00	
8	FORMATO, ORGANIZACIÓN Y CALIDAD DE LA INFORMACIÓN	9,00	Revisar fechas, revisión normas APA, no dejar títulos sueltos

Obteniendo una nota de: **9,00** Por lo tanto, **APRUEBA** ; debiendo el o los investigadores acatar el siguiente artículo:

Art. 36.- De los estudiantes que aprueban el informe final del TIC con observaciones.- Los estudiantes tendrán el plazo de 10 días para proceder a corregir su informe final del TIC de conformidad a las observaciones y recomendaciones realizadas por los miembros del Tribunal de sustentación de la pre-defensa.

Para constancia del presente, firman en la ciudad de Tulcán el **lunes, 20 de mayo de 2024**


MSC. GEORGINA GUADALUPE ARCOS PONCE
PRESIDENTE TRIBUNAL


MSC. CARLITOS ALBERTO GUANO CÁRDENAS
DOCENTE TUTOR


MSC. MARCO ANTONIO YANDÚN VELASTEGUI
DOCENTE

Anexo 2. Certificado del abstract por parte de idiomas



**UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI
FOREIGN AND NATIVE LANGUAGE CENTER**

ABSTRACT- EVALUATION SHEET				
NAME: Pérez Enríquez Kimberthy Nayelly y Tutachá Delgado Wilmer Orlando				
DATE: 27 de mayo de 2024				
Topic: "Visión artificial en la agricultura de precisión para la identificación de enfermedades en cultivos de arveja Quantum"				
MARKS AWARDED QUANTITATIVE AND QUALITATIVE				
VOCABULARY AND WORD USE	Use new learnt vocabulary and precise words related to the topic	Use a little new vocabulary and some appropriate words related to the topic	Use basic vocabulary and simplistic words related to the topic	Limited vocabulary and inadequate words related to the topic
	EXCELLENT: 2 <input checked="" type="checkbox"/>	GOOD: 1 Vera Játiva Edwin Andrés,5 <input checked="" type="checkbox"/>	AVERAGE: 1 <input type="checkbox"/>	LIMITED: 0,5 <input type="checkbox"/>
WRITING COHESION	Clear and logical progression of Ideas and supporting paragraphs.	Adequate progression of Ideas and supporting paragraphs.	Some progression of ideas and supporting paragraphs.	Inadequate Ideas and supporting paragraphs.
	EXCELLENT: 2 <input checked="" type="checkbox"/>	GOOD: 1,5 <input checked="" type="checkbox"/>	AVERAGE: 1 <input type="checkbox"/>	LIMITED: 0,5 <input type="checkbox"/>
ARGUMENT	The message has been communicated very well and identify the type of text	The message has been communicated appropriately and identify the type of text	Some of the message has been communicated and the type of text is little confusing	The message hasn't been communicated and the type of text is inadequate
	EXCELLENT: 2 <input checked="" type="checkbox"/>	GOOD: 1,5 <input checked="" type="checkbox"/>	AVERAGE: 1 <input type="checkbox"/>	LIMITED: 0,5 <input type="checkbox"/>
CREATIVITY	Outstanding flow of ideas and events	Good flow of Ideas and events	Average flow of Ideas and events	Poor flow of Ideas and events
	EXCELLENT: 2 <input checked="" type="checkbox"/>	GOOD: 1,5 <input checked="" type="checkbox"/>	AVERAGE: 1 <input type="checkbox"/>	LIMITED: 0,5 <input type="checkbox"/>
SCIENTIFIC SUSTAINABILITY	Reasonable, specific and supportable opinion or thesis statement	Minor errors when supporting the thesis statement	Some errors when supporting the thesis statement	Lots of errors when supporting the thesis statement
	EXCELLENT: 2 <input checked="" type="checkbox"/>	GOOD: 1,5 <input checked="" type="checkbox"/>	AVERAGE: 1 <input type="checkbox"/>	LIMITED: 0,5 <input type="checkbox"/>
TOTAL/AVERAGE	9 - 10: EXCELLENT 7 - 8,9: GOOD 5 - 6,9: AVERAGE 0 - 4,9: LIMITED		TOTAL 9	



**UNIVERSIDAD POLITÉCNICA ESTATAL DEL
CARCHI FOREIGN AND NATIVE LANGUAGE
CENTER**

Informe sobre el Abstract de Artículo Científico o Investigación.

Autor: Pérez Enríquez Kimberlhy Nayelly y Tutachá Delgado Wilmer Orlando

Fecha de recepción del abstract: 27 de mayo de 2024

Fecha de entrega del informe: 27 de mayo de 2024

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 los rubrics de evaluación de la traducción en Inglés, ésta alcanza un valor de 9, por lo cual se validó dicho trabajo.

Atentamente



Ing. Edison Peñañiel Arcos MSc
Coordinador del CIDEN

Anexo 3. Informe de Turniting

Tesis Perez- Tutachá.pdf

INFORME DE ORIGINALIDAD

5%	5%	0%	%
INDICE DE SIMILITUD	FUENTES DE INTERNET	PUBLICACIONES	TRABAJOS DEL ESTUDIANTE

FUENTES PRIMARIAS

1	repositorio.upec.edu.ec Fuente de Internet	2%
2	hdl.handle.net Fuente de Internet	<1%
3	investigacion.uva.es Fuente de Internet	<1%
4	repository.unab.edu.co Fuente de Internet	<1%
5	repositorio.utn.edu.ec Fuente de Internet	<1%
6	repositorio.espam.edu.ec Fuente de Internet	<1%
7	dspace.unach.edu.ec Fuente de Internet	<1%
8	www.publicacionesupec.org Fuente de Internet	<1%
9	lajc.epn.edu.ec Fuente de Internet	<1%

Anexo 4. Encuesta dirigida a agricultores

UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI
FACULTA DE INDUSTRIAS AGROPECUARIAS Y CIENCIAS AMBIENTALES
CARRERA DE COMPUTACIÓN

NOMBRE: _____ **GENERO:** F___ M___
FECHA: _____

Esta encuesta busca saber qué tan inmersos están los agricultores en el uso de la tecnología con respecto al control de enfermedades en la arveja y cómo se aplica.

Instrucciones: Lea cuidadosamente cada una de las preguntas y responda de acuerdo a su conocimiento como agricultor.

Preguntas

1. ¿Cuántos años de experiencia tiene usted como agricultor?

- () De 10 a 20 años
- () De 21 a 30 años
- () Mas de 31 años

2. ¿Cuántas hectáreas destina usted para el cultivo de arveja?

3. ¿Cuáles son las variedades de arveja que usted siembra en la zona?

- () Sindamanoy
- () Andina
- () San Isidro
- () Quantum
- () Otro -----

4. ¿Cuántas veces en el año siembra ALVERJA?

- () 1
- () 2

5. Seleccione la o las enfermedades más frecuentes que se presentan en el cultivo de arveja.

- () Antracnosis
- () Fusarium
- () Oídio
- () Otro -----

6. Seleccione la o las plagas más frecuentes que se presentan en el cultivo de arveja

- Minador
- Barrenador
- Trips
- Otro -----

7. ¿Con qué frecuencia realiza el monitoreo de plagas y enfermedades en su cultivo de arveja?

- Diariamente
- Cada 8 días
- Cada 15 días
- Cada mes

8. ¿Considera usted necesario ayuda técnica de algún agrónomo para el manejo de plagas y enfermedades en el cultivo de arveja?

- Si
- No
- ¿Por qué? -----

9. ¿En base a su experiencia cómo identifica una enfermedad en el cultivo de arveja?

- Mira cómo está el tallo
- Mira cómo está la hoja
- Mira el cómo está el suelo
- Mira que el cultivo está disminuyendo en el crecimiento
- Otros _____

10. ¿Qué recomendaría para obtener un cultivo libre de plagas y enfermedades?

- Fertilización adecuada
- Aplicación de agroquímicos
- Buscar un técnico agrónomo
- Otros _____

11. Usted como agricultor ¿cree que el uso de fungicidas funciona para controlar enfermedades en el cultivo?

- Pocas veces funciona
- Si funciona
- No funciona
- No funciona en todo el cultivo

12. ¿En caso de existir una aplicación para su teléfono celular que identifique enfermedades, la utilizaría?

- Sí
- No
- Talvez

13. ¿Cree usted que el uso de herramientas tecnológicas ayudaría a detectar enfermedades en el cultivo de arveja?

- Sí
- No
- Talvez

14. ¿Cuál de las siguientes opciones debería tener la aplicación para identificar enfermedades de su interés?

- Confiable
- Gratuita
- Fácil de usar

15. ¿Con el uso de la visión artificial usted cree que se puede mejorar la calidad y rendimiento de nuestros cultivos?

- Sí
- No
- ¿Por qué? -----

Anexo 5. Entrevista dirigida a agrónomos

FICHA DE ENTREVISTA

Datos del entrevistado

Apellido		Nombre
Sexo	Estado civil	
Movilidad de su lugar de residencia		

Datos del entrevistador

Apellido	Nombre
----------	--------

Datos de la entrevista

Lugar	Fecha
Hora de inicio	Hora de finalización
Observaciones	

GUIA DE ENTREVISTA

La siguiente entrevista tiene como objetivo obtener información verídica que aporta significativamente al desarrollo de la investigación.

Entrevista

1. ¿Cómo se monitorean y controlan las enfermedades en el cultivo actualmente?

2. ¿Cuáles son las condiciones de desarrollo de hongos en el cultivo de arveja?

3. ¿En qué momento se debe detectar una enfermedad en el cultivo sin afectar toda la cosecha?

4. ¿Cuáles son las medidas clave para evitar la propagación de enfermedades y lograr un cultivo exitoso?

5. ¿Podría una aplicación mejorar la calidad nutricional de la arveja en el cultivo?

6. ¿Apoya un prototipo para la detección temprana de enfermedades en la arveja?

Anexo 6. Registro fotográfico



Figura 86. Verificación de enfermedades en el cultivo de arveja



Figura 87. Demostración del software a los agricultores



Figura 88. Detección de las enfermedades en tiempo real

PLAN DE DESARROLLO DE SOFTWARE V1.0

**Visión Artificial en la agricultura de precisión para la identificación de enfermedades
en cultivo de arveja Quantum**

Generado por

Pérez Enríquez Kimberlhy Nayelly

Tutachá Delgado Wilmer Orlando

GENERALIDADES DEL PROYECTO

Descripción del proyecto

El proyecto titulado "Visión Artificial en la Agricultura de Precisión para la Identificación de Enfermedades en el Cultivo de Arveja Quantum" tiene como propósito principal el desarrollo de una herramienta tecnológica con visión artificial, diseñada para detectar enfermedades en las etapas iniciales del cultivo de arvejas. Esta iniciativa se llevará a cabo específicamente en la comunidad La Delicia, situada en la ciudad de San Gabriel.

Propósito

El propósito fundamental de este documento es proporcionar una visión general acerca de la organización y gestión del proyecto, siendo un componente esencial de la documentación requerida para la implementación exitosa de la metodología XP.

Historial de revisiones

Tabla 60. Historial revisiones

Fecha	Versión	Descripción	Autores	Aprobado por
	1.0	Evaluación y validación del proyecto.	Kimberlhy Pérez Wilmer Tutachá	

Alcance

Formular una propuesta que emplee herramientas de programación y tecnologías con el propósito de fomentar y facilitar la interacción del usuario. Este objetivo proporciona una orientación fundamental para el equipo de investigación durante la ejecución del proyecto.

Objetivos

- Buscar información pertinente para el desarrollo del proyecto, considerando las herramientas proporcionadas por la metodología XP.
- Elegir herramientas de programación adecuadas para desarrollar una herramienta de visión artificial que identifique enfermedades en cultivos de arvejas.
- Establecer los estándares de metodología de desarrollo de software con el fin de asegurar la adecuada operatividad del proyecto
- Desarrollar una herramienta tecnológica eficiente para identificar enfermedades en etapas tempranas de los cultivos de arvejas.

Suposiciones y restricciones

Tomando en cuenta las entrevistas y encuestas realizadas se muestra las siguientes restricciones:

- La herramienta tecnológica debe tener la capacidad de detectar las principales enfermedades en las arvejas durante las primeras etapas de su crecimiento, que incluyen Antracnosis, Fusarium y Oídio.
- La herramienta tecnológica debe contar con una interfaz intuitiva y comprensible que facilite la interacción del agricultor.

Evolución del plan de desarrollo de software

Este plan de desarrollo será revisado solo si se realiza una modificación significativa en la aplicación.

ORGANIZACIÓN DEL PROYECTO

Recursos humanos y profesionales

Tabla 61. Recursos humanos y profesionales

Nombre	Información de contacto	Celular
Kimberlhy Nayelly Pérez Enríquez	Kimberlhy.perez@upec.edu.ec	0988403058
Wilmer Orlando Tutachá Delgado	Wilmer.tutacha@upec.edu.ec	0968257384

Roles y responsabilidades

Tabla 62. Roles y responsabilidades

Rol	Responsabilidad	Asumido por
Consultor	Establece conocimientos a los investigadores en la solución de problemas.	Msc. Carlitos Guano
Programadores	Encargados de desarrollar la codificación del aplicativo.	Kimberlhy Pérez Wilmer Tutachá
Cliente	Encargado de brindar información pertinente al proyecto.	Agricultores

GESTION DEL PROYECTO

Plan del proyecto

Tabla 63. Plan del proyecto

Fase	Iteraciones	Duración
Fase de planificación	2	2 meses
Fase de diseño	2	4 meses
Fase de codificación	2	6 meses
Fase de pruebas	2	8 meses

Fases y líneas base.

Tabla 64. Fases y líneas base

Fase	Descripción	Objetivos del ciclo de vida
Planificación	Se recopila información específica esencial para el desarrollo del proyecto, junto con la selección de herramientas que serán utilizadas en las fases subsiguientes para avanzar en su ejecución.	Recopilar información directa cliente consultor para la elaboración del proyecto.
Diseño	Se diseña el maquetado del proyecto tomando en cuenta lo necesario del aplicativo con el fin de obtener un modelo base.	Diseñar maquetado que realice los requerimientos de la fase de planificación
Codificación	Se desarrolla el proceso de la programación en parejas con la finalidad de que se pueda resolver cualquier error en el entorno de trabajo	Presentar los componentes de código que realicen las funciones necesarias en el aplicativo en desarrollo.
Pruebas	Se ejecutan pruebas para el desarrollo del proyecto, verificando que el agricultor sea capaz de realizar sus propias pruebas y validar su función del aplicativo.	Ejecutar pruebas del correcto funcionamiento del aplicativo, considerando igual las pruebas de aceptación.

Objetivos y riesgos de cada fase

Tabla 65. Objetivos y riesgos de cada fase

Fase	Objetivo asociado	Riesgos contemplados
Planificación	Recopilación de información para el desarrollo de requerimientos del aplicativo.	La recopilación de información es insuficiente y no cumple con los requisitos establecidos. Falta de claridad en la búsqueda de información.
Diseño	Crear un maquetado de diseño acorde a los requerimientos.	El diseño de interfaz no cumple con las necesidades del proyecto. Desacuerdo de los agricultores con el diseño.

Codificación	Generar el código base del aplicativo para verificar las funcionalidades del proyecto	Falta de experiencia en el uso de herramientas de programación. Errores en la ejecución del código.
Pruebas	Generar pruebas de las funcionalidades y pruebas de aceptación del aplicativo	Dificultades en el funcionamiento del aplicativo. Poco interés y apoyo por parte del agricultor.

Diagrama de Gantt

Tabla 66. Diagrama de Gantt

Fase	Fecha de inicio	Duración en meses	Fecha fin
Planificación	30/06/2022	2	03/08/2022
Diseño	12/08/2022	4	09/12/2022
Codificación	12/12/2022	6	15/06/2023
Pruebas	19/06/2023	8	12/02/2024

Calendario del proyecto

Tabla 67. Calendario del proyecto

Fases	Meses																				
	J	J	A	S	O	N	D	E	F	M	A	M	J	J	A	S	O	N	D	E	F
Planificación	■	■																			
Diseño			■	■	■	■	■	■													
Codificación							■	■	■	■	■	■	■	■							
Pruebas															■	■	■	■	■	■	■

Control de desviación a la planificación

Tabla 68. Control de desviación a la planificación

Ítem	Probabilidad	Impacto
1	Altamente probable	Bajo
2	Bastante probable	Moderado
3	Posiblemente	Medio
4	Poco probable	Alto
5	Altamente improbable	Muy alto

Matriz de riesgo

Tabla 69. Matriz de riesgo

Ítem	Riesgo	Probabilidad	Impacto	Valor de riesgo
R1	La recopilación de información es insuficiente y no cumple con los requisitos establecidos.	2	2	4
R2	Falta de claridad en la búsqueda de información.	2	2	4
R3	El diseño de interfaz no cumple con las	1	1	1

R4	necesidades del proyecto. Desacuerdo de los agricultores con el diseño.	3	4	12
R5	Falta de experiencia en el uso de herramientas de programación.	3	5	15
R6	Errores en la ejecución del código.	3	5	15
R7	Dificultades en el funcionamiento del aplicativo.	3	4	12
R8	Poco interés y apoyo por parte del agricultor.	2	2	4

Tabla 70. Matriz de calor

Matriz de calor

	Riesgo alto
	Riesgo moderado
	Riesgo bajo

		MATRIZ DE RIESGO				
		PROBABILIDAD				
Impacto		Altamente improbable	Poco probable	Posiblemente	Bastante probable	Altamente probable
		1	2	3	4	5
Bajo	1	R1	R8			
Moderado	2	R2		R7		
Medio	3	R3	R4			R6
Alto	4				R5	
Muy alto	5					

Manual del Usuario para la Aplicación Móvil de Identificación de Enfermedades en Arvejas



PLANT VETCH

QUANTUM

Autores:

Pérez Enríquez Kimberlhy Nayelly

Tutachá Delgado Wilmer Orlando

Introducción

Este manual se ha diseñado con el propósito de facilitar el uso eficiente de la aplicación. Incorpora capturas de pantalla de fácil comprensión que mejoran la claridad de la navegación y las funciones disponibles.

Requisitos de software

Para asegurar el óptimo desempeño de la aplicación, es esencial cumplir con los siguientes requisitos de software mínimos en el dispositivo móvil:

Es necesario tener un sistema operativo Android 8 o una versión más reciente.

Requisitos de hardware

Para asegurar un rendimiento óptimo de la aplicación, es necesario que el dispositivo móvil cumpla con los siguientes requisitos de hardware mínimos:

- Memoria RAM de 2 GB o superior.
- Resolución de pantalla de 720 x 1520 o superior.
- Cámara funcional de 13 MP o superior.
- Conexión a internet de al menos 3 Mbps o una conexión 3G o superior.

Funcionamiento de la aplicación

En tu dispositivo móvil, ubicarás el ícono identificado como " Plant Vetch". Para acceder a la aplicación, simplemente deberás seleccionarlo.



Figura 89. Icono de la aplicación

Posteriormente, la pantalla presentará una interfaz de inicio diseñada para que el usuario pueda registrarse. Este proceso se realiza mediante su dirección de correo electrónico y contraseña

Login



Figura 90. Inicio de sesión

Si has olvidado tu contraseña, también dispones de un botón que te facilitará el proceso de restablecimiento de esta.

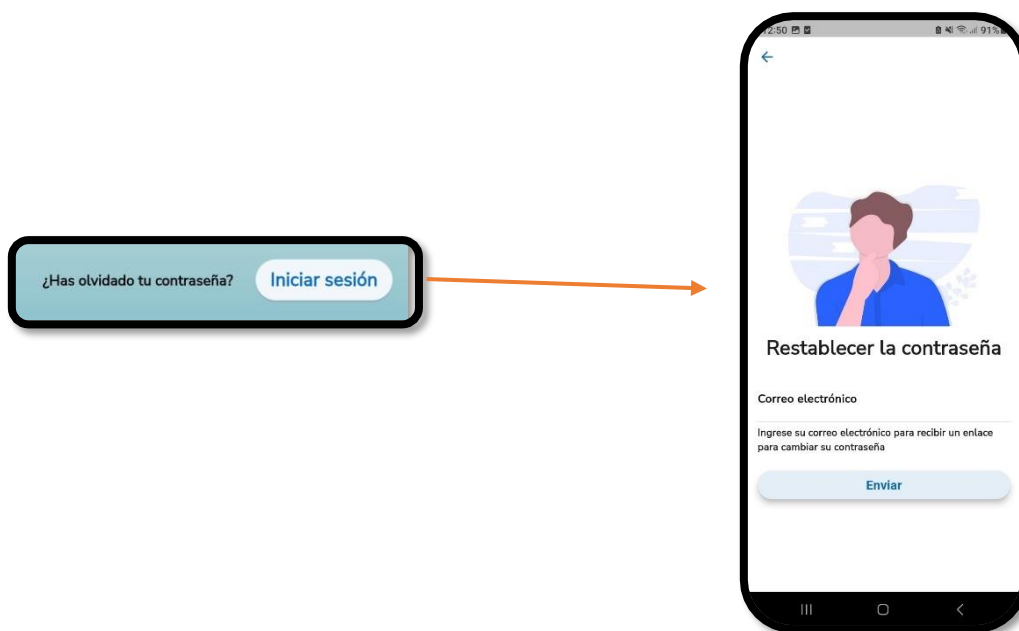


Figura 91. Restablecer contraseña

En esta sección, tienes la opción de registrarte mediante tu cuenta de Google para acceder directamente a la aplicación.

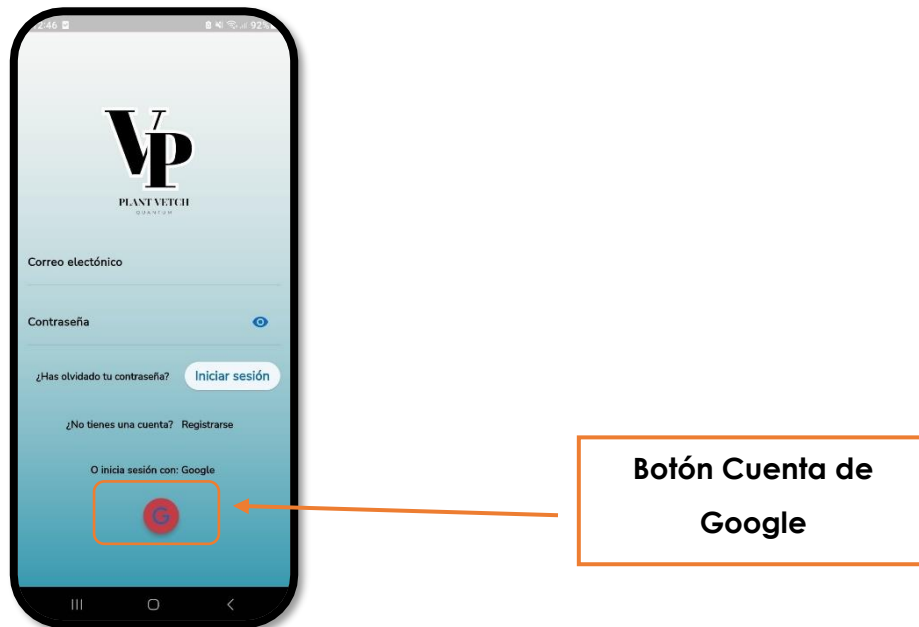


Figura 92. Registrarse con Google

En esta sección, puedes acceder utilizando tu cuenta con dos opciones simples: usar tu cuenta actual o iniciar sesión con otra cuenta si prefieres.

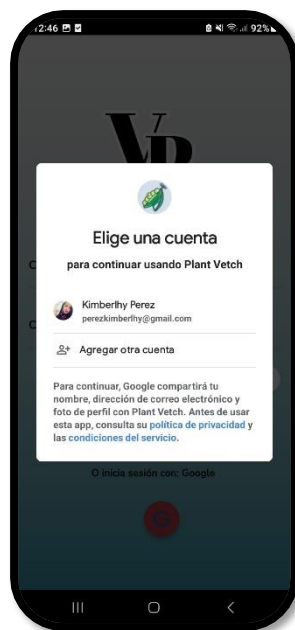


Figura 93. Elegir cuenta de correo electrónico

Una vez que accedes, la pantalla de inicio se mostrará, ofreciendo información introductoria sobre cómo funciona la aplicación.

Menú de inicio



Figura 94. Menú inicio

Manejo Agrónomo



Figura 95. Manejo Agrónomico

En esta sección, encontrarás siete botones, cada uno brindando información crucial que debes tener en cuenta para alcanzar un cultivo exitoso. Las áreas que abordamos son las siguientes:

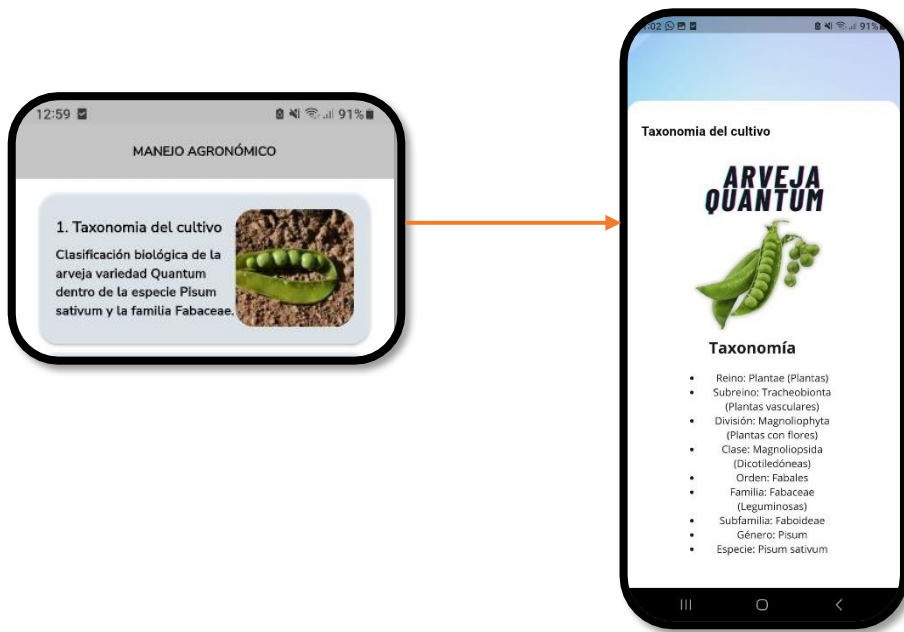


Figura 96. Taxonomía del cultivo

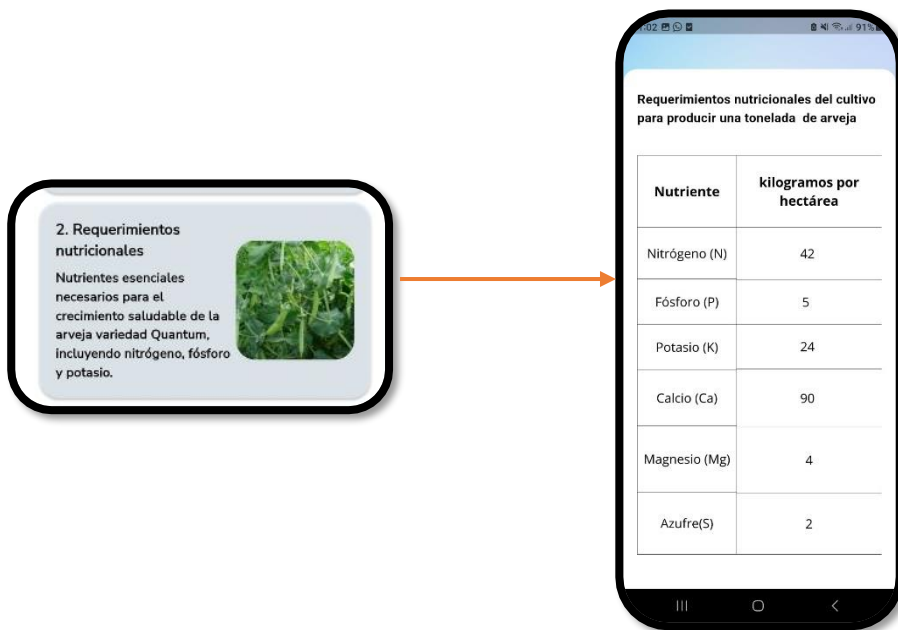


Figura 97. Requerimientos nutricionales

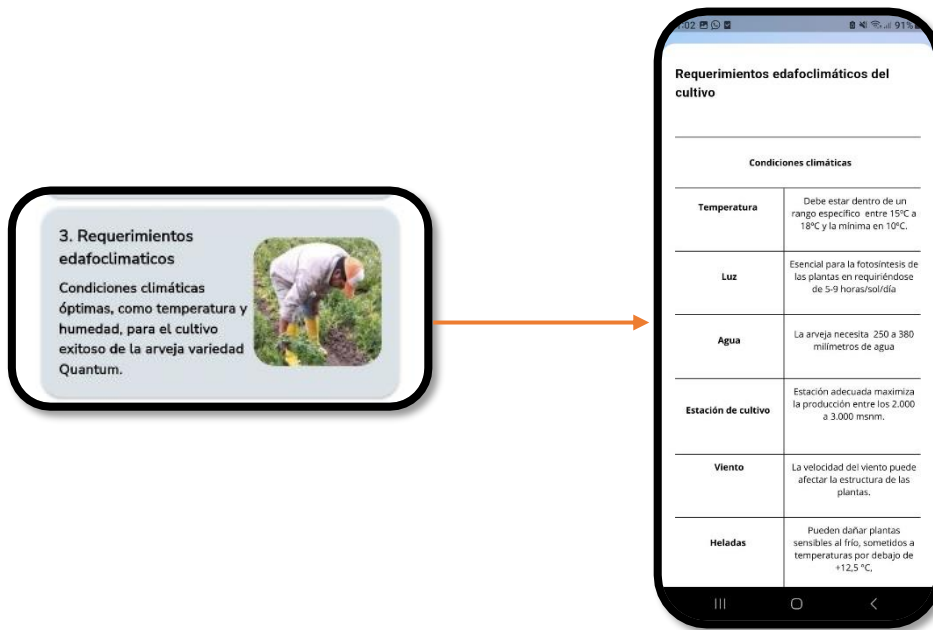


Figura 98. Requerimientos edafoclimáticos

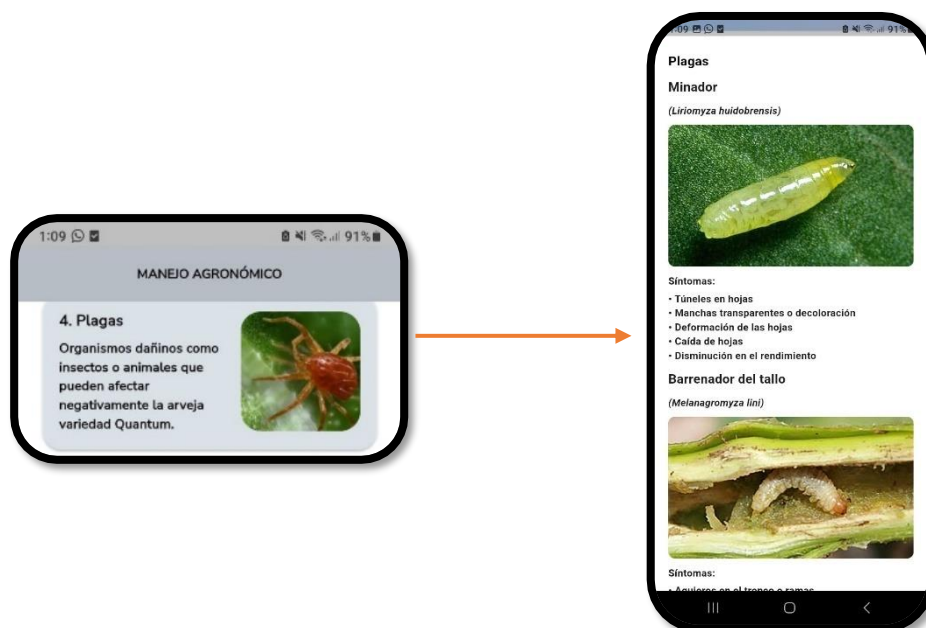



Figura 99. Plagas

5. Enfermedades

Afecciones causadas por patógenos como hongos, bacterias o virus que pueden debilitar o dañar la arveja variedad Quantum.



Amarillamiento o marchitez
(*Xanthomonas spp.*)



Sintomas:

- Color amarillento en las hojas
- Marchitez de las hojas y los tallos
- Retraso en el crecimiento
- Pérdida de rigidez en la estructura de la planta

Signos:

- Presencia de patógenos
- Secreción de sustancias pegajosas en las hojas
- Síntomas en raíces
- Disminución en la producción de flores o frutos

Condiciones Favorables:

- Humedad excesiva del suelo
- Ataque de patógenos del suelo
- Falta de nutrientes esenciales
- Condiciones climáticas adversas

Mancha de Ascochyta
(*Ascochyta spp.*)



Figura 100. Enfermedades

6. Cosecha

Proceso de recolección de guisantes maduros de la arveja variedad Quantum para su consumo o almacenamiento.



Cosecha

Sindamanyo

Cosecha	Producción
Grano verde a los 130 días	Grano verde entre 5.500 y 6.400 kilos por hectárea
Grano seco a los 150 días	Grano seco entre 1.200 y 1.600 kilogramos por hectárea

San Isidro

Figura 101. Cosecha

En este apartado, también encontrarás una serie de botones con consejos prácticos sobre cómo llevar a cabo el proceso de siembra, desde la siembra hasta la cosecha.

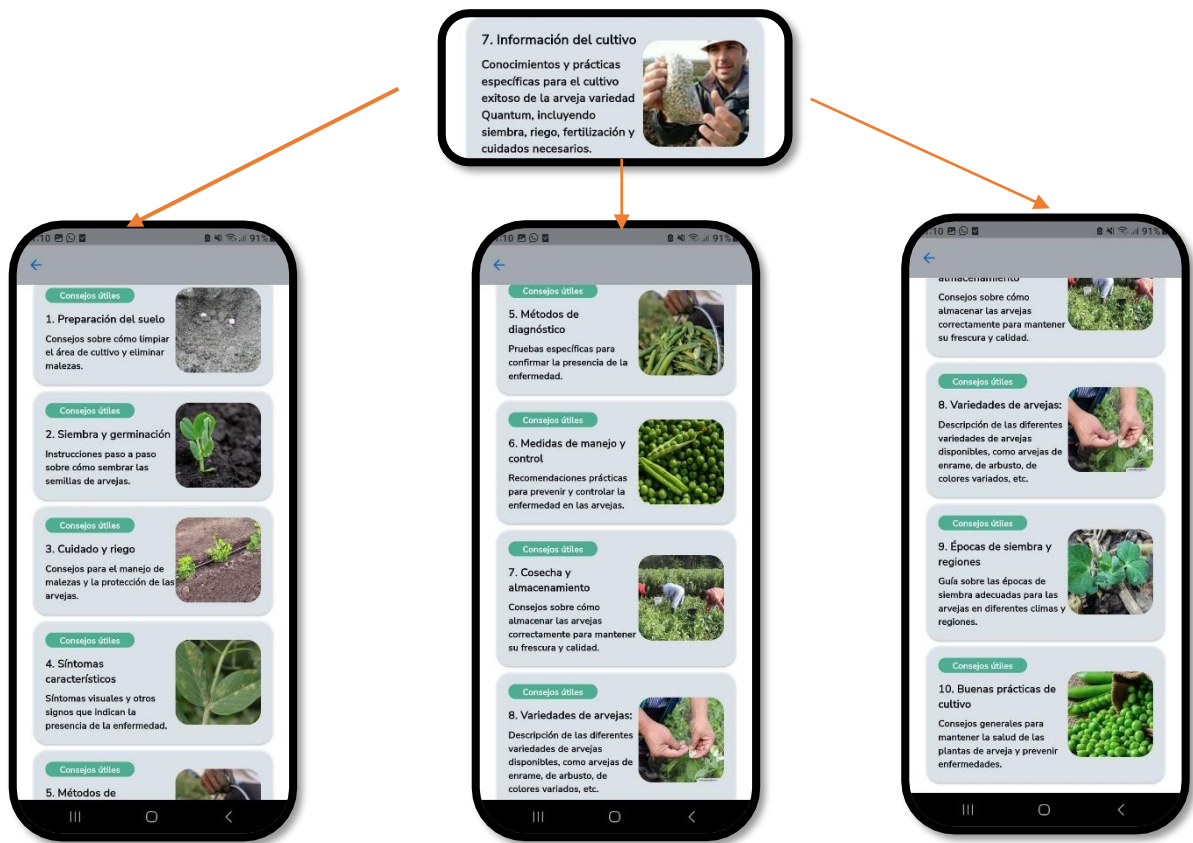


Figura 102. Información del cultivo

Detección de enfermedades

En esta sección, se lleva a cabo el proceso de detección de enfermedades.



Figura 103. Detección enfermedades

El primer botón te dará acceso a la cámara de tu dispositivo para capturar una imagen de la planta enferma. Posteriormente, esta fotografía será procesada con el objetivo de detectar la enfermedad que pueda presentar.

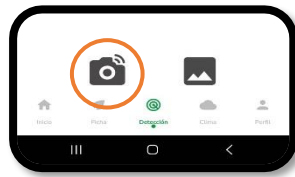


Figura 104. Icono tomar foto desde cámara

A través del segundo botón puedes explorar tu galería de imágenes y seleccionar una foto previamente capturada. De manera similar, la imagen será procesada y mostrará los resultados de la enfermedad detectada.

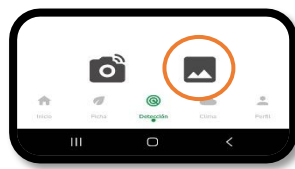


Figura 105. Icono seleccionar desde galería

En esta sección, cada imagen presenta la enfermedad con formas y colores distintivos. Este enfoque busca resaltar de manera efectiva las características únicas de cada afección.

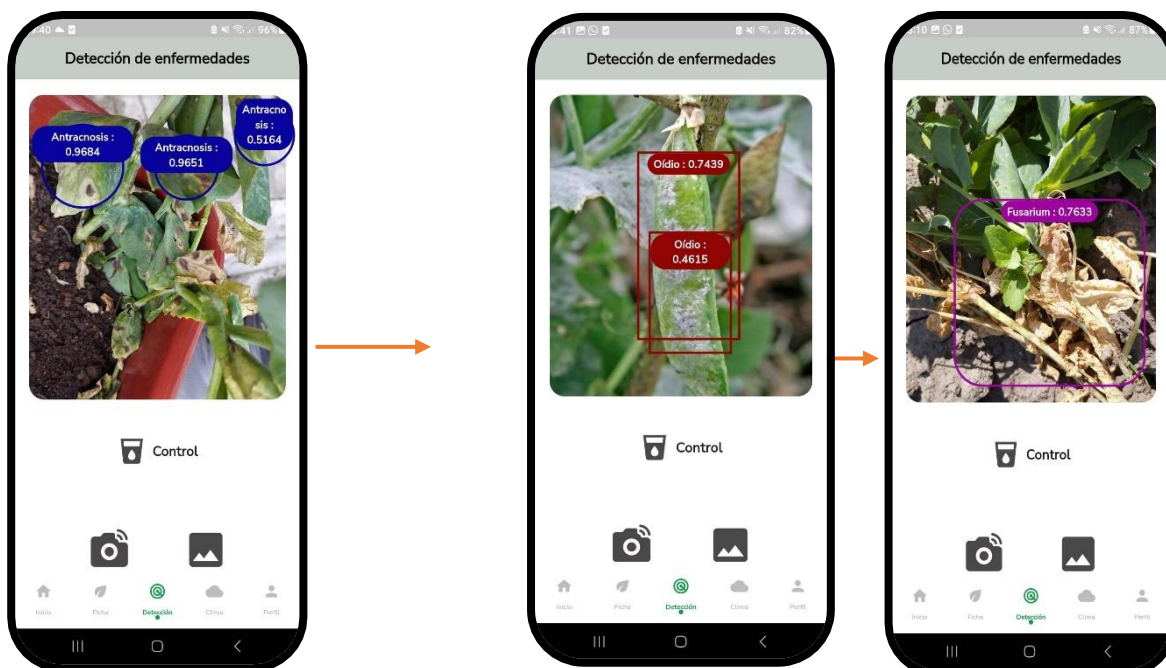


Figura 106. Detección de enfermedades

Una vez identificada la enfermedad, se presenta el control correspondiente para combatirla. En este caso, ofrecemos tanto las dosificaciones preventivas como las curativas específicas para cada una de las enfermedades, permitiendo así abordarlas de manera efectiva.



Figura 107. Icono control

En esta sección, encontraras el control específico para cada una de las enfermedades, junto con sus correspondientes recomendaciones.

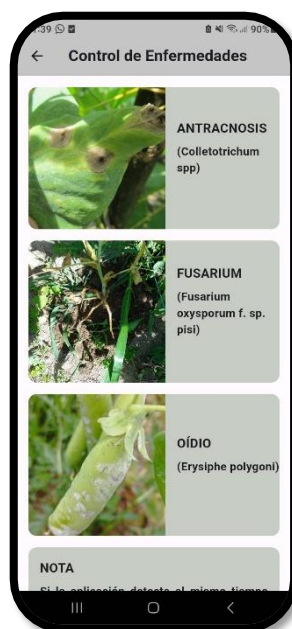


Figura 108. Dosificación preventiva y curativa

Clima

A continuación, se presentan datos informativos fundamentales para el éxito del cultivo. En este contexto, se han incorporado las zonas meteorológicas junto con la altitud, por lo que estos factores desempeñan un papel crucial en el desarrollo óptimo de los cultivos. A continuación, se detalla la información correspondiente a estas zonas.



Figura 109. Zona meteorológica Tulcán



Figura 110. Zona meteorológica Huaca



Figura 111. Zona meteorológica Carmelo



Figura 112. Zona meteorológica Mira

Perfil

En esta sección, podrás visualizar tu perfil personal, incluyendo el correo electrónico con el que te registraste. Esto te brinda la oportunidad de verificar que tu cuenta ha sido guardada correctamente.



Figura 113. Perfil

Además, esta sección incluye el manual de usuario de la aplicación. Al seleccionarlo, los usuarios tendrán acceso a una guía detallada sobre cómo utilizar todas las funcionalidades de la aplicación.

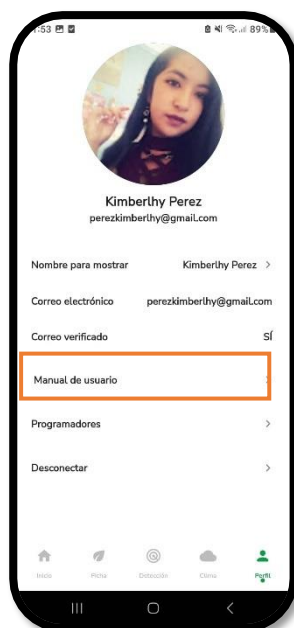


Figura 114. Manual de usuario

Para concluir, esta sección presenta los datos de los programadores de la aplicación

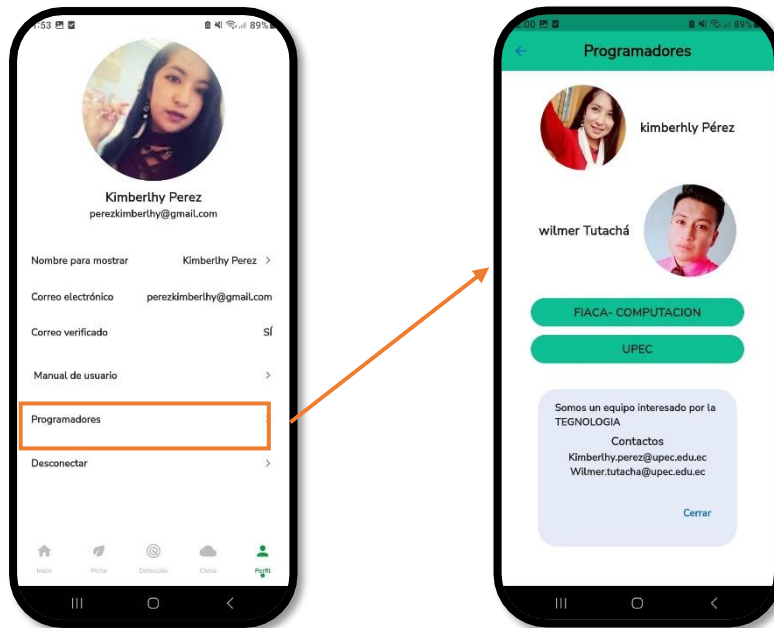


Figura 115. Programadores