

UNIVERSIDAD NACIONAL JOSÉ MARÍA ARGUEDAS

FACULTAD DE INGENIERÍA

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS



Presentado por

JHACKELYN ESTHEFANY LEON HURTADO

**APLICACIÓN MÓVIL (SICKCORN) EN LA
DETECCIÓN DE PUCCINIA SORGHI (ROYA
COMÚN) Y SPIROPLASMA KUNKELI
(PUKA PONCHO) EN LAS PLANTAS DE
MAÍZ, DISTRITO DE TALAVERA**

Asesor

DR. ENRIQUE EDGARDO CONDOR TINOCO

Co - Asesor

MSc. DANIEL MINAYA GUTIERREZ

**TESIS PARA OPTAR EL TÍTULO PROFESIONAL DE
INGENIERO DE SISTEMAS**

ANDAHUAYLAS – APURIMAC – PERÚ

2024



APROBACION DEL ASESOR

Quién suscribe:

Dr. Enrique Edgardo Condor Tinoco por la presente:

CERTIFICA,

Que, la Bachiller en Ingeniería de Sistemas, JHACKELYN ESTHEFANY LEON HURTADO ha iniciado el proceso de asesoramiento y firmamos el compromiso hasta la culminación satisfactoria del Informe Final de Tesis intitulado: “APLICACIÓN MÓVIL (SICKCORN) EN LA DETECCIÓN DE PUCCINIA SORGHI (ROYA COMÚN) Y SPIROPLASMA KUNKELI (PUKA PONCHO) EN LAS PLANTAS DE MAÍZ, DISTRITO DE TALAVERA “, para optar el Título Profesional de Ingeniera de Sistemas.

Andahuaylas, 20 de agosto de 2024.

Dr. Enrique Edgardo Condor Tinoco
Asesor

Jhackelyn Esthefany Leon Hurtado
Tesista



APROBACION DEL CO-ASESOR

Quién suscribe:

M.Sc. Daniel Minaya Gutiérrez por la presente:

CERTIFICA,

Que, la Bachiller en Ingeniería de Sistemas, JHACKELYN ESTHEFANY LEON HURTADO ha iniciado el proceso de asesoramiento y firmamos el compromiso hasta la culminación satisfactoria del Informe Final de Tesis intitulado: "APLICACIÓN MÓVIL (SICKCORN) EN LA DETECCIÓN DE PUCCINIA SORGHI (ROYA COMÚN) Y SPIROPLASMA KUNKELI (PUKA PONCHO) EN LAS PLANTAS DE MAÍZ, DISTRITO DE TALAVERA ", para optar el Título Profesional de Ingeniera de Sistemas.

Andahuaylas, 20 de agosto de 2024.

MSc. Daniel Minaya Gutiérrez
Co-Asesor

Jhackelyn Esthefany Leon Hurtado
Tesista



**UNIVERSIDAD NACIONAL
JOSÉ MARÍA ARGUEDAS**

Licenciada por la Sunedu

ACTA DE SUSTENTACIÓN DE TESIS

En la Av. José María Arguedas del Local Académico SL01 (Ccoyahuacho) en el auditorio de la Escuela Profesional de Ingeniería de Sistemas de la Universidad Nacional José María Arguedas ubicado en el distrito de San Jerónimo de la Provincia de Andahuaylas, siendo las 11:00 horas del día martes 16 de julio de 2024, se reunieron los docentes: MSc. Carlos Yinmel Castro Buleje, MSc. Jesús Farfán Inca Roca, MSc. Rubén Aplaza Apaza, en condición de integrantes del Jurado Evaluador del Informe Final de Tesis intitulado: "APLICACIÓN MÓVIL (SICKCORN) EN LA DETECCIÓN DE PUCCINIA SORGHI (ROYA COMÚN) Y SPIROPLASMA KUNKELI (PUKA PONCHO) EN LAS PLANTAS DE MAÍZ, DISTRITO DE TALAVERA." cuya autora es la Bachiller en Ingeniería de Sistemas **JHACKELYN ESTHEFANY LEON HURTADO**, el asesor Dr. Enrique Edgardo Cóndor Tinoco y el coasesor MSc. Daniel Minaya Gutiérrez, con el propósito de proceder a la sustentación y defensa de dicha tesis.

Luego de la sustentación y defensa de la tesis, el Jurado Evaluador **ACORDÓ: Aprobado** por **Unanimidad** a la Bachiller en Ingeniería de Sistemas **JHACKELYN ESTHEFANY LEON HURTADO**, obteniendo la siguiente calificación y mención:

| Nota escala vigesimal | | Mención |
|-----------------------|---------|---------|
| Números | Letras | |
| 14 | Catorce | Bueno |

En señal de conformidad, se procedió a la firma de la presente acta en 03 ejemplares.

MSc. Carlos Yinmel Castro Buleje
Presidente del Jurado Evaluador

MSc. Jesús Farfán Inca Roca
Primer Miembro del Jurado Evaluador

MSc. Rubén Aplaza Apaza
Segundo Miembro del Jurado Evaluador





ANEXO 31



APROBACIÓN DEL JURADO DICTAMINADOR

LA TESIS: Aplicación Móvil Sickcorn en la Detección de Puccinia Sorghi (Roya Común) y Spiroplasma Kunkeli (Puka Poncho) en las Plantas de Maíz, Distrito de Talavera; para optar el Título Profesional de Ingeniero de Sistemas, ha sido evaluada por el Jurado Dictaminador conformado por:

PRESIDENTE: M.Sc. Carlos Yínmel Castro Buleje.

PRIMER MIEMBRO: M.Sc. Jesús Farfán Inca Roca.

SEGUNDO MIEMBRO: M.Sc. Rubén Apaza Apaza.

Habiendo sido aprobado por UNANIMIDAD, en la ciudad de Andahuaylas el día 16 del mes de julio de 2024.

Andahuaylas, 23 de septiembre de 2024

M.Sc. Carlos Yínmel Castro Buleje.
PRESIDENTE DEL JURADO DICTAMINADOR

M.Sc. Jesús Farfán Inca Roca.
PRIMER MIEMBRO DEL JURADO DICTAMINADOR

M.Sc. Rubén Apaza Apaza.
SEGUNDO MIEMBRO DEL JURADO DICTAMINADOR



Universidad Nacional José María Arguedas

Identidad y Excelencia para el Trabajo Productivo y el Desarrollo

Unidad de Investigación de la Facultad Ingeniería
C019-2024

Andahuaylas, 12 de agosto de 2024

La Unidad de Investigación de la Facultad de Ingeniería, expide la:

Constancia

De porcentaje de similitud (20%) según el software Turnitin, al informe final de investigación: Aplicación móvil (Sickcorn) en la detección de *Puccinia sorghi* (roya común) y *Spiroplasma kunkeli* (pukaponcho) en las plantas de maíz, distrito de Talavera. Presentado por la Bach. **Jhackelyn Esthefany León Hurtado** cuyo Asesor es el Dr. Enrique Edgardo Córdor Tinoco y coasesor el MSc. Daniel Minaya Gutiérrez.

.....
Dra. María del Carmen Delgado Laime
Presidente de la Unidad de Investigación de la
Facultad de Ingeniería

.....
MSc. Fidelia Tapia Tadeo
Miembro de la
Unidad de Investigación de la Facultad de Ingeniería

.....
Mro. Richard Carrión Abollaneda
Miembro de la
Unidad de Investigación de la Facultad de Ingeniería

NOMBRE DEL TRABAJO

APLICACIÓN MÓVIL (SICKCORN) EN LA DETECCIÓN DE PUCCINIA SORGHI (ROYA COMÚN) Y SPIROPLASMA KUNKELI (PUKA PONCHO), DISTRITO DE TALAVERA.

AUTOR

JHACKELYN ESTHEFANY LEON HURTADO

RECUENTO DE PALABRAS

27068 Words

RECUENTO DE CARACTERES

157957 Characters

RECUENTO DE PÁGINAS

121 Pages

TAMAÑO DEL ARCHIVO

3.0MB

FECHA DE ENTREGA

Aug 5, 2024 10:12 PM GMT-5

FECHA DEL INFORME

Aug 5, 2024 10:14 PM GMT-5

● **20% de similitud general**

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para cada base de datos.

- 19% Base de datos de Internet
- Base de datos de Crossref
- 4% Base de datos de publicaciones
- Base de datos de contenido publicado de Crossref

● **Excluir del Reporte de Similitud**

- Material bibliográfico
- Coincidencia baja (menos de 10 palabras)



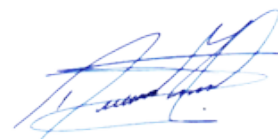
Tesista

Jhackelyn Esthefany Leon Hurtado



Asesor

Dr. Enrique Edgardo Condor Tinoco



Co-Asesor

MSc. Daniel Minaya Gutiérrez

● 20% de similitud general

Principales fuentes encontradas en las siguientes bases de datos:

- 19% Base de datos de Internet
- Base de datos de Crossref
- 4% Base de datos de publicaciones
- Base de datos de contenido publicado de Crossref

FUENTES PRINCIPALES

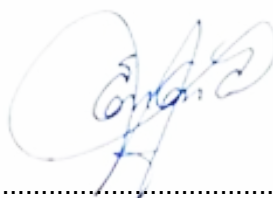
Las fuentes con el mayor número de coincidencias dentro de la entrega. Las fuentes superpuestas no se mostrarán.

| | | |
|---|--|-----|
| 1 | repositorio.unajma.edu.pe Internet | 3% |
| 2 | sedici.unlp.edu.ar Internet | 2% |
| 3 | ars.usda.gov Internet | 1% |
| 4 | hdl.handle.net Internet | 1% |
| 5 | repositorio.upn.edu.pe Internet | <1% |
| 6 | idus.us.es Internet | <1% |
| 7 | repositorio.ucv.edu.pe Internet | <1% |
| 8 | es.slideshare.net Internet | <1% |



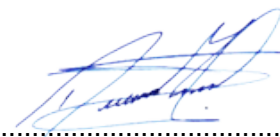
Tesista

Jhackelyn Esthefany Leon Hurtado



Asesor

Dr. Enrique Edgardo Condor Tinoco



Co-Asesor

MSc. Daniel Minaya Gutiérrez
Descripción general de fuentes

- 9

aws.amazon.com

Internet

<1%
- 10

slideshare.net

Internet

<1%
- 11

coursehero.com

Internet

<1%
- 12

repositorio.utn.edu.ec

Internet

<1%
- 13

Bernedo, Juan Francisco Mendoza | Bustamante, Fernando Jesus Sald...

Publication

<1%
- 14

repositorio.lamolina.edu.pe

Internet

<1%
- 15

slideplayer.es

Internet

<1%
- 16

upc.aws.openrepository.com

Internet

<1%
- 17

repositorio.upao.edu.pe

Internet

<1%
- 18

inf.u-szeged.hu


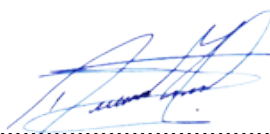
Internet

<1%
- 19

secgen.unajma.edu.pe

Internet

<1%

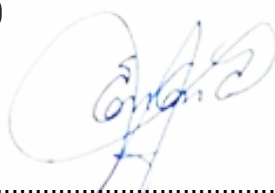
| | | |
|--|---|--|
| 20 <p>gredos.usal.es</p> <p>Internet</p> <1% |  |  |
| <p>Tesista</p> <p>Jhackelyn Esthefany Leon Hurtado</p> | <p>Asesor</p> <p>Dr. Enrique Edgardo Condor Tinoco</p> | <p>Co-Asesor</p> <p>MSc. Daniel Minaya Gutiérrez</p> <p>Descripción general de fuentes</p> |

| | | |
|----|--|-----|
| 21 | 1library.co Internet | <1% |
| 22 | revistas.unamba.edu.pe Internet | <1% |
| 23 | fdocuments.mx Internet | <1% |
| 24 | purl.org Internet | <1% |
| 25 | repositorio.upagu.edu.pe Internet | <1% |
| 26 | foroiso25000.blogspot.com Internet | <1% |
| 27 | researchgate.net Internet | <1% |
| 28 | 123ered.blogspot.com Internet | <1% |
| 29 | libros.catedu.es Internet | <1% |
| 30 | powtoon.com Internet | <1% |
| 31 | repositorio.une.edu.pe Internet | <1% |
| 32 | dspace.uce.edu.ec:8080 Internet | <1% |



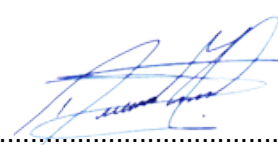
Tesista

Jhackelyn Esthefany Leon Hurtado



Asesor

Dr. Enrique Edgardo Condor Tinoco



Co-Asesor

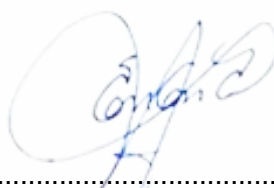
MSc. Daniel Minaya Gutiérrez
Descripción general de fuentes

| | | |
|----|---|-----|
| 33 | repositorio.utc.edu.ec Internet | <1% |
| 34 | agroperu.pe Internet | <1% |
| 35 | dspace.espol.edu.ec Internet | <1% |
| 36 | gob.pe Internet | <1% |
| 37 | scribd.com Internet | <1% |
| 38 | repositorio.uncp.edu.pe Internet | <1% |
| 39 | eprints.uanl.mx Internet | <1% |
| 40 | prezi.com Internet | <1% |
| 41 | repositorio.unheval.edu.pe Internet | <1% |
| 42 | costoya.com Internet | <1% |
| 43 | doczz.net Internet | <1% |
| 44 | dropbox.com Internet | <1% |



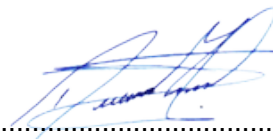
Tesista

Jhackelyn Esthefany Leon Hurtado



Asesor

Dr. Enrique Edgardo Condor Tinoco



Co-Asesor

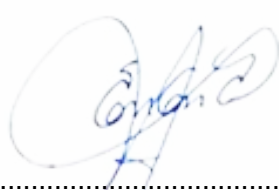
MSc. Daniel Minaya Gutiérrez
Descripción general de fuentes

| | | |
|----|--|-----|
| 45 | repositorio.ug.edu.ec Internet | <1% |
| 46 | lensculture.com Internet | <1% |
| 47 | docplayer.es Internet | <1% |
| 48 | huggingface.co Internet | <1% |
| 49 | repositorio.espe.edu.ec Internet | <1% |
| 50 | lanner-america.com Internet | <1% |
| 51 | addi.ehu.es Internet | <1% |
| 52 | dspace.ucuenca.edu.ec Internet | <1% |
| 53 | e-spacio.uned.es Internet | <1% |
| 54 | rdu.unc.edu.ar Internet | <1% |
| 55 | repositorio.unap.edu.pe Internet | <1% |
| 56 | repositorio.unsaac.edu.pe Internet | <1% |



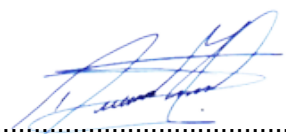
Tesista

Jhackelyn Esthefany Leon Hurtado



Asesor

Dr. Enrique Edgardo Condor Tinoco



Co-Asesor

MSc. Daniel Minaya Gutiérrez
Descripción general de fuentes

- 57
rinacional.tecnm.mx
<1%

Internet
- 58
Astrid Vanessa Padilla Jimenez, Camilo Andres Pulido Rojas, Leonard...
<1%

Crossref
- 59
dspace.ups.edu.ec
<1%

Internet
- 60
malaysia.shafaqna.com
<1%

Internet
- 61
repositorio.unamba.edu.pe
<1%

Internet
- 62
repositorio.uteq.edu.ec
<1%

Internet
- 63
risti.xyz
<1%

Internet
- 64
anyflip.com
<1%


Internet
- 65
es.unionpedia.org
<1%

Internet
- 66
par.nsf.gov
<1%

Internet
- 67
repositorio.usmp.edu.pe
<1%

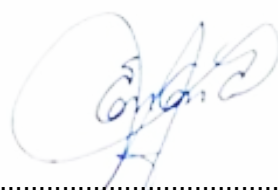
Internet
- 68
repositorio.uss.edu.pe
<1%

Internet




Tesista

Jhackelyn Esthefany Leon Hurtado



Asesor

Dr. Enrique Edgardo Condor Tinoco

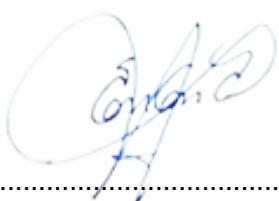


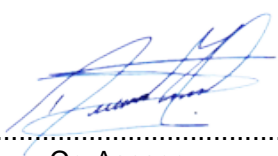
Co-Asesor

MSc. Daniel Minaya Gutiérrez
Descripción general de fuentes

- 69
revistas.unla.edu.ar
<1%
Internet
- 70
adama.com
<1%
Internet
- 71
catamarcaguia.com.ar
<1%
Internet
- 72
xn--baterasadaptador-dsb.es
<1%
Internet
- 73
Darko, Michael A.. "Multi-Class Anomaly-Based Intrusion Detection Sy...
<1%
Publication
- 74
Galván, José Guillermo Balbuena. "Modelos de Detección de Emocione...
<1%
Publication
- 75
get-winearticles.blogspot.com
<1%
Internet
- 76
idm.fing.edu.uy
<1%
Internet
- 77
palermo.edu.ar
<1%
Internet
- 78
repositorio.unprg.edu.pe
<1%
Internet
- 79
sintesis.ugto.mx
<1%
Internet

80
accu-chek.es
<1%
Internet





.....
 Tesista
 Jhackelyn Esthefany Leon Hurtado

.....
 Asesor
 Dr. Enrique Edgardo Condor Tinoco

.....
 Co-Asesor
 MSc. Daniel Minaya Gutiérrez
 Descripción general de fuentes

- 81
eumed.net
<1%

Internet
- 82
redalyc.org
<1%

Internet
- 83
Cynthia Sophia Arellano Benitez, Irving D. Hernandez. "DEVELOPMENT...
<1%

Publication
- 84
D. Kacalek, D. Dusek, J. Novak, M. Slodicak, J. Bartos, V. Cernohous, V....
<1%

Crossref
- 85
Martinez Pallares, Fabian M.. "Luminescent Hydroxyapatite: Degradati...
<1%

Publication
- 86
Mohammad Agung Wibowo, Muhamad Soleh, Winangsari Pradani, Ach...
<1%

Crossref
- 87
Rodrigues, Tiago. "RGBD-Based Automatic Stem Selection for Selectiv...
<1%

Publication
- 88
docs.google.com
<1%

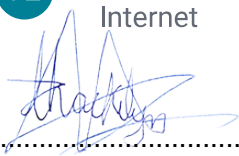
Internet
- 89
dspace.espoch.edu.ec
<1%

Internet
- 90
dspace.ueb.edu.ec
<1%

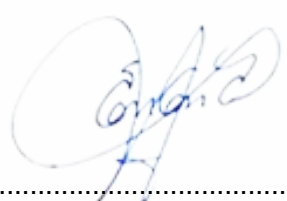
Internet
- 91
dspace.unitru.edu.pe
<1%

Internet
- 92
idoc.pub
<1%


Internet



.....
 Tesista
 Jhackelyn Esthefany Leon Hurtado

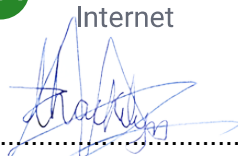


.....
 Asesor
 Dr. Enrique Edgardo Condor Tinoco



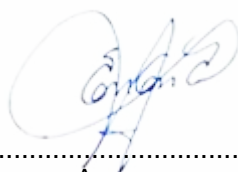
.....
 Co-Asesor
 MSc. Daniel Minaya Gutiérrez
 Descripción general de fuentes

| | | |
|-----|--|-----|
| 93 | poznan.ksiegarnienaukowe.pl Internet | <1% |
| 94 | pt.scribd.com Internet | <1% |
| 95 | red.uao.edu.co Internet | <1% |
| 96 | repositorio.enap.gov.br Internet | <1% |
| 97 | repositorio.unjfsc.edu.pe Internet | <1% |
| 98 | repositorio.utea.edu.pe Internet | <1% |
| 99 | ri.uaemex.mx Internet | <1% |
| 100 | rua.ua.es Internet | <1% |
| 101 | techlib.net Internet | <1% |
| 102 | tesis.ucsm.edu.pe Internet | <1% |
| 103 | wicc.it.itba.edu.ar Internet | <1% |
| 104 | academia.edu Internet | <1% |



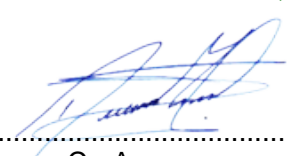
Tesis

Jhackelyn Esthefany Leon Hurtado



Asesor

Dr. Enrique Edgardo Condor Tinoco



Co-Asesor

MSc. Daniel Minaya Gutiérrez
Descripción general de fuentes

105

greenpeace.org

Internet

<1%

106

nobleprog.co

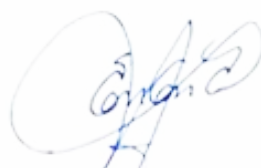
Internet

<1%



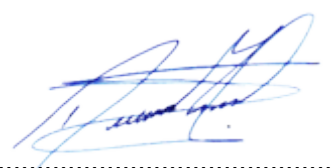
Tesista

Jhackelyn Esthefany Leon Hurtado



Asesor

Dr. Enrique Edgardo Condor Tinoco



Co-Asesor

MSc. Daniel Minaya Gutiérrez

DEDICATORIA

Dedico el presente trabajo de investigación en memoria de Pelusa (Bombón) quien fue mi compañera y protectora durante todo esta etapa, además de todas las personas que contribuyeron en el desarrollo, tanto de esta investigación como en mi desarrollo profesional.

JELH

AGRADECIMIENTOS

Quiero expresar mi más profundo agradecimiento a mis padres por haber sido mi principal sostén durante todo el proceso de investigación. Su apoyo fue esencial y fundamental en cada etapa de mi vida.

A la Universidad Nacional José María Arguedas, le estoy agradecida por haberme otorgado la oportunidad de formar parte de su comunidad estudiantil. Reconozco y valoro la formación profesional que recibí en esta institución, la cual ha sido crítica para alcanzar mis metas y superar los desafíos que se me presentan en el camino.

Asimismo, deseo expresar mi más sincero agradecimiento al ingeniero Iván Soria, Daniel Minaya, Enrique Tinoco, a mis estimados amigos, docentes, jurados y asesores, cuya dedicación y sabiduría fueron pilares fundamentales en mi desarrollo académico.

A todos aquellos que han contribuido de alguna manera a mi formación y desarrollo, les expreso mi más sincero agradecimiento.

Índice

| | |
|--|----|
| Capítulo 1 Problema de Investigación | 7 |
| 1.1. Realidad Problemática..... | 7 |
| 1.2. Formulación del Problema..... | 8 |
| 1.2.1. Problema General..... | 8 |
| 1.2.2. Problemas Específicos..... | 9 |
| 1.3. Objetivos..... | 9 |
| 1.3.1. Objetivo General | 9 |
| 1.3.2. Objetivos Específicos..... | 9 |
| 1.4. Justificación | 10 |
| 1.5. Limitaciones..... | 11 |
| Capítulo 2 Antecedentes..... | 12 |
| 2.1. Antecedentes Internacionales..... | 12 |
| 2.2. Antecedentes Nacionales | 13 |
| Capítulo 3 Marco Teórico | 15 |
| 3.1. El Maíz..... | 15 |
| 3.1.1. Origen del Maíz..... | 15 |
| 3.1.2. Ciclo de Vida del Maíz..... | 16 |
| 3.1.3. Desarrollo de la Planta..... | 19 |
| 3.1.4. Variedades del Maíz..... | 21 |
| 3.2. Enfermedades en Plantas de Maíz..... | 30 |
| 3.2.1. Enfermedades Foliarias. | 30 |
| 3.2.2. Enfermedades Causadas por Cronistas. | 30 |
| 3.2.3. Enfermedades Causadas por Bacterias:..... | 30 |
| 3.2.4. Plagas | 31 |
| 3.2.5. Puccinia Sorghi (Roya común)..... | 31 |
| 3.2.6. Spiroplasma Kunkeli (puka poncho). | 32 |
| 3.3. Sistemas de Información..... | 33 |
| 3.4. Tecnologías Móviles..... | 34 |
| 3.5. Dispositivo Móvil..... | 34 |
| 3.5.1. Tipos de Dispositivos Móviles | 35 |
| 3.5.2. Sistema Operativo para Móviles..... | 35 |
| 3.5.3. Android..... | 36 |

| | |
|--|----|
| 3.5.4. Arquitectura Android..... | 36 |
| 3.6. Aplicación Móvil..... | 37 |
| 3.6.1. Tipos de Aplicaciones para Dispositivos Móviles | 38 |
| 3.7. Inteligencia Artificial..... | 38 |
| 3.7.1. Componentes de la Inteligencia Artificial..... | 39 |
| 3.7.2. Tipos de Aprendizaje..... | 40 |
| 3.7.3. Transfer Learning | 40 |
| 3.7.4. Visión Artificial | 41 |
| 3.8. Preprocesamiento de Imágenes..... | 43 |
| 3.8.1. Etapas del Procesamiento de Imágenes:..... | 43 |
| 3.9. Redes Neuronales Artificiales | 45 |
| 3.9.1. Arquitectura de la Red Neuronal | 46 |
| 3.9.2. Tipos de Redes Neuronales Artificiales | 46 |
| 3.9.3. Redes Neuronales Convolucionales (CNN)..... | 48 |
| 3.9.4. Componentes de la Arquitectura CNN..... | 49 |
| 3.9.5. Arquitecturas de Redes Neuronales Convolucionales..... | 50 |
| 3.10. Metodología Fundamental para Ciencia de Datos de IBM | 51 |
| 3.10.1. Etapa 1: Comprensión del Negocio | 52 |
| 3.10.2. Etapa 2: Enfoque Analítico..... | 52 |
| 3.10.3. Etapa 3: Requisitos de Datos..... | 52 |
| 3.10.4. Etapa 4: Recopilación de Datos..... | 53 |
| 3.10.5. Etapa 5: Comprensión de Datos..... | 53 |
| 3.10.6. Etapa 6: Preparación de Datos..... | 53 |
| 3.10.7. Etapa 7: Modelado | 53 |
| 3.10.8. Etapa 8: Evaluación | 53 |
| 3.10.9. Etapa 9: Implementación..... | 54 |
| 3.10.10. Etapa 10: Retroalimentación..... | 54 |
| 3.11. Metodología de Desarrollo Mobile-D..... | 54 |
| 3.11.1. Fase 1: Exploración | 55 |
| 3.11.2. Fase 2: Iniciación..... | 55 |
| 3.11.3. Fase 3: Producción..... | 55 |
| 3.11.4. Fase 4: Estabilización | 55 |
| 3.11.5. Fase 5: Pruebas..... | 56 |

| | |
|---|----|
| 3.12. ISO/IEC 25000 | 56 |
| 3.12.1. ISO/IEC 25010:2011 | 56 |
| 3.12.2. ISO/IEC 25022:2016 | 57 |
| 3.12.3. ISO/IEC 25023:2016 | 58 |
| 3.12.4. ISO/IEC 25040:2011 | 59 |
| 3.13. Herramientas..... | 62 |
| 3.13.1. Google Colab | 62 |
| 3.13.2. Kaggle..... | 62 |
| 3.13.3. Python..... | 62 |
| 3.13.4. Librerías | 63 |
| 3.13.5. Android Studio..... | 64 |
| 3.13.6. Java..... | 64 |
| 3.13.7. Firebase..... | 64 |
| 3.13.8. Dataset..... | 64 |
| 3.14. Métricas de Evaluación..... | 65 |
| 3.14.1. Accuracy..... | 65 |
| 3.14.2. Precisión..... | 65 |
| 3.14.3. Recall..... | 66 |
| 3.14.4. F1 score..... | 66 |
| 3.14.5. Matriz de Confusión..... | 67 |
| Capítulo 4 Metodología de la Investigación | 68 |
| 4.1. Tipo de Investigación..... | 68 |
| 4.2. Enfoque de Investigación..... | 68 |
| 4.3. Diseño de Investigación..... | 68 |
| 4.4. Nivel de Investigación | 69 |
| 4.5. Población..... | 69 |
| 4.6. Muestra..... | 69 |
| 4.7. Técnicas e Instrumentos de Recopilación de Datos | 71 |
| 4.8. Confiabilidad de los Datos..... | 71 |
| 4.9. Técnicas y Análisis de Datos | 71 |
| 4.10. Operacionalización de Variables..... | 72 |
| Capítulo 5 Resultados..... | 74 |
| 5.1. Evaluación de los Modelos..... | 74 |

| | |
|--|-----|
| 5.1.1. Evaluación del Modelo Implementado | 74 |
| 5.1.2. Evaluación del Modelo VGG16..... | 77 |
| 5.1.3. Evaluación del Modelo MobileNetV2 | 80 |
| 5.1.4. Evaluación de la Taza de Error de los Modelos..... | 83 |
| 5.2. Evaluación de la Aplicación Móvil | 84 |
| 5.2.1. Precisión de la Aplicación Móvil | 85 |
| 5.2.2. Exactitud (Accuracy) de la Aplicación Móvil | 85 |
| 5.2.3. Sensibilidad (Recall) de la Aplicación Móvil..... | 85 |
| 5.2.4. FI-score de la Aplicación Móvil | 86 |
| 5.3. Resultados de la ISO/IEC 25000..... | 86 |
| 5.3.1. Resultados de la Evaluación de la Subcaracterísticas de Calidad | 86 |
| 5.3.2. Resultados de la Evaluación de la Característica de Calidad..... | 88 |
| Capítulo 6 Discusión | 89 |
| Conclusiones..... | 91 |
| Recomendaciones..... | 92 |
| Referencias Bibliográficas..... | 93 |
| Anexo | 100 |

Lista de Tablas

| | |
|---|----|
| Tabla 1 Matriz de Confusion..... | 67 |
| Tabla 2 Operacionalización de Variables..... | 72 |
| Tabla 38 Resultado del Modelo Implementado | 74 |
| Tabla 39 Resultado del Modelo VGG16..... | 78 |
| Tabla 40 Resultados del Modelo MobilenetV2..... | 81 |
| Tabla 41 Taza de Error..... | 84 |
| Tabla 42 Resultados de la Aplicación Móvil..... | 84 |
| Tabla 44 Resultados de la Característica de Calidad | 88 |

JELH

Lista de Figuras

| | |
|---|----|
| Figura 1 Puccinia Sorghi..... | 32 |
| Figura 2 Spiroplasma Kunkeli..... | 33 |
| Figura 3 Estructura Red Neuronal..... | 46 |
| Figura 4 Estructura Básica de la Red Neuronal Convolutcional..... | 49 |
| Figura 5 Metodología Fundamental para la Ciencia de Datos..... | 52 |
| Figura 6 Ciclo de Vida de la Metodología Mobile-D..... | 55 |
| Figura 7 Divisiones de Calidad..... | 56 |
| Figura 8 Características y Subcaracterísticas de Calidad Según SQuaRE..... | 57 |
| Figura 9 Matriz de Confusión del Modelo Implementado | 75 |
| Figura 10 Matriz de Confusión del Modelo VGG16..... | 78 |
| Figura 11 Matriz de Confusión del Modelo MobileNetV2 | 81 |

JELH

RESUMEN

La identificación de enfermedades en las plantas de maíz presenta un desafío para los productores, debido a la vasta cantidad de enfermedades existentes en el mundo agrícola. Sin embargo, en la actualidad es invaluable realizar un monitoreo constante a la producción para disminuir la afección de las enfermedades, por la carencia de herramientas tecnológicas que ayuden a realizar un monitoreo para una detección precisa a tiempo.

Es ese sentido, el presente estudio propuso determinar la precisión de la aplicación móvil desarrollada para la identificación de *Puccinia sorghi* (roya común), *Spiroplasma kunkeli* (puka poncho), capturando las imágenes de las plantas de maíz afectadas utilizando la técnica de observación directa empleando una cámara de celular del modelo Samsung Galaxy M32 como instrumento para luego realizar la validación de los datos y el preprocesamiento en el cual se utilizó la técnica del histograma, el filtro gaussiano, el filtro de nitidez, segmentación con el algoritmo Canny y el resaltado HSV. La población de estudio estuvo compuesta de dos clases de enfermedades: *Puccinia sorghi* (roya común), *Spiroplasma kunkeli* (puka poncho) y planta sana. La cantidad de datos de la muestra fue 346 imágenes que tengan algunas de las afecciones en el distrito de Talavera. Con la finalidad de alcanzar este objetivo, se implementó un modelo de clasificación basado en redes neuronales convolucionales. El proceso incluyó varias etapas esenciales realizadas por la metodología fundamental para ciencia de datos de IMB, en la que se evaluó la arquitectura VGG16, MobilenetV2 y la implementada en CNN por el investigador con la finalidad de obtener el modelo que tenga mejores resultados para que se integren en la aplicación móvil desarrollada en Android Studio, demostrando la precisión de que está logrando un valor del 92.6% de precisión para la clase puccinia sorgui y 94.4% para la clase spiroplasma kunkeli; también se realizaron pruebas de calidad de la aplicación móvil por medio de la familia de la ISO/IEC 25000.

Palabras claves: aplicación móvil, red neuronal convolucional, puccinia sorghi, spiroplasma kunkeli.

ABSTRACT

The identification of diseases in corn plants presents a challenge for producers, due to the vast number of diseases existing in the agricultural world. However, currently it is invaluable to carry out constant monitoring of production, to reduce the impact of diseases, due to the lack of technological tools that help carry out monitoring for accurate detection in time.

In this sense, the present study proposed to determine the accuracy of the mobile application developed for the identification of *Puccinia sorghi* (common rust), *Spiroplasma kunkeli* (puka poncho), capturing images of the affected corn plants using the direct observation technique using a cell phone camera of the Samsung Galaxy M32 model as an instrument to then carry out data validation and preprocessing in which the histogram technique, the Gaussian filter, the sharpness filter, segmentation with the Canny algorithm and HSV highlighting were used. . The study population was composed of two classes of diseases: *Puccinia sorghi* (common rust), *Spiroplasma kunkeli* (puka poncho) and healthy plant. The amount of data in the sample was 346 images that have some of the conditions in the Talavera district.

In order to achieve this objective, a classification model based on convolutional neural networks was implemented. The process included several essential stages carried out by the IMB fundamental methodology for data science, in which the VGG16, MobilenetV2 architecture and the one implemented in CNN by the researcher were evaluated in order to obtain the model that has the best results so that the integrate into the mobile application developed in Android Studio, demonstrating the precision that is achieving a value of 92.6% precision for the puccinia sorgui class and 94.4% for the spiroplasma kunkeli class; Quality tests of the mobile application were also carried out through the ISO/IEC 25000 family.

Keywords mobile application, convolutional neural network, puccinia sorghi, spiroplasma kunkeli.

CHUMASQA

Sara yurakunapi unquykuna riqsichiyqa ruruchi kunapaq sasachakuymi, chakra llamkaypi achka unquykuna kasqanrayku. Ichaqa kunan pachaqa mana chaniyuqmi sapa kutilla ruruchiy qhawayta ruway, unquykuna pisiyachinapaq, mana herramientas tecnológicas nisqakuna kasqanrayku, chaykunan yanapan qhawayta ruwayta, tiempollanpi allinta tarinapaq.

Kay yuyaypiqa, kay kunan yachayqa, Puccinia sorghi (herrumbre común), Spiroplasma kunkeli (puka poncho) riqsichinapaq ruwasqa aplicacin móvil nisqapa chiqap kayninta yachanapaqmi yuyaychakurqa, chaypim hapirqa imahinakuna sara yurakuna afectasqa, chay técnica de observación directa nisqawan, célula nisqawan telefono cámara kaqmanta Samsung Galaxy M32 modelo kaqmanta huk instrumento hina chaymanta ruwanapaq datos validación chaymanta preprocesamiento maypichus técnica histograma kaqmanta, filtro Gaussiano kaqmanta, filtro de nitidez kaqmanta chaymanta HSV resaltay kaqmanta llamk'achirqanku. Kay estudio población iskay clase unquykunamanta ruwasqa karqa: Puccinia sorghi (herrumbre común), Spiroplasma kunkeli (puka poncho) chanta qhali planta, kay cantidad de datos kay muestra kaqpi karqa 346 imaymanakuna mayqinkunachus wakin condiciones kaqwan kanku kay distrito Talavera kaqpi . Kay ruwayta hunt'anapaqmi, redes neuronales convolucionales nisqapi ruwasqa modelo de clasificación nisqa ruwakurqa. Kay ruwayqa achka etapas esenciales ruwasqa IMB fundamental metodología ciencia de datos kaqpaq, maypichus VGG16, MobilenetV2 arquitectura chaymanta chay CNN kaqpi investigadorpa implementasqa kaqnin chaninchasqa karqa chaymanta chay modelota tarinapaq mayqinchus aswan allin ruwaykunayuq chaymanta chay integrado Android Studio kaqpi ruwasqa kuyuchina ruwanaman, 94.7% chiqan kay chanin aypaypaq chiqan kayninta rikuchispa, chaymanta prueba calidad kaqmanta kay móvil aplicación kaqmanta ISO/IEC 25000 ayllu kaqnintakama.

Sapaq simikuna: apaykachanapaq apaykachana, Ilika neuronal convolucional, puccinia sorghi, spiroplasma kunkeli.

INTRODUCCIÓN

El maíz es uno de los cereales más importantes del mundo porque es un alimento fundamental tanto para humanos como para animales. El maíz es uno de los productos más significativos del Perú; se cultivan en promedio 520 mil hectáreas al año y aproximadamente 82 mil familias dependen directamente de este cultivo, lo cual demuestra su relevancia socioeconómica. (García, 2020)

En la actualidad se vio afectada la producción y la calidad del grano, por las diversas plagas y enfermedades que atacan este producto. Según (Agudelo & Sarria, 2006-2007, pág. 7) Las condiciones ambientales, el tipo de suelo, la susceptibilidad de los materiales y las enfermedades virales, que fomenten la migración, establecimiento y supervivencia de insectos vectores, son factores que fomenten la enfermedad.

Los dispositivos móviles se definen como dispositivos electrónicos compactos y portátiles. Concebidos para facilitar su transporte y uso mientras una persona se encuentra en movimiento, su diseño está orientado a ofrecer una amplia gama de funciones que abarcan desde la comunicación hasta el acceso a internet, pasando por el entretenimiento y la productividad.

La detección rápida y precisa de estas enfermedades se ha convertido en un desafío para los productores ya que se requiere experiencia para detectarlas. Para abordar este problema, este artículo se centra en evaluar la precisión de las aplicaciones móviles que utilizan redes neuronales convolucionales (CNN) para identificar enfermedades de las plantas de maíz. El objetivo principal de este estudio fue determinar la precisión de una aplicación móvil para identificar *Puccinia sorghi* (roya común) y *Spiroplasma kunkii* (puka poncho) en plantas de maíz del distrito de Talavera.

En general, este estudio tiene como objetivo enriquecer la detección de enfermedades de las plantas de maíz y al mismo tiempo proporcionar una herramienta útil para la identificación de enfermedades.

La estructura de la tesis está representada de la siguiente manera: En el primer capítulo, se aborda la problemática, la formulación del problema, los objetivos y la justificación del estudio. En el segundo capítulo abordamos los antecedentes de investigación y las bases teóricas que fundamentan esta investigación, mientras que en el tercer capítulo se detalla la metodología empleada en el estudio. En el cuarto capítulo, se aborda el uso de la metodología fundamental para ciencia de datos de IBM y la metodología de desarrollo Mobile-D para el desarrollo de la aplicación móvil, además de la implementación de la ISO/IEC 25000. En el quinto capítulo, se presentan los resultados obtenidos, por último, se proporciona una discusión de los resultados junto a los anexos.

JELH

Capítulo 1

Problema de Investigación

1.1. Realidad Problemática

Los problemas de enfermedades en las plantas de maíz son un problema importante para la agricultura y la seguridad alimentaria, ya que el maíz es uno de los cultivos más importantes del mundo y las enfermedades afectan significativamente la cantidad y calidad del cultivo. Las plantas de maíz son susceptibles a una variedad de enfermedades causadas por patógenos como hongos, bacterias, virus y nematodos, que pueden provocar una reducción del rendimiento de los cultivos, una reducción de la calidad del grano, como también la propagación de enfermedades durante la temporada de crecimiento. Las enfermedades de los cultivos aumentan el costo de la producción agrícola porque se deben implementar estrategias adicionales para controlar los microorganismos patógenos y reducir su impacto en los cultivos. (Orbera et al., 2014, pág. 2)

Las condiciones ambientales, los tipos de suelo y la susceptibilidad material contribuyen a la aparición de enfermedades y las enfermedades virales contribuyen a las condiciones para la migración, el asentamiento y la supervivencia de los insectos vectores. (Agudelo & Sarria, 2006-2007, pág. 7).

El maíz no sólo es uno de los cereales más cultivados, sino que también es considerado el cereal más importante del mundo en términos de producción (Bisetti, 2023).

En Perú, es bien sabido que el maíz es un alimento básico para la población rural de pocos recursos económicos que utiliza pequeñas parcelas de tierra cada año para cultivar maíz para su propio consumo (ASPROMAD, 2015, pág. 8).

En julio de 2022, la producción de almidón de maíz del país alcanzó las 78.285 toneladas, un 25,8% más que el año anterior, incluyendo Cusco (210,7%), Amazonía (128,9%), Huancavelica (25,5%), Huánuco (14,6%) y La Libertad 9,1%), mientras que Moquegua (-60,8%), Lima (-56,6%),

Cajamarca (-41,4%), Apurímac (-16,9%), Lambayeque (-10,6%), Ayacucho (-10,6%) disminuyeron (INEI, 2022).

En el maíz, las pérdidas de rendimiento y la reducción de la productividad varían en diferentes partes del país dependiendo de los daños causados por las diferentes plagas en el campo, poscosecha y el cambio climático. En años normales, varían del 10 al 20 por ciento en años húmedos, aumentando principalmente debido a la mayor incidencia de pudrición de la mazorca, especialmente en zonas montañosas (Calvo & Jara, 2014).

Las plagas que tienden a ocurrir en el sur del Perú son: *Puranius* sp, *diabrotica* spp, *spodeptera frugiperda*, *capitarsia turbata*, *heliiothis zea*, *pagiocerus frontalis*, las plagas y enfermedades más ocasionales son: *bipolares* sp, *cercospora maidis*, *puccina maidis*, las plagas ocasionales son: *spodoptera frugiperda*, *copitarsia turbata*, *cicadellidos*, *heliiothis zea*, *thrips* sp, ácaros, *diplovia* y *fusarium* sp (Catalán, 2017).

Para controlar eficazmente las enfermedades en los cultivos de maíz, se recomienda tomar medidas de prevención, detección o control en las primeras etapas y en niveles bajos de un brote de enfermedad, en lugar de intentar controlar la enfermedad sólo después de que haya causado daños importantes. (Unisem, 2022)

Mediante la implementación de una aplicación móvil (SickCorn) utilizando un modelo de red neuronal convolucional, busca detectar la *Puccinia sorghi* (roya común) y la *spiroplasma kunkeli* (puka poncho) en plantas de maíz del distrito de Talavera mediante la presentación de fotografías o imágenes de hojas enfermas. Si estos dos virus no se detectan a tiempo, pueden propagarse a diferentes zonas agrícolas, afectando la producción y provocando pérdidas a los agricultores. El distrito de Talavera está ubicado en la provincia de Andahuaylas, en el sur de los Andes, entre 13° 39' 12" S y 73° 23' 18" W en el Mar Mediterráneo a una altitud de 2.830 metros.

1.2. Formulación del Problema

1.2.1. Problema General

¿Cuál es el nivel de precisión de la aplicación móvil (SICKCORN) en la detección de *Puccinia sorghi* (roya común) y *Spiroplasma kunkeli* (puka poncho) en las plantas de maíz, Distrito de Talavera?

1.2.2. Problemas Específicos

- ¿De qué manera se realiza la recolección y preprocesamiento de datos para la detección de *Puccinia sorghi* (roya común) y *Spiroplasma kunkeli* (puka poncho) en las plantas de maíz, Distrito de Talavera?
- ¿De qué manera se realiza la evaluación del modelo en la detección de *Puccinia sorghi* (roya común) y *Spiroplasma kunkeli* (puka poncho) en las plantas de maíz, Distrito de Talavera?
- ¿Cómo implementar la aplicación móvil (SICKCORN), para la detección de *Puccinia sorghi* (roya común) y *Spiroplasma kunkeli* (puka poncho) en las plantas de maíz, Distrito de Talavera?

1.3. Objetivos

1.3.1. Objetivo General

Determinar el nivel de precisión de la aplicación móvil (SICKCORN) en la detección de *Puccinia sorghi* (roya común) y *Spiroplasma kunkeli* (puka poncho) en las plantas de maíz, Distrito de Talavera.

1.3.2. Objetivos Específicos

- Recolectar y preprocesar los datos para la detección de *Puccinia sorghi* (roya común) y *Spiroplasma kunkeli* (puka poncho) en las plantas de maíz, Distrito de Talavera.
- Evaluar el modelo en la detección de *Puccinia sorghi* (roya común) y *Spiroplasma kunkeli* (puka poncho) en las plantas de maíz, Distrito de Talavera.

- Desarrollar la aplicación móvil (SICKCORN) para la detección de *Puccinia sorghi* (roya común) y *Spiroplasma kunkeli* (puka poncho) en las plantas de maíz, Distrito de Talavera.

1.4. Justificación

La producción de maíz es fundamental para la alimentación y la economía de diversas zonas agrícolas, entre ellas el distrito de Talavera. No obstante, las enfermedades provocadas por agentes patógenos como la *Puccinia sorghi*, responsable de la roya común, y el *Spiroplasma kunkeli*, una bacteria transmitida por la cigarrita que ocasiona el enanismo del maíz, constituye una importante amenaza para la productividad y la calidad del grano.

El maíz es el cultivo más importante en términos de distribución en el Perú, con aproximadamente 520.000 hectáreas de maíz cultivadas en todo el país cada año, y aproximadamente 82.000 familias dependen directamente de este cultivo, lo que demuestra su importancia socioeconómica para el país. (García, 2020)

A pesar de los esfuerzos del productor, la detección temprana y precisa de estas enfermedades sigue siendo un desafío propenso a errores, por lo que a menudo conduce a pérdidas significativas de la producción. Además, las técnicas tradicionales de detección suelen ser costosas y requieren tiempo, debido a que requiere equipo especializado, como también complicada ya que el agricultor debe tener conocimientos de estas afecciones, por lo que debe ser capacitado constantemente ya que las enfermedades van evolucionando con el tiempo.

Por ello, este estudio propone el desarrollo de una innovadora aplicación móvil denominada “SickCorn” que permitirá a los productores de la región de Talavera detectar de forma rápida y precisa la *Puccinia sorghi* (*Puccinia vulgaris*) y *Spiroplasma kunkeli* (puka poncho) en plantas de maíz.

La implementación exitosa de esta aplicación traerá importantes beneficios. En primer lugar, facilita la detección temprana y el seguimiento de enfermedades y promueve la

implementación oportuna de medidas de control. Esto, a su vez, ayuda a reducir las pérdidas de rendimiento y a mantener plantas de maíz sanas, contribuyendo así a la seguridad alimentaria y la estabilidad económica.

Además, la aplicación móvil "SickCorn" podría presentarse como una herramienta asequible de bajo coste para los productores agrícolas. Al aprovechar la amplia presencia de dispositivos móviles. En consecuencia, este enfoque democratizaría el acceso a la tecnología de detección de enfermedades en el sector agrícola, lo que a su vez fortalecería la capacidad de respuesta de los agricultores ante las amenazas fitosanitarias.

En conclusión, la investigación propuesta tiene como objetivo abordar una necesidad en la agricultura del distrito de Talavera al desarrollar una solución innovadora y práctica para la detección de enfermedades en las plantas de maíz. Se prevé que la investigación resulte beneficiosa para los agricultores de la región, y también que pueda tener impacto positivo en la promoción de la seguridad alimentaria como el fomento del desarrollo sostenible.

1.5. Limitaciones

La aplicación móvil para la detección de *Puccinia sorghu* (roya común) y *Spiroplasma kunkeli* (puka poncho) en plantas de maíz puede ser una herramienta valiosa para los agricultores y profesionales agrícolas, sin embargo, tiene algunas limitaciones en cuanto a la obtención de datos debido a la temporada de producción de maíz, ya que es un producto que se produce anualmente. Además de ello, la variabilidad de las condiciones ambientales produce alteraciones en el entorno, por lo que afecta la condición de la calidad de las imágenes, por ende, afecta a la precisión de la detección, debido a que el clima puede influir en la apariencia de las plantas y enfermedades, dificultando el análisis. Por otro lado, la eficacia de la aplicación puede depender de las capacidades del dispositivo móvil, incluida la calidad de la cámara y las capacidades de preprocesamiento.

Capítulo 2

Antecedentes

2.1. Antecedentes Internacionales

Carranza & Parrales (2018), Escuela Superior Politécnica del Litoral, con su investigación titulada "Esquematizar un sistema de control de enfermedades y plagas que afectan al cultivo de maíz, mediante el monitoreo continuo con un dron e información de estas por medio de un aplicativo móvil", este proyecto busca beneficiar a agricultores y técnicos agrícolas mediante la inspección periódica de cultivos de maíz para detectar enfermedades y plagas. Las imágenes capturadas por el dron se comparan con datos registrados en la base de datos en la aplicación para detectar enfermedades.

Caiza & Lema (2020), Universidad Técnica de Cotopaxi, llevaron a cabo su investigación titulada "Aplicación móvil para la detección y medición de altura en plantas de maíz utilizando Deep learning", El objetivo principal es desarrollar una aplicación móvil que utiliza aprendizaje profundo, redes neuronales convolucionales y algoritmos de medición de altura 2D para detectar y medir la altura de plantas de maíz negro Sapon utilizando imágenes digitales de plantas de maíz negro Sapon. Modelo Después del entrenamiento y evaluación de la precisión de detección. . alcanzó el 98,75% y el margen de error en la medición de la altura fue de 1,97. La aplicación móvil fue desarrollada utilizando el método Crisp-dm y Mobile-D. Además, el modelo también pasó la tecnología de validación cruzada.

Bravo (2020), Instituto del Tecnológico Nacional de México, en su investigación titulada "Detección automática del daño causado por el gusano cogollero en la planta de maíz", cuyo objetivo es diseñar y desarrollar una herramienta basada en tecnología de visión por computadora que pueda identificar daños por lombrices en las plantas de maíz y así determinar si hay daños dentro de la cámara de la planta de maíz. Luego se realizó el procesamiento de imágenes digitales, donde se

utilizó el método outsu para la técnica de segmentación para binarizar y clasificar las hojas de las plantaciones de maíz de campo utilizando: Clasificador KNN Entre las 960 imágenes utilizadas para la evaluación de la clasificación, estaban dañadas y. El resultado de clasificación para imágenes no dañadas fue del 88%, mientras que la tasa de error de clasificación fue del 12%.

2.2. Antecedentes Nacionales

Ordoñez & Bustamante (2018), Universidad Privada del Norte, llevaron a cabo su investigación titulada "Desarrollo de una aplicación móvil que permita la detección y sugerencia de tratamientos para enfermedades en cultivos de maíz". El principal objetivo de este estudio es desarrollar una aplicación práctica diseñada para identificar y recomendar tratamientos para enfermedades de los cultivos de maíz. La aplicación utiliza los principios, técnicas y algoritmos del procesamiento de imágenes digitales, utilizando la biblioteca OpenCV 3.1 como herramienta principal. Además, la aplicación móvil se desarrolla según diferentes etapas del ciclo de vida iterativo mediante la captura de imágenes con un teléfono móvil y continúa con un preprocesamiento que utiliza un filtro gaussiano para la reducción de ruido, un algoritmo de detección de bordes Canny que se utiliza para la segmentación en un área de región para describir regiones e identificar enfermedades utilizando métodos teóricos de toma de decisiones. La aplicación se probó tanto en ambientes controlados como reales. Analizando cultivos de la Hacienda Roque Ángulo en la provincia de Ascope, obtuvo una precisión del 94.34%, desempeño (exactitud) del 94.94%, especificidad del 66.67% y sensibilidad del 100% con un análisis de 67 plantas de maíz.

Córdova & Periche (2022) Universidad Cesar Vallejo, con la investigación titulada "Detección de plaga cogollero (*Spodoptera frugiperda*) mediante un sistema de reconocimiento de imagen en cultivo de maíz, Tambogrande Piura 2021", El objetivo de este proyecto es implementar un sistema de reconocimiento de imágenes para detectar la plaga *Spodoptera frugiperda* en cultivos de maíz. El estudio utilizó métodos de recolección de datos como la observación y las encuestas, y

se evaluaron nueve cultivos mediante el método del diente de sierra. Los resultados mostraron que la eficiencia de detección aumentó del 2% al 23%, lo que indica que el sistema puede detectar plagas de manera efectiva en los cultivos de maíz.

Salinas & Alva (2023), Universidad Privada Antenor Orrego, con su investigación titulada “Detección de plagas y enfermedades en los cultivos de maíz utilizando procesamiento de imágenes con redes neuronales convolucionales en el distrito de Cascas La Libertad, 2022”, El objetivo es entrenar y evaluar 4 modelos de redes neuronales: ResNet50, EfficientNetB0, MobileNetv2 y DesNet201, a medida que se recolectaron imágenes de clasificación de tizón, roya común, lombrices y plantas sanas y se les aplicaron métodos de procesamiento de datos, tales como: Filtros mejorados y funciona para mejorar el aprendizaje del modelo, lo que da como resultado que la red Desnet201 funcione bien con una precisión de gusanos de 0,97, una precisión de roya común de 0,95, una precisión de pudrición de hojas de 0,98 y una precisión de plantas sanas de 0,73. La precisión se probó utilizando la matriz de incertidumbre y se concluyó que con suficientes imágenes y técnicas apropiadas, el modelo puede proporcionar resultados.

Capítulo 3

Marco Teórico

3.1. El Maíz

El maíz, conocido científicamente como "*Zea mays*", es una planta que forma parte de la familia de las gramíneas. Este cultivo se produce a nivel global debido a su valor nutricional y a la variedad de formas en las que puede ser utilizado.

El maíz es el cultivo más popular en términos de promoción peruana. Según estimaciones, en nuestro país se cultiva maíz en cerca de 520 hectáreas de tierra cada año, y unas 82.000 familias dependen directamente de él, lo que demuestra su importancia económica para el país (García, 2020).

En Perú, el maíz se cultiva en una amplia gama de altitudes, hasta los 4000 metros sobre el nivel del mar. Esta planta presenta una importante diversidad genética en rasgos como dureza, color, tamaño de grano, altura de la planta, superficie foliar, etc., lo que la hace muy importante (Navarro & Piña, 2021).

3.1.1. Origen del Maíz

Según la investigación llevada a cabo por científicos peruanos y extranjeros, el Perú, al igual que México, se considera la cuna mundial del maíz. Sin embargo, el Perú destaca por contar con una mayor diversidad de variedades silvestres de este cereal (Grobman, 2012).

El maíz en mazorca destaca por su singularidad entre los cereales. El registro fósil de los primeros maíces de varias partes de México muestra una transición de espiguillas parecidas a teosintle con algunos granos fácilmente quebradizos a mazorcas de maíz con muchos granos adheridos al raquis.

Desde culturas antiguas como los olmecas y teotihuacanos en Centroamérica hasta los incas y quechuas en la región de los Andes en América del Sur, el maíz ha sido esencial para la historia

y el desarrollo de las civilizaciones en América. En el siglo XX y principios del XXI, el maíz es un grano muy importante en muchos sectores económicos alrededor del mundo. El maíz se utiliza en los países desarrollados como alimento, para la producción de alimentos procesados y, recientemente, para la producción de hidrocarburos. Una gran cantidad de maíz se utiliza para consumo humano en América Latina y países africanos, porque es necesario para la vida de agricultores y aldeanos. Sin embargo, debido a la situación económica, la marginación y la pobreza de estas personas, se sospecha que la variedad de maíz se perderá porque los países económicos dedicados a estas personas están disminuyendo. Para solucionar este problema es necesario replantar maíz con conocimiento de su origen y diversidad (Serratos, 2009).

Esta planta, científicamente conocida como maíz, es originaria de América, y se cree que su creación y cultivo comenzó hace miles de años en México y Centroamérica. Este grano fue importante para civilizaciones antiguas como los mayas y los aztecas.

3.1.2. Ciclo de Vida del Maíz

Barandiarán (2020) señala que el ciclo de vida del maíz comprende un proceso que se extiende a lo largo de varios meses y se divide en diversas etapas principales, que incluyen:

- **Germinación:** La vida del maíz inicia cuando germina la semilla, en este proceso está involucrado la absorción de agua como la activación de la semilla para comenzar a desarrollar un brote, las raíces comienzan a crecer en busca de nutrientes y agua, mientras el tallo emerge.
- **Fase vegetativa:** La planta de maíz se centra en el crecimiento vegetativo, el tallo y las hojas se desarrollan rápidamente, durante esta fase la planta no produce espigas o mazorcas, sino se centra en establecer una base sólida.
- **Floración:** Durante la etapa de floración, la planta de maíz sigue su desarrollo, manifestando la aparición de inflorescencias masculinas y femeninas. Las inflorescencias masculinas se sitúan en la parte superior de la planta, liberando polen

que es transportado por el viento para fertilizar las inflorescencias femeninas, las cuales se encuentran más abajo en la planta.

- **Desarrollo de mazorcas:** Luego de la fertilización exitosa, las inflorescencias femeninas se desarrollan en mazorcas de maíz, en la que cada mazorca tiene un número de granos de maíz que se desarrollan a partir de los óvulos fertilizados, la planta continúa creciendo en altura y desarrollando hojas adicionales.
- **Maduración:** Durante esta etapa, los granos de maíz siguen su desarrollo, mientras que tanto las hojas como el tallo de la planta comienzan a secarse y adquieren un tono amarillento. Este proceso culmina cuando los granos de maíz alcanzan su madurez y se secan por completo dentro de la mazorca.
- **Cosecha:** La planta de maíz llega al punto de estar completamente madura y los granos alcancen su punto óptimo de humedad, es el momento de la cosecha, las mazorcas se recolectan de la planta y se separa de la paja.
- **Ciclo de siembra:** Después de la cosecha se prepara el terreno y se siembran nuevas semillas de maíz u otro producto, en la provincia de Andahuaylas se prepara el terreno y se espera hasta la siguiente primavera para cultivar debido a la helada y el friaje.

3.1.2.1. Órganos vegetativos

- **El sistema radicular:** Se compone de dos tipos de raíces: raíces semilleras y raíces adventicias. Las raíces de las semillas se desarrollan a partir de semillas durante la germinación e incluyen raíces radiales y raíces laterales. Las raíces adventicias, en cambio, se forman en la primera parte de la planta y se extienden en forma de cono invertido, tocando el suelo y creciendo por encima del mismo.
- **El tallo:** El maíz tiene tres funciones principales: apoyo a las plantas, transporte de nutrientes y almacenamiento de carbohidratos. El número de nudos y nudos que forman

el tallo varía de una planta a otra, influyendo así en su desarrollo. El crecimiento del tallo se logra mediante el alargamiento de las células internodales; en condiciones adversas, este retraso puede ser mínimo.

- **Las hojas:** Las hojas del maíz crecen de las vainas del tallo. Su rango depende del tamaño de los nodos. Estas hojas se distribuyen a lo largo del tallo; Además, tienen forma alargada y lanceolada, y la parte media está en vena. Su morfología es diferente: hojas rectas con muchas hojas y hojas largas con hojas pequeñas que permiten la exposición al sol.

3.1.2.2. Órganos Reproductivos

- **La panoja:** Cuando las semillas de maíz están en sus etapas finales de crecimiento, todas las hojas han crecido, las mazorcas comienzan a crecer y el polen comienza a producirse. Este polen se libera de las ramas primarias y secundarias y es transportado por el viento para la polinización. Se estima que el 90% del polen cae a menos de 50 metros de una planta.
- **La mazorca:** Se origina a partir de yemas axilares en los nudos, produciendo tallos cortos, delgados y flexibles en los que crecen las brácteas, las hojas flexibles que protegen el grano en crecimiento. Inicialmente, la mazorca tiene el hierro básico para formar dos granos, lo que da como resultado una flor con dos hojas, solo una sobrevive y la otra produce granos. El tamaño de las mazorcas se ve afectado por las condiciones ambientales en las que se cultivan las plantas, y la alta densidad de plantas provoca una reducción en el tamaño de las mazorcas.
- **El grano de maíz:** Es el fruto llamado cariósido, donde la pared del saco embrionario se une con la semilla. Consta de tres partes principales: el pericarpio, que es la capa exterior que protege el material interior; de arroz..

3.1.3. Desarrollo de la Planta

El desarrollo del cultivo implica una secuencia obligatoria de etapas o fases que ocurren en un orden preciso e irreversible, marcando el inicio de nuevos órganos. Este proceso, cualitativo por naturaleza, se conoce como ciclo de desarrollo y abarca desde la germinación de la semilla hasta la floración y formación del fruto. Esta etapa se divide en dos fases principales: desarrollo vegetativo y desarrollo reproductivo (Fassio et al., 1998).

Según Barandiarán (2020), El crecimiento de la planta de maíz se divide en dos etapas fenológicas principales: vegetativa (V) y reproductiva (R). La vegetación comienza con la aparición de las plántulas y está marcada por el brote de cada hoja de la planta, que se caracteriza por la aparición de una lígula. Estos estados se numeran V1, V2, etc., hasta llegar al estado VT, que es similar a la aparición del mieda. Mientras tanto, las etapas reproductivas van desde la R1, que se caracteriza por la aparición del estigma, hasta la R6, que se relaciona con la madurez psicológica.

3.1.3.1. Estado Vegetativo

- En la etapa V1 del desarrollo del trigo, emerge la primera hoja embrionaria. La zona de cultivo se encuentra entre 2,5 y 3,8 centímetros bajo tierra. El meristemo apical es donde se forman las partes vegetativa y reproductiva de la planta.
- La etapa V3 del desarrollo del maíz es el área de cultivo que aún se encuentra bajo tierra. Aunque el daño a las partes aéreas de la planta no la mata y no afecta significativamente la fruta, las hojas y mazorcas de la planta comienzan a formarse en este momento. Por eso, es importante controlar los insectos que pueden atacar las plántulas y asegurarse de que no compitan con el césped.
- En la etapa V5 de crecimiento del maíz, el área de cultivo es en el suelo. En este punto finaliza la formación de hojas y espigas y comienza el proceso de espiga. Las plantas

suelen crecer unos 20 cm en esa época, por lo que es importante mantener el jardín libre de malas hierbas en esa época.

- En la etapa V6 del maíz, la zona de crecimiento consiste en el meristemo de la espiga y la superficie del suelo. En esta etapa el crecimiento se acelera, lo que genera más nutrientes para la planta
- Durante la etapa V9 de la planta de maíz, las mazorcas y los tallos crecen rápidamente. Además, los cogollos en las laderas comienzan a crecer desde varios lugares sobre el suelo.
- Estado V12: Se define el número de hileras, pero se determina el número de semillas (óvulos) y el tamaño de la espiga, pero no el número de semillas por hilera, definido en el estado V17. La cantidad de semillas y el tamaño de la espiga disminuyen debido a la falta de nutrientes y humedad.
- Estado V15: Es importante que el campo tenga un suministro continuo de agua, al menos hasta una semana después de la condición R1, porque la falta de agua reducirá el rendimiento.
- Estado V18: mazorcas visibles y sostenibles
- Estado VT: la planta alcanza su tamaño máximo e inicia la antesis.

3.1.3.2. Estado Reproductivo

- Estado R1: El estigma se poliniza y las brácteas alcanzan su tamaño máximo entre R1-R2. La humedad insuficiente afectará la polinización e inhibirá el desarrollo del grano.
- Estado R2 o ampolla: La mazorca alcanza su tamaño máximo, el estigma comienza a secarse, el embrión es visible y rápidamente comienza a acumularse materia seca, y el contenido de humedad del grano es del 85%.

- Estado R3: Unas tres semanas después de la floración, el embrión crece rápidamente, con la misma acumulación de materia seca y el contenido de agua es del 80%.
- Estado R4 o pastoso: Aproximadamente 4 semanas después de la floración, comienzan a formarse hojas embrionarias y raíces de semillas, se acumula materia seca y las semillas en la vaina comienzan a encogerse con un contenido de humedad del 70%.
- Estado R5 o dentado: Aproximadamente siete semanas después de la floración, los granos comenzaron a secarse a partir de la parte superior, con un contenido de humedad del 75%.
- Estado R6 o madurez fisiológica: Los granos alcanzan su total desarrollo, el contenido de humedad está entre el 30-35%, y están listo para su cosecha.

3.1.4. Variedades del Maíz

La evolución de las razas se origina desde épocas antiguas, la evolución de las razas de maíz ha estado ligada al papel fundamental desempeñado por los agricultores, relacionadas con la producción y el mantenimiento de las semillas, conservando los recursos genéticos mediante la selección para la siembra de variedades específicas. Además, los agricultores son los principales usuarios de su producto final. La raza Confite Morocho es una de las variedades más antiguas cultivadas en Ayacucho, donde se han descubierto fragmentos de maíz de aproximadamente 6.000 años. La raza Ancashino es una de las razas de maíz más variables de la sierra, siendo una de las razas de maíz más variables de la sierra. Las variedades de esta raza varían en función del tono del grano y su adaptación a diversas circunstancias ambientales.

Salhuana (1942) La raza se define como un conjunto de individuos que comparten características morfológicas y fisiológicas comunes, determinadas por genes compartidos. La clasificación racial del maíz se realizó mediante datos biométricos de las colecciones de maíz en Perú, teniendo en cuenta características de la planta, la panoja, tanto externas como internas de la

mazorca, el grano y aspectos citológicos. Este procedimiento permitió establecer seis grupos distintos en concordancia con la evolución del maíz en el territorio nacional.

3.1.4.1. Razas Primitivas.

Las peculiaridades predominantes de estas razas son la precocidad de las plantas, las mazorcas pequeñas, las plumas largas, los granos pequeños, las corontas delgadas, la estructura simple de la coronta, las cúpulas grandes y la escasa inundación del tejido del raquis.

Confite Morocho: Esta variación de maíz se distingue por su escasa altura de aproximadamente 1.30 metros, de tonalidad rojizo y de maduración temprana, florece a los 116 días después de la siembra. No presenta macollo, y posee alrededor de 12 hojas en promedio, las cuales son pequeñas y angostas. La mazorca es pequeña y tiene una forma cilindro-cónica, con un tenue ahusamiento en la punta y en la base. Los corontas de sus corontas son delgadas y contienen granos dispuestos en forma de espiguillas que tienden a separarse. En general, el color predominante de los granos es rojo, representando aproximadamente el 70%, mientras que un 3% son incoloros. Esta variación se cultiva en áreas que oscilan entre 2500 y 3000 metros sobre el nivel del mar (Grobman et al., 1961).

Confite Puntigudo: La variación de maíz se distingue por su escasa altura, de aproximadamente 1.16 metros, de tonalidad rojizo y con macollos. Por lo general, dispone de aproximadamente 9 hojas y es de maduración temprana. En localidades de altitud elevada, florece a los 137 días después de la siembra. Mientras que, en localidades de altitud media y baja, este proceso se produce más rápido, ocurriendo a los 115 y 84 días respectivamente. Las mazorcas son reducidas y ligeramente ahusadas en ambos extremos, con un promedio de 16 hileras. Se cultivan en altitudes que van desde los 2500 hasta los 3500 metros sobre el nivel del mar, y su área de distribución es una de las más amplias entre las razas serranas. Es más robusta que otras razas debido a su reducido número de hojas de panca que se abren cuando la mazorca se encuentra madura. Suelen tener un aspecto puntiagudo, con endospermo blanco. Esta variedad se distingue

por tener una macolla más profunda, una longitud mayor y un mayor número de hileras. El confite puntiagudo guarda una gran similitud con la raza Pisanckalla de Bolivia y Chile, el Canguil de Ecuador y el Pichinga Aristado de Paraguay (Grobman et al., 1961).

Kculli: Esta variación de maíz se distingue por su reducida altura, específicamente de 92 centímetros, y su maduración temprana se produce a los 60 días después de la siembra. Al contrario de otras variedades, no presenta macollos, pero suele tener alrededor de 10 hojas. La mazorca es corta y posee una forma esférica cónica, con una punta ligeramente ahusada y un ancho intermedio. Contiene una cantidad aproximada de 12 hileras, y aproximadamente 8 pancas. Los granos son largos y frecuentemente anchos, con un endospermo blanco y harinoso y una aleurona morada en ocasiones. El pericarpio y la coronta presentan tonalidades de cereza a morado oscuro. Dado que se emplea con frecuencia como colorante natural en alimentos, bebidas y mazamorras, esta variación se cultiva habitualmente en altitudes superiores a los 3000 metros sobre el nivel del mar. Su distribución geográfica es similar a la de las variedades Confite Morocho y Confite Puntiagudo. También comparten algunas características, como un alto índice de venación, pocas ramificaciones y un pedúnculo de panoja largo (Grobman et al., 1961).

Confite Puneño: La raza peruana más pequeña mide 56 cm de altura y presenta un intenso color rojo purpura intenso con seis hojas sin macollos. La mazorca florece a los 90 días y suele tener una mazorca pequeña de forma ovoide, sin ahusamiento en la base, lo que la hace más ancha en la base. Tiene 12 disposiciones irregulares de granos y un péndulo delgado, con granos pequeños, pero más largos, gruesos y anchos que el Confite puntiagudo. La raza de maíz que se cultiva entre los 3600 y 3900 metros cuadrados en la meseta del Collao, situada en el Lago Titicaca, en Perú y Bolivia (Grobman et al., 1961).

Enano: Esta variedad de maíz a menudo alcanza una altura promedio de 1.36 metros y se caracteriza por su maduración muy temprana, similar a las razas más antiguas de la costa, como el Mochero y el Chapareño. Esta variedad tiene unas 13 hojas largas y angostas, sin presencia de

macollos. Las plantas tienen un intenso color verde y están protegidas por una tenue pubescencia de pelos de textura media. Las mazorcas suelen ser pequeñas con forma cónica, de alrededor de 16 hileras de granos dispuestas de manera irregular. El pedúnculo es bastante ancho en comparación con el diámetro de la coronta. Los granos son resistentes, reducidos y de tonalidad blanco terroso, con una aleurona incolora. En lo que respecta a su apariencia, el centro de la mazorca exhibe un 20% de tonalidad roja y un 80% de tonalidad marrón. Esta variedad se cultiva típicamente en la ceja de selva a una altitud de 270 metros sobre el nivel del mar. Debido a que la morfología de la mazorca de la variedad Enano guarda similitudes con el Confite Puneño, especialmente en lo que respecta a la forma y dimensiones de la mazorca, al tipo y tamaño de los granos (Grobman et al., 1961).

3.1.4.2. Razas Derivadas de las Primitivas.

Dado que se originaron a partir de selecciones raciales antiguas o poblaciones híbridas derivadas de cruzamientos entre maíces reventadores primitivos y sus derivados inmediatos, estas razas de maíz siguen una secuencia cronológica aproximada a su origen. Se suele suponer que se originó en la época precolombina.

Chapareño: Se caracteriza por tener alrededor de 12 hojas y florecer aproximadamente a los 80 días de su siembra, este tipo de raza de maíz. Las mazorcas son reducidas y de forma cilindrocónica, con un diámetro de 4 cm y un largo de 11 cm. Los granos de estas mazorcas suelen tener un tamaño mediano, alargados y delgados, con un endospermo generalmente blanco y harinoso, aunque a veces puede tener un tono amarillento. La aleurona presenta una tonalidad púrpura, mientras que el pericarpio presenta una tonalidad incolora y la tusa presenta una tonalidad blanca. Esta raza puede ser cultivada en altitudes comprendidas entre 10 y 500 metros sobre el nivel del mar. (Grobman et al., 1961)

Chullpi: Las plantas de maíz son altamente elevadas, debido a su altura de aproximadamente 2.00 metros, con un tonalidad que oscila entre rojizo y verde. Las mazorcas son cortas y tienen una forma ovoide, con una disposición facetada, contienen aproximadamente 18

hileras de granos. Los granos son largos y delgados, con un endospermo vitreo arrugado que tiende a ser dulce, mientras que el pericarpio es incoloro, mientras que la tusa es predominantemente blanca. La especie de esta especie habita en las montañas de la sierra peruana, ubicadas entre 2400 y 3400 metros sobre la superficie del mar. Bolivia, Ecuador, Chile, Colombia y México poseen rasgos similares. (Grobman et al., 1961)

Paro: Se trata de plantas que alcanzan una altura de 1.30 metros y son de tonalidad rojo, cuyas mazorcas son pequeñas y cónicas de 12 cm de largo y 5 cm de diámetro, con 14 a 16 hileras irregulares. Los granos son delgados y acuminados, el endospermo es harinoso blanco y el pericarpio con la tusa tiene una variedad de tonalidades. (Grobman et al., 1961)

Morocho: La planta de maíz exhibe características superiores al confite morocho. Es habitual alcanzar una altura de aproximadamente 1.60 metros y ostenta un tonalidad rojo claro. No contiene macollos y posee una cantidad aproximada de 9 hojas. Esta variación florece a los 143 días posteriores a su sembrado. Las mazorcas son delgadas y cilíndricas, de aproximadamente 15 cm de longitud, y albergan diez hileras irregulares de granos redondeados, con un endospermo amarillo cristalino en el exterior y harinoso en el interior. El pericarpio no tiene un color y, por lo general, la tusa es blanca. El cultivo de esta especie es habitual en altitudes que oscilan desde los 2000 hasta los 3500 metros sobre el nivel del mar (Grobman et al., 1961).

Huancavelicano: Las plantas poseen una altura de 1.40 metros y no poseen macollos. Sus mazorcas son delgadas y cilindro cónicas de 12 cm de largo y 5 cm de diámetro, con 8 hileras regulares y granos anchos, gruesos y puntiagudos, ligero pico con endospermo blanco harinoso, una capa cristalina y amarilla. Las flores florecen entre 120 y 140 días. (Grobman et al., 1961).

Ancashino: La planta alcanza una altura de 1,40 metros, suele ser de color rojo pálido a verde y tiene 11 hojas. La floración se produce a los 140 días. Las orejas son cónicas, de diámetro decreciente hacia la punta, de 15 cm de largo y 5 cm de diámetro, con un promedio de 14 hileras

irregulares. Los granos son alargados, muy suaves, con un endospermo blanco polvoriento, y el pericarpio y la hierba son de una variedad de marrones. (Grobman et al., 1961)

Piscacorunto: Es una planta de tamaño mediano, que alcanza los 1,50 metros de altura, de color púrpura rojizo y 12 hojas y cogollos. Florecerá después de 130 días. Las orejas son pequeñas, de forma cónica a esférica y cilíndrica, de 12 cm de largo y 5 cm de diámetro, dispuestas en 12 filas regulares. Los granos son grandes, largos y anchos, con un endospermo blanco polvoriento y aleurona moteada de color púrpura. (Grobman et al., 1961)

Cuzco cristalino amarillo: Este tipo de planta puede alcanzar una altura de 1,60 m y tiene un color rojizo. En 130 días florecen 12 hojas y flores. Las orejas son cónicas, de 15 cm de largo y 5 cm de ancho, y constan de 8 partes regulares. Los granos son planos, grandes y redondos con endospermo cristalino amarillo en el exterior y harina blanca en el medio. Vive en la Cordillera de los Andes y suele cultivarse a altitudes de 3000 metros. (Grobman et al., 1961)

Cuzco: Es una planta de color rojizo que alcanza una altura de 1,60 metros y florece en 148 días. Su pistón es cilíndrico y tiene forma de huevo, con un espesor de base y punta de 15 cm de longitud y 5 cm de diámetro. Hay ocho hileras regulares y los granos son planos, grandes y redondos con un endospermo blanco polvoriento y un pericarpio incoloro, blanco o rojo. El cultivo se realiza a gran altura, entre 2400 y 3300 metros sobre el nivel del mar. (Grobman et al., 1961)

Granada: Estas plantas de maíz son pequeñas, miden alrededor de 1,36 metros de altura, de color rojo brillante o violeta y tienen alrededor de 10 hojas pequeñas y estrechas. La floración dura unos 140 días después de la siembra, y la panícula es pequeña y redonda como una granada, de unos 8,58 cm de largo y 5,16 cm de diámetro. Estas mazorcas contienen alrededor de 12 hileras irregulares de granos largos y anchos con endospermo blanco polvoriento. Esta variedad se cultiva en altitudes elevadas de 2600 a 3300 metros sobre el nivel del mar. (Grobman et al., 1961)

3.1.4.3. Razas de Reciente Derivación.

Las variedades que tienen ciertas similitudes y se cree que se derivan de las variedades mencionadas anteriormente se remontan a la época inca y precolombina. Estas variedades se caracterizan por un mejor desarrollo y rendimiento vegetativo.

Huachano: Esta planta alcanza los 2 metros de altura y tiene 13 hojas de color verde. Las panículas son gruesas y cilíndricas, de 15 cm de largo y 6 cm de diámetro, con 13 hileras regulares, y los granos son grandes y planos, con corona cóncava. Se cultiva a gran altura con una altitud media de 45 metros. (Grobman et al., 1961)

San jerónimo huancavelicano: Las plantas de esta variedad suelen alcanzar una altura de unos 1,20 metros y suelen ser de color rojo o morado. El número medio de hojas es de 14 y el período de floración es de unos 134 días. Las panículas miden unos 10 cm de largo y 6 cm de diámetro, con 12 hileras regulares, y los granos son de tamaño mediano, largos y gruesos, con un endospermo blanco polvoriento y aleurona incolora. La piel varía de incolora a roja y marrón. Esta variedad se suele cultivar en altitudes entre 2500 y 3500 metros sobre el nivel del mar. (Grobman et al., 1961)

Cuzco gigante: Las plantas de esta variedad pueden alcanzar una altura de 2,00 metros y se caracterizan por un color rojo pálido. Tiene unas 10 hojas y florece al cabo de 140 días, por lo que puede considerarse una planta de floración tardía. Las orejas son grandes y gruesas, de forma ovalada, de unos 18 cm de largo y 7 cm de ancho, con ocho hileras regulares de núcleos. Estos granos son famosos por su gran tamaño, siendo los más grandes de todas las variedades, gruesos, planos y de forma circular. Esta variedad se cultiva a gran altura con una altitud media de 2800 metros. (Grobman et al., 1961)

Amiláceo: El Choculero altamente productivo es el resultado de cuatro ciclos de selección iterativa de una población creada al cruzar las variedades Cakawacintore y San Geronimo. El primero proviene de los valles altos de México y se caracteriza por su alto rendimiento y estabilidad en diferentes ubicaciones. San Gerónimo, por otro lado, es un cultivo autóctono de grano pequeño,

temprano, amiláceo y de excelente calidad para la producción de maíz (INIA, 2004). Esta variedad se cultiva en una amplia gama de altitudes y se caracteriza por un alto contenido de almidón y granos dentados. La altura de la planta puede alcanzar los 1,5-3 metros. Estas características hacen que esta cepa sea muy valorada por su versatilidad.

Los granos de estos tipos de maíces están compuestos principalmente por almidón de textura natural suave y ligera. Sin embargo, esta característica los hace muy susceptibles al ataque de insectos (INIA, 2020).

3.1.4.4. Razas Introducidas.

Se trata de variedades de maíz traídas de otras regiones y países. A diferencia de las variedades autóctonas de la región, estas variedades contribuyeron a la diversidad genética del país.

Arizona: La planta alcanza una altura de unos 2,80 metros y suele tener un llamativo color verde y un total de 10 hojas. Esta variedad tiene una precocidad moderada y florece al cabo de unos 95 días. Las orejas son de tamaño mediano y cilíndricas, de unos 15 cm de largo y 5 cm de diámetro. Consta de 14 hileras regulares, los granos son de tamaño mediano, planos y aserrados, con un endospermo blanco cristalino que contiene aleurona incolora. El cultivo se realiza a una altitud de aproximadamente 2200 metros. (Grobman et al., 1961)

Cubano dentado amarillo: Las plantas de esta variedad alcanzan una altura de 2,50 metros y tienen una precocidad media, floreciendo aproximadamente a los 95 días después de la siembra. Las orejas son de tamaño mediano y cilíndricas, de unos 15 cm de largo y 5 cm de diámetro. Estas mazorcas suelen tener de 14 a 16 hileras de granos en el medio del tallo. Los granos son de tamaño mediano, con un endospermo amarillo cristalino y una cubierta general polvorienta. Esta variedad se cultiva principalmente en las regiones costeras del norte y centro del país. (Grobman et al., 1961)

Chuncho: La planta tiene un color rojo violeta brillante, alcanza los 3 metros de altura, tiene 15 hojas largas y anchas, un tallo grueso y exuberante y raíces adventicias. Florece tarde, con un periodo de floración de 130-140 días. Las orejas son de tamaño mediano y gruesas, de 18 cm de

largo y 6 cm de diámetro, en forma de cilindro cónico con 16 hileras irregulares. Los granos son de tamaño mediano, planos y aserrados, cultivados. (Grobman et al., 1961)

3.1.4.5. Razas Incipientes.

Estas nuevas variedades están surgiendo actualmente o han sido caracterizadas recientemente. Muestran progreso evolutivo y tienen una distribución geográfica más restringida.

Jora: Las plantas de esta variedad pueden alcanzar una altura de hasta 2,50 metros, son de color verde rojizo y tienen un total de 14 hojas. El periodo de floración suele ser de unos 100 días. La panícula es de tamaño mediano y cilíndrica, de unos 15 cm de largo y 5 cm de diámetro, con 14 hileras de granos dispuestas regularmente. Estos granos son de tamaño mediano y ancho, y el endospermo es blanco, polvoriento o cristalino. El color de la tusa varía entre blanco, rojo y marrón. (Grobman et al., 1961)

Coruca: Las plantas de esta variedad son de tamaño mediano, alcanzando una altura de unos 2,00 m, de color verde claro. Tiene un total de 12 hojas, está medianamente madura y tiene un periodo de floración de unos 100 días. El pistón es grueso, de tamaño mediano, cilíndrico-cónico, de unos 14 cm de largo y unos 6 cm de ancho, con extremos dentados. El endospermo del grano es de color blanco polvoriento, mientras que la aleurona y el pericarpio son incoloros y el pericarpio es marrón o moteado de marrón. La tusa puede ser blanca, roja o marrón. Esta variedad se cultiva principalmente en zonas costeras. (Grobman et al., 1961)

Sarco: Es una variedad emergente, geográficamente similar a la Ancasino y con características altamente productivas. Las orejas de esta raza son largas, anchas y cónicas. El grano suele ser polvoriento y la cáscara suele ser de color marrón a rojo, pero puede ser incolora. (Grobman et al., 1961)

3.1.4.6. Razas Perfectamente Definidas.

Este grupo racial tiene una distribución geográfica limitada y se distingue como un híbrido segregado.

San Gerónimo: Es una planta de rápido crecimiento que crece a una altitud media de 3.200 metros y tiene de 9 a 11 hojas delgadas de color rojo sol. Sus orejas tienen forma de granadas hechas de granos de harina redondos y blancos dispuestos irregularmente. (Grobman et al., 1961)

Amarillo Huancabamba: Son plantas de tamaño mediano con espigas altas. La tasa de maduración es media, las mazorcas son cortas y de diámetro pequeño, los granos son anchos, de grosor medio y largos, el endospermo es amarillo, el pericarpio es rojo y la aleurona es violeta. Se puede cultivar en altitudes elevadas entre 1800 y 2400 metros sobre el nivel del mar.

3.2. Enfermedades en Plantas de Maíz

La enfermedad se define como un proceso o un estado resultante que afecta a un organismo vivo, manifestado por una perturbación perjudicial de su salud. Hay una variedad de factores que pueden causar esta condición, ya sean internos o externos al organismo afectado. (Jara, 2014)

Según Verdeguer (1985), “el término enfermedad aplicado a una planta hace referencia a cualquier condición en la misma que interfiera con su desarrollo normal”.

3.2.1. Enfermedades Foliares.

“Estas enfermedades son provocadas por hongos que tienden a manifestarse en las últimas etapas del cultivo, surgiendo antes o en proximidad a la fase de floración” (Agudelo & Sarria, 2006-2007).

3.2.2. Enfermedades Causadas por Cronistas.

Se refiere a un grupo de enfermedades causadas por microorganismos eucariotas y se identifica por síntomas semihojales, o la aparición de áreas brillantes que contrastan con el color verde de las hojas y cubren aproximadamente la mitad del limbo. (Agudelo & Sarria, 2006-2007)

3.2.3. Enfermedades Causadas por Bacterias:

Si las plantas afectadas permanecen en el campo durante mucho tiempo, la enfermedad se propaga rápidamente mediante la exposición a los insectos y la lluvia. Los síntomas característicos incluyen marchitez vascular, necrosis y pudrición blanda. (Agudelo & Sarria, 2006-2007)

Las enfermedades reducen los rendimientos y reducen la calidad de los cultivos en diversos grados, dependiendo de cuándo afecta la enfermedad a los órganos que ataca y el grado de daño causado. (Quevedo, 2013)

3.2.4. Plagas.

Una plaga se define como cualquier organismo vivo, ya sea animal, microbiano o vegetal, que tiene un impacto negativo directo en la producción agrícola. Esto incluye todos los organismos que son dañinos para las plantas y los productos vegetales. Estos se pueden dividir en tres grupos principales: animales superiores (insectos, ácaros, nematodos, aves, roedores, etc.), microorganismos (viroides, virus, micoplasmas, bacterias, hongos, etc.) y plantas superiores (malezas). Anteriormente, estos organismos se clasificaban en plagas (animales superiores), enfermedades (microorganismos patógenos) y malas hierbas (plantas superiores). (Jara, 2014)

Los microorganismos no deseados como plantas, animales, insectos y microorganismos que interfieren con las actividades humanas se denominan plagas. Estas criaturas pueden causar daño al morder, destruir cultivos alimentarios y dañar propiedades. (NPIC, 2021)

3.2.5. Puccinia Sorghi (Roya común).

La enfermedad prevalece en todo el mundo, especialmente en climas templados subtropicales y regiones montañosas húmedas. A medida que la planta se acerca a la floración, la roya se vuelve más visible y se caracteriza por la presencia de pequeñas pústulas polvorientas en la superficie de ambas hojas. Inicialmente estas pústulas son de color marrón claro, pero a medida que la planta madura, las lesiones se vuelven negras. (Castro & McNab, 2004)

Figura 1

Puccinia Sorghi



Nota. Enfermedades del maíz: una guía para su identificación en el campo, cuarta edición (pg. 8).

Fuente: (Castro & McNab, 2004).

3.2.6. Spiroplasma Kunkeli (puka poncho).

La enfermedad es causada por la bacteria Mollicute, sin paredes celulares, que se transmite por un insecto vector llamado *Dalbulus maydis*. Los síntomas visuales incluyen la aparición de franjas amarillas marchitas en las hojas de las panículas, acortamiento de los entrenudos, enrojecimiento de los márgenes de las hojas, formación de múltiples panículas, engrosamiento de los tallos y polinización inexacta debido a la reducción del llenado del grano. Estos síntomas pueden ocurrir individualmente o en combinación, según el momento de la infección y el híbrido de maíz afectado.

Spiroplasma kunceri es el fitoplasma que causa el puka poncho y típicamente se encuentra en regiones templadas por debajo de los 3.000 metros sobre el nivel del mar, especialmente en cultivos tempranos. Este fitoplasma es transmitido por saltahojas del tipo *Darbrus maydis*, que ataca los cultivos en una etapa temprana y también transmite el virus rayado. Si no se controlan las poblaciones de saltahojas, los daños pueden ser graves y provocar una falta de producción de mazorcas y granos. Además, la jalá afectada por esta enfermedad no es adecuada como alimento

para el ganado. Los síntomas de la enfermedad suelen aparecer como manchas rojizas o amarillentas en las hojas.

Figura 2

Spiroplasma Kunkeli



Nota. Características de la afección *Spiroplasma kunkeli*. Obtenido de la página agro empresario.

Fuente: (Giménez, 2019) (<https://agroempresario.com/publicacion/45738/enfermedades-en-maiz-atentos-a-spiroplasma-en-la-region-central-y-a-mal-de-rio-cuarto-en-el-norte/>)

3.3. Sistemas de Información

Un sistema de información es más que un conjunto de programas y dispositivos informáticos utilizados para la gestión diaria de las actividades productivas. Se ha expandido y evolucionado con el tiempo y se considera una herramienta para reducir la burocracia al facilitar las transacciones. (Hernández, 2003)

Los sistemas de información constan de elementos que recopilan, almacenan, procesan y distribuyen datos para apoyar la toma de decisiones. Estos sistemas permiten a las empresas e instituciones capturar, almacenar y hacer fluir información relevante para funcionar de manera óptima, con el objetivo de mejorar la eficiencia, la productividad y la toma de decisiones.

3.4. Tecnologías Móviles

La definición proporcionada por IBM describe la computación omnipresente como tecnología móvil que evoluciona con los usuarios. Incluye dispositivos móviles de comunicación bidireccional, dispositivos informáticos y la infraestructura de red que los combina. Esta tecnología se caracteriza por la presencia de dispositivos con acceso a Internet, como teléfonos inteligentes, tabletas y relojes inteligentes.

La tecnología móvil incluye una variedad de dispositivos y servicios diseñados para la comunicación con la capacidad de acceder a información sobre la marcha. Esto incluye el uso de redes inalámbricas y aplicaciones móviles, estas tecnologías permiten conectividad constante, acceso a Internet y comunicación en tiempo real, brindando mayor flexibilidad y facilidad de acceso a la información en cualquier momento y lugar.

3.5. Dispositivo Móvil

Baz et al. (2011) Un dispositivo móvil es un dispositivo pequeño capaz de manejar conectividad de red persistente o intermitente, con memoria limitada y destinado a ejecutar una función particular, aunque también puede ejecutar funciones más amplias.

Es un tipo de computadora con memoria, conexión a internet y procesamiento. A pesar de que fue creado para una función particular, también es capaz de realizar otras funciones más amplias. (Díaz, 2017)

Enriquez & Casas (2013) Cualquier software creado especialmente para dispositivos móviles se considera aplicación móvil. El concepto "móvil" se refiere a la capacidad de acceder a aplicaciones, datos y dispositivos en cualquier momento y lugar. Estas aplicaciones se crean teniendo en cuenta las limitaciones de los propios dispositivos.

El término "dispositivos móviles" se refiere a dispositivos electrónicos compactos y portátiles que tienen como objetivo facilitar su transporte y uso cuando una persona está en

movimiento. Su diseño se enfoca en brindar una amplia gama de funciones, que incluyen desde la comunicación hasta el acceso a Internet, así como el entretenimiento y la productividad.

3.5.1. Tipos de Dispositivos Móviles

Baz et al. (2011) Para definir los dispositivos móviles, el equipo de Innovación de Movilidad Global de DuPont estableció los siguientes criterios, teniendo en cuenta la amplia gama de funcionalidades asociadas a ellos..

- Los dispositivos móviles con capacidades de datos limitadas se refieren a los teléfonos móviles tradicionales, que se distinguen por tener una pantalla de texto pequeña.
- El dispositivo móvil de datos básicos: se caracterizan por tener una pantalla de tamaño mediano, menú o navegación basada en iconos.
- Los dispositivos móviles con datos mejorados se distinguen por contar con pantallas de tamaño mediano a grande.

Según Peña et al. (2016) los tipos de dispositivos móviles son los siguientes;

- Teléfonos.
- Personal digital assistant (PDAs).
- Consolas.

3.5.2. Sistema Operativo para Móviles

Un conjunto de programas informáticos que permiten la gestión efectiva de los recursos de un dispositivo móvil se conoce como sistema operativo para móviles. Este software operativo tiene la intención de realizar varias tareas y servir como intermediario entre el hardware del dispositivo y el usuario (Paez, 2021).

El uso de teléfonos inteligentes, que permiten la conexión entre el usuario y la máquina, se ha logrado gracias a que ahora tienen un sistema operativo que funciona dentro del teléfono. (Jaramillo, 2017).

Según Baz et al. (2011) Las habilidades multimedia de los dispositivos y cómo interactúan con el usuario están determinadas por el sistema operativo que se emplea. Estos sistemas operativos incluyen:

- Symbian: El sistema operativo más comúnmente utilizado en smartphones, por ende, el que cuenta con la mayor cantidad de aplicaciones desarrolladas para su plataforma, es actualmente Symbian, este sistema ocupa más del 65% del mercado de sistemas operativos.
- Windows Mobile: Se trata de un sistema operativo desarrollado desde cero que incorpora algunas convenciones de la interfaz de usuario tradicional de Windows.
- Android: Es un sistema operativo móvil creado por Google, fundamentado en Linux y Java, este sistema fue lanzado bajo la licencia Apache
- iPhone os: Se trata de un sistema operativo que es una versión simplificada de Mac OS X, especialmente diseñada para los procesadores ARM. Sin embargo, no permite la instalación de ninguna aplicación que no esté firmada por Apple.
- BlackBerry OS: Es un sistema operativo multitarea que está arrasando en la escena empresarial, en especial por sus servicios para correo y teclado QWERTY.
- Palm OS: Es un sistema operativo que está evolucionando hacia su completa funcionalidad en el entorno de Linux como sistema operativo.

3.5.3. Android

Android es un sistema operativo desarrollado por Google para dispositivos electrónicos y móviles; se basa en sistemas de escritorio como GNU/Linux. (Jaramillo, 2017)

Según (Villalón, 2021) Android es un sistema operativo móvil que no se limita a ese mercado, ya que está presente en varios dispositivos, como tabletas y televisores, gracias a su base en el kernel de Linux, que lo ha permitido expandirse a todas las áreas tecnológicas.

3.5.4. Arquitectura Android

Para satisfacer las necesidades de cualquier dispositivo Android, la arquitectura de Android consta de varios componentes. Un conjunto de bibliotecas de C/C++, presentadas a través de un marco de servicios de aplicación, y un kernel de Linux de código abierto forman parte del software. (Villalón, 2021)

La plataforma para ejecutar aplicaciones Android es la máquina virtual Dalvik (DVM), mientras que el kernel de Linux es el componente principal del sistema operativo del teléfono inteligente. Las partes fundamentales de la arquitectura de Android incluyen:

- Aplicaciones
- Marco de aplicación
- Tiempo de ejecución de Android
- Bibliotecas de plataforma
- Linux Kernel

3.6. Aplicación Móvil

Las aplicaciones móviles, son herramientas de software para teléfonos inteligentes, tabletas y smartphones, creadas en una variedad de lenguajes de programación. Por ser útiles, dinámicas y fáciles de instalar, se destacan. (Calvo, 2023)

"Es un programa de software capaz de llevar a cabo una o varias funciones específicas que el usuario requiera. Por lo general, estos programas son pequeños y especializados" (Milenium, 2020).

Es importante destacar que, aunque todas las aplicaciones son programas, no todas son aplicaciones. Una aplicación es un programa informático que se desarrolla para realizar o facilitar una tarea en un dispositivo informático. (Ccopa, 2019)

Las aplicaciones móviles pueden desplegar una amplia gama de funcionalidades, como la recopilación de datos, la realización de investigaciones, análisis, facilitación de la comunicación y mejora de la eficiencia en diversos campos y disciplinas.

3.6.1. Tipos de Aplicaciones para Dispositivos Móviles

Según Lisandro et al. (2013) presentan tres enfoques para desarrollo de aplicaciones para dispositivos móviles

- **Aplicaciones web:** Este tipo de aplicaciones están destinadas a funcionar en el navegador del dispositivo móvil; son independientes de la plataforma y no requieren adaptarse a ningún ambiente operativo particular.
- **Aplicaciones nativas:** Es necesario considerar el tipo de dispositivo, el sistema operativo y su versión para que estas aplicaciones puedan interactuar con todas las funciones del dispositivo, ya que están diseñadas para funcionar en una plataforma específica.
- **Aplicaciones híbridas:** Se crean utilizando tecnología web y se ejecutan en un contenedor web del dispositivo móvil.

3.7. Inteligencia Artificial

El campo de la inteligencia artificial (IA), que incluye el aprendizaje profundo (Deep learning) y el aprendizaje automático (machine learning), surgió en la década de 1950 y se define como el proceso de automatizar las tareas intelectuales que realizan los seres humanos. (Bonilla, 2020)

Hay muchas perspectivas sobre el término inteligencia artificial a lo largo del tiempo. La habilidad de pensar o de actuar de manera inteligente, donde las máquinas pueden pensar y actuar de manera racional, es el tema de algunas definiciones. Según otros, la inteligencia artificial es la habilidad de emular lo que resultaría de una acción estrictamente racional. (Martínez, 2018)

De acuerdo con Pino et al. (2001), La inteligencia artificial es un campo de la ingeniería y la ciencia que busca entender lo que se conoce como comportamiento inteligente desde una

perspectiva informática; además, se centra en la creación de artefactos que muestren este tipo de comportamiento.

3.7.1. Componentes de la Inteligencia Artificial

3.7.1.1. Aprendizaje Automático (machine learning).

Se basa en algoritmos, los cuales permiten que una máquina aprenda de acciones y datos anteriores. Busca patrones en los datos y, como resultado, toma decisiones más acertadas, lo que resulta en mejores resultados. (Modi, 2021)

Según Pino et al. (2004), El aprendizaje automático es una parte esencial del comportamiento inteligente de los seres humanos; este proceso les permite acumular experiencia y adaptarse a entornos cambiantes, lo que requiere el uso de nuevas estrategias; es un proceso que les permite acumular experiencia y adaptarse a entornos cambiantes.

El aprendizaje automático es una rama de la inteligencia artificial que se centra en la creación de modelos y algoritmos que enseñan a las computadoras a realizar tareas particulares basadas en datos.

3.7.1.2. Aprendizaje Profundo (Deep learning).

Con nodos neuronales conectados en forma de red, se basa en la estructura y funcionamiento del cerebro humano. Esto permite que las máquinas realicen tareas de manera natural, como lo harían los humanos. (Modi, 2021)

Según Serrato (2021), El aprendizaje profundo es una rama del aprendizaje automático que aprende procesando datos mediante redes neuronales artificiales. Las máquinas aprenden a través del refuerzo positivo y negativo de las tareas que realizan; por lo tanto, requieren un procesamiento constante para avanzar; estas redes imitan las redes neuronales biológicas del cerebro humano.

Para hacer predicciones, los algoritmos de aprendizaje automático emplean datos etiquetados y estructurados; esto implica organizar y definir características particulares durante la entrada de datos para el modelo; sin embargo, no siempre es necesario utilizar datos estructurados.

3.7.1.3. Visión Artificial.

Es una tecnología que utiliza máquinas para reconocer imágenes automáticamente y describirlas de manera efectiva y precisa. En la identificación de objetos, el reconocimiento facial, la clasificación, la recomendación, el monitoreo y la detección, utiliza inteligencia artificial y aprendizaje automático para procesar datos con precisión. (Amazon, 2023)

3.7.2. Tipos de Aprendizaje

3.7.2.1. Aprendizaje Supervisado.

El método de aprendizaje que utiliza implica un agente externo, como un maestro o un supervisor; este agente dirige el proceso al determinar la respuesta que se espera que la red produzca a partir de una entrada determinada. (Durán, 2017)

3.7.2.2. Aprendizaje Reforzado.

“Este método es comparable al aprendizaje supervisado; sin embargo, en este caso, cada objeto relacionado con una entrada está acompañado por una nota, que es una evaluación del desempeño de la red en comparación con varias secuencias de entrada. (Durán, 2017)

3.7.2.3. Aprendizaje no Supervisado.

La matriz de pesos y bias se modifican en respuesta a las entradas de la red porque la red aprende a clasificar los patrones de entrada en una serie finita de clases; son aquellos en los que no se dispone de una batería de ejemplos previamente clasificados. (Durán, 2017)

3.7.3. Transfer Learning

El aprendizaje por transferencia es una técnica del campo del aprendizaje automático e inteligencia artificial que implica adaptar (transferir) un modelo previamente entrenado en una tarea para que pueda realizar una tarea relacionada pero diferente.

“El aprendizaje por transferencia intenta mejorar el aprendizaje automático tradicional transfiriendo el conocimiento aprendido en una o más tareas de origen utilizándolo para mejorar el aprendizaje en una tarea” (Torrey & Shavlik, 2010, pág. 23).

Según Yosinski et al. (2014), Un método de aprendizaje automático conocido como transferencia de aprendizaje consiste en capacitar una arquitectura neuronal para una tarea determinada y luego volver a utilizar esa arquitectura para otra tarea similar.

El transferencia de aprendizaje Si las características del modelo aprendidas en la primera tarea son generales o similares a la tarea que realizamos, podemos entrenar nuestro modelo de aprendizaje profundo con un número de datos relativamente pequeño y encontrar soluciones favorables a nuestro problema (Salinas, 2021). Ejemplos de aprendizaje por transferencia:

- Como extractor de funciones fijas: En este método se eliminan las capas de clasificación del modelo de redes neuronales convolucionales (CNN) que se habían entrenado previamente en una base de datos inicial. Para capacitar a un nuevo clasificador, se emplea el resto del modelo como un extractor de características fijas. Por último, se agrega una capa densa que refleje el número de clases presentes en el nuevo conjunto de datos.
- Ajuste fino (Fine-tuning): El nuevo conjunto de datos es necesario para reemplazar y entrenar al clasificador en la parte superior del modelo. Los pesos de la red pre-entrenada también se ajustan, lo que permite la actualización de sus pesos mediante la propagación hacia atrás del error. Esto podría incluir ajustar solo una parte del nivel superior de la red y mantener algunas capas fijas.

3.7.4. Visión Artificial

Según Veintimilla & Sigüencia (2014), La visión artificial, también conocida como visión por computadora o visión técnica, surgió junto con el avance de la inteligencia artificial con el propósito de permitir que una máquina pueda comprender una escena o las características de una imagen y que una máquina pueda interpretar toda la información contenida en los elementos de una imagen.

El objetivo de los sistemas de visión artificial es replicar la visión humana. La falta de conocimiento sobre las áreas exactas en las que el cerebro procesa toda la información visual hace que el sistema visual humano sea difícil de estudiar. (Pérez, 2018)

La visión artificial es un campo de la inteligencia artificial que se enfoca en crear sistemas capaces de comprender, analizar y interpretar información visual obtenida de imágenes y videos. También se conoce como visión por computadora.

3.7.4.1. Tareas de Visión Artificial

- **Clasificación de imágenes:** Las computadoras pueden ver una imagen y clasificar con precisión a qué clase pertenecen gracias a la visión artificial, ya que reconoce las clases y las etiquetas en función de las características visuales que identifica en la imagen.
- **Detección de objetos:** La clasificación se emplea en la visión artificial para distinguir, ordenar y organizar imágenes; se utiliza en procesos industriales y de fabricación para regular aplicaciones autónomas, detectar personas y objetos y enviar alertas cuando sea necesario.
- **Seguimiento de objetos:** Para distinguir y rastrear los elementos que pertenecen a categorías particulares, cada objeto identificado se delimita con un cuadro delimitador en la imagen, lo que permite su seguimiento a través de los diversos marcos o frames de un video. Utiliza modelos de aprendizaje profundo.

- **Segmentación:** Es un algoritmo que utiliza los píxeles de las imágenes para reconocer un objeto al dividirlos en diferentes áreas. Para determinar la forma del objeto, esta técnica rastrea los píxeles, lo que facilita su análisis y etiquetado.

3.8. Preprocesamiento de Imágenes

El propósito del procesamiento digital de imágenes en la visión computacional es mejorar la calidad de las imágenes para su posterior interpretación o uso (Sucar & Gómez, 2008).

Las etapas del procesamiento de imágenes incluyen la captura o adquisición de imágenes, el preprocesamiento, la segmentación, el reconocimiento y la extracción de características. (García, 2008)

3.8.1. Etapas del Procesamiento de Imágenes:

- **Adquisición:** La adquisición de imágenes se realiza mediante los siguientes procesos: La digitalización mediante escáner o la captura a través de cámaras digitales:
 - Proceso de formación de imágenes: Es un procedimiento que proyecta una información luminosa en un plano 2D.
 - Representación de imágenes digitales: Es un vector o una matriz de números binarios (0,1), que se representa como pequeños cuadros en forma de mosaico individual, pixel.
 - Dispositivos de captura: Los dos tipos de dispositivos de captura compiten por una imagen digital, ya sea mediante un escáner o una cámara fotográfica digital, en términos de resolución y calidad.
 - Formatos de almacenamiento: Los datos representados por la imagen se organizan y almacenan mediante un formato de fichero de imagen. Existen varios formatos que se utilizan según las necesidades de almacenamiento, y se distinguen por los niveles de profundidad admitidos, el tipo de compresión y otras características.
- **Preprocesamiento**

- Técnicas de reducción de ruido mediante filtros: Para realizar una operación aritmética con un número de píxeles cercanos, la mayoría de las técnicas de reducción de ruido emplean el método de filtrado espacial, Edamns (20006), que consiste en recorrer toda la imagen, pixel a pixel. Es conocido como ventana, máscara o kernel el conjunto de estos píxeles cercanos; estos últimos pueden ser de 3x3, 5x5 y así sucesivamente. La mayoría de los algoritmos son combinaciones lineales de los píxeles de la ventana; el vector o la matriz de convolución o el núcleo se utilizan para representar un filtro.
 - Filtro Gaussiano: El reductor de ruido Gaussiano se basa en una máscara que se convoluciona con una imagen. Este tipo de máscara elimina detalles de la imagen y la difumina para disminuir el ruido. En el procesamiento de imágenes, una máscara en dos dimensiones es el interés. (Aguilar, 2007)
- Realce de colores (HSV): Como codifica el color en tres partes: saturación (pureza), croma (tonalidad) e intensidad (brillo), se considera que el modelo HSI es el más cercano a la percepción humana. Estos componentes se representan en un espacio tridimensional similar a un cubo, donde cada eje corresponde a un componente primario. Además, el modelo se puede representar como un cilindro, con la altura que corresponde a la intensidad, la distancia al eje central que corresponde a la saturación y el ángulo al croma que corresponde al vértice opuesto; el origen del cubo es el negro y el vértice opuesto es blanco. (Sucar & Gómez, 2008)

- **Segmentación**

Durante la segmentación, la imagen se divide en categorías para clasificar las partes que componen la imagen. Éstas se denominan áreas o regiones de la imagen y cada una consta de propiedades diferentes. Los atributos básicos de la segmentación de imágenes son:

luminancia en imágenes monocromáticas, componentes de color en imágenes en color, textura, etc. La fase de segmentación del procesamiento de imágenes digitales combina muchas técnicas: detección de bordes, umbralización y métodos basados en regiones.

(Gómez & Guerrero, 2016)

3.9. Redes Neuronales Artificiales

Las redes neuronales buscan replicar el esquema neuronal tanto en sus unidades de procesamiento (neuronas) como en sus conexiones y la forma en que se establecen.

Las redes neuronales artificiales se utilizan para estimar o aproximar funciones que pueden depender de una gran cantidad de entradas, muchas de las cuales pueden ser desconocidas.

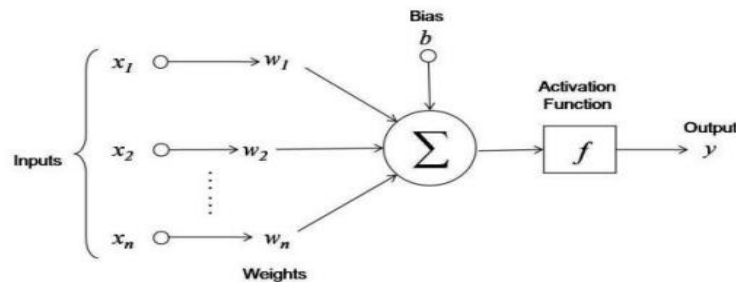
(Bonilla, 2020)

El campo de las redes neuronales artificiales se basa en el estudio de redes neuronales reales en el cerebro y tiene como objetivo resolver problemas aplicando el conocimiento adquirido en casos similares que ya se han resuelto anteriormente. (Nacelle, 2009)

Según Basogain (1998), Las redes neuronales artificiales pueden simular el comportamiento de cualquier función, ya sea lineal o no lineal. Es un modelo paramétrico con un conjunto de parámetros que se ajustan durante el proceso de aprendizaje. Estos parámetros incluyen ponderaciones y sesgos. Las redes neuronales artificiales constan de un conjunto de modelos lineales que, combinados, pueden resolver problemas complejos. Por tanto, se puede concluir que es un modelo de procesador lineal. La estructura básica se muestra en la Figura 3, que muestra la entrada, el desplazamiento, la función de habilitación y la salida.

Figura 3

Estructura Red Neuronal



Nota. Estructura de una neurona artificial. Adaptado del trabajo de fin de grado: Reconocimiento de imágenes mediante redes neuronales convolucionales (pg. 5). Fuente: (Martínez, 2018).

3.9.1. Arquitectura de la Red Neuronal

Las redes neuronales artificiales (RNA) tienen una arquitectura completamente diferente a la de las computadoras tradicionales de un solo procesador, ya que en los sistemas informáticos neuronales cada elemento de procesamiento (PE) puede realizar solo una o, como máximo, varias operaciones. La potencia del procesador en una RNA se mide por el número de interconexiones que se actualizan cada segundo durante el entrenamiento o aprendizaje, ya que la arquitectura de las RNA se basa en la organización de un sistema de procesamiento paralelo en el que se conectan diferentes procesadores. Sin embargo, un procesador es una unidad de procesamiento simple diseñada para sumar muchas entradas con la capacidad de ajustar automáticamente conexiones ponderadas. (Basogain, 1998)

La arquitectura de una red neuronal se refiere a su topología, estructura o patrón de conexión, donde los nodos se conectan a través de sinapsis. El comportamiento de la red está determinado por la estructura de estas conexiones sinápticas, que es direccional. Las neuronas suelen dividirse en unidades estructurales llamadas capas, donde un grupo de capas forma una red neuronal. (Larrañaga et al. 2013)

3.9.2. Tipos de Redes Neuronales Artificiales

Según Google Cloud, una red neuronal consta de neuronas artificiales y unidades de procesamiento que se utilizan para clasificar y verificar información. Las redes con más de tres capas de entrenamiento se denominan "redes neuronales profundas" o "aprendizaje profundo".

3.9.2.1. Las Redes Neuronales Prealimentadas (FF).

En este tipo de red, los datos fluyen a través de múltiples capas de neuronas artificiales en una dirección hasta obtener un resultado. Normalmente, se utiliza un algoritmo de corrección de errores llamado retropropagación, que atraviesa la red hacia atrás para detectar errores y mejorar su precisión.

3.9.2.2. Las Redes Neuronales Recurrentes (RNN).

Al igual que otras arquitecturas de redes neuronales, las redes neuronales recurrentes (RNN) aplican pesos a cada nodo de la red. Los RNN tienen la capacidad única de retener una "memoria" de lo que sucedió en la capa anterior mientras procesan la entrada actual.

3.9.2.3. Las Redes Neuronales Recurrentes de Memoria a Largo/Corto Plazo (LSTM).

Son una forma avanzada de redes neuronales recurrentes (RNN). Mientras que el RNN tradicional tiene dificultades para recordar información durante mucho tiempo debido al problema del desvanecimiento del gradiente, LSTM utiliza dispositivos de memoria que pueden recordar múltiples capas de información para superar este problema. Esto les permite capturar dependencias a largo plazo en datos de secuencia y los hace particularmente efectivos para tareas que requieren una comprensión a largo plazo de las secuencias, como el procesamiento del lenguaje natural y la generación de texto.

Según Bonilla (2020), Las redes recurrentes son modelos de redes neuronales artificiales en las que las conexiones entre unidades forman ciclos dirigidos que crean una secuencia en la que los

caminos a lo largo de los vértices y aristas están definidos por un conjunto determinado de aristas y siguen un orden determinado.

3.9.2.4. Las Redes Neuronales Convolucionales (CNN).

Estas redes se utilizan ampliamente en el reconocimiento de imágenes y utilizan varias capas diferentes para filtrar diferentes partes de una imagen antes de reutilizarlas. El proceso comienza con capas, buscando características simples como bordes y textura. Luego, estas características se combinan en capas posteriores para buscar características más complejas y abstractas, como formas y patrones. Este enfoque de múltiples capas permite a las CNN capturar y aprender de manera eficiente características relevantes de las imágenes.

CNN, abreviatura de red neuronal convolucional, es una red neuronal artificial que imita la corteza visual del cerebro humano para procesar sus capas e identificar diferentes características de entrada. Estas redes contienen varias capas ocultas especiales dispuestas en una estructura jerárquica. (Artola, 2019)

Según Bonilla (2020), Las redes convolucionales son los modelos más utilizados para el procesamiento de imágenes y la visión por computadora porque están diseñadas para imitar la estructura de la corteza visual animal, y las redes neuronales convolucionales tienen neuronas dispuestas en tres dimensiones.

3.9.3. Redes Neuronales Convolucionales (CNN)

Una red convolucional (CNN) es una arquitectura de red de aprendizaje profundo que aprende directamente de los datos sin extraer características manualmente.” (Hernández, 2021)

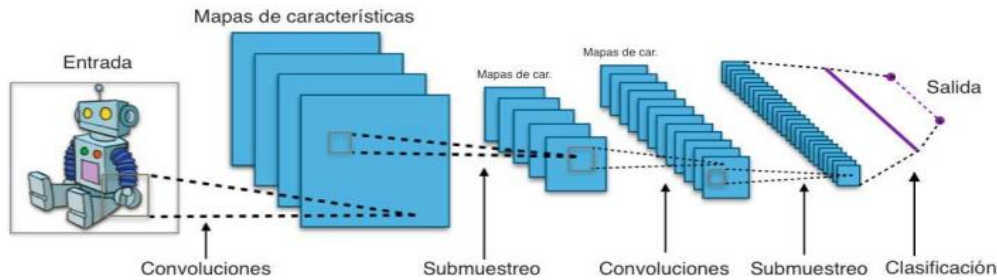
Una red neuronal convolucional (CNN) es una red neuronal de avance multicapa diseñada específicamente para el reconocimiento y clasificación de imágenes. (Martínez, 2018)

Según Durán (2017), Las redes convolucionales se construyen con la entrada a la red como imágenes que permiten codificar ciertas propiedades en la arquitectura, aumentando así la

eficiencia y reduciendo la cantidad de parámetros utilizados en la red. La Figura 4 proporciona una representación visual de la estructura básica de la arquitectura.

Figura 4

Estructura Básica de la Red Neuronal Convolutiva



Nota. Obtenido del trabajo de fin de grado: Redes neuronales convolucionales en R: Reconocimiento de caracteres escritos a mano (pg. 24). Fuente: (Durán, 2017)

Según Martínez (2018), La estructura de una red convolucional tiene principalmente tres capas: capa convolucional, capa convolucional y capa completamente conectada. Estas capas realizan una serie de transformaciones y submuestreos hasta que finalmente, mediante el uso de una capa combinada (perceptrón multicapa), se obtiene el resultado deseado, igual al número de categorías.

Las redes neuronales convolucionales (CNN) utilizan un sistema de etiquetado para clasificar datos visuales y comprender imágenes, lo que implica analizar la imagen a nivel de píxel y asignar un valor de etiqueta a cada píxel. Estos valores se utilizan para realizar una operación matemática llamada convolución que permite predecir la imagen. Una CNN comienza reconociendo contornos, colores y formas, luego repite el proceso de predicción en varias iteraciones para aumentar la precisión de los resultados. (Amazon, 2023)

Las redes neuronales convolucionales son un tipo de red neuronal profunda diseñada para procesar redes de datos estructurados, lo que las hace útiles para tareas de visión por computadora.

3.9.4. Componentes de la Arquitectura CNN

3.9.4.1. Capa Convolutacional.

Es la operación principal de la red neuronal convolutacional, debiendo utilizarse una o más capas de este tipo en la red en cuestión. Filtros entrenables en grupo (Bonilla, 2020).

3.9.4.2. Capa de Agrupación (pooling layer).

La capa combinada toma los mapas de características creados por la capa convolutacional y los clasifica en imágenes, reduce la calidad de la imagen, reduce la complejidad del modelo, evita la adición e intenta suavizar la información en la salida de la capa convolutacional. (Bonilla, 2020)

3.9.4.3. Capa de Unidades Lineales Rectificadas ReLu (Rectified Linear Units Layer).

Este tipo de características son muy comunes en las redes neuronales convolutacionales y se pueden usar con frecuencia en redes neuronales artificiales; después de cada capa convolutacional, el efecto es agregar una carga no uniforme a la función de activación sin cambiar el cuerpo de ganancia de la capa convolutacional. (Bonilla, 2020)

3.9.4.4. Capa Completamente Conectada (Full connected layer).

Esta capa es una arquitectura totalmente integrada de todas las neuronas de la capa anterior y mapas de características de la capa anterior. Esto depende de la capacidad total de entrada, por lo que hay muchas conexiones. En esta capa se debe establecer el número de neuronas activas. Su función es formar una colección de información obtenida para calcular la distribución final. (Bonilla, 2020)

3.9.4.5. Capa Pérdida (Loss layer).

En esta capa, las predicciones se comparan con los valores reales de la imagen. Se utiliza un clasificador de pérdidas Softmax, una función euclidiana que se utiliza para eliminar el ruido en las etiquetas de las imágenes.

3.9.5. Arquitecturas de Redes Neuronales Convolutacionales

3.9.5.1. VGG16 (Visual Geometry Group)

Es una red neuronal profunda que utiliza capas convolucionales de 3×3 y convolucionales de 2×2 para extraer características de las imágenes. Además, contiene 13 capas convolucionales, 5 capas convolucionales y 3 capas convolucionales totalmente integradas, seguidas de una capa de salida. Aunque tiene más parámetros que otras arquitecturas existentes, su simplicidad y unidad muestran un buen desempeño en muchas tareas de visión por computadora. (Enciso et al., 2023)

La red neuronal VGG16 se caracteriza por una estructura simple basada en la repetición de muchos bloques. El número "16" se refiere a todos los procesos incluyendo sus pesos, 13 al proceso convolucional y 3 a todas las operaciones que permiten aprender la capa gruesa o todas las capas. (Miranda, 2019)

3.9.5.2. MobileNetV2

La arquitectura MobileNet está diseñada para funcionar sin la necesidad de grandes recursos informáticos, como dispositivos móviles. Se admite el color RGB; además, esta red neuronal reduce significativamente el procesamiento y la memoria sin comprometer la precisión. Otro desarrollo de esta arquitectura es la comunicación directa, dirigida a controlar el flujo de gradientes en la red. Al agregar este contenido, la red puede encontrar activaciones no modificadas en el bloque. (Navarro & Martínez, 2022)

Es una arquitectura de red neuronal convolucional basada en una red residual convolucional, donde existen conexiones residuales entre capas de cuello de botella. Para evitar errores en la estructura, la arquitectura es una primera capa con 32 filtros, seguida de una segunda capa con las 19 capas restantes (Del Carpio & Núñez, 2022).

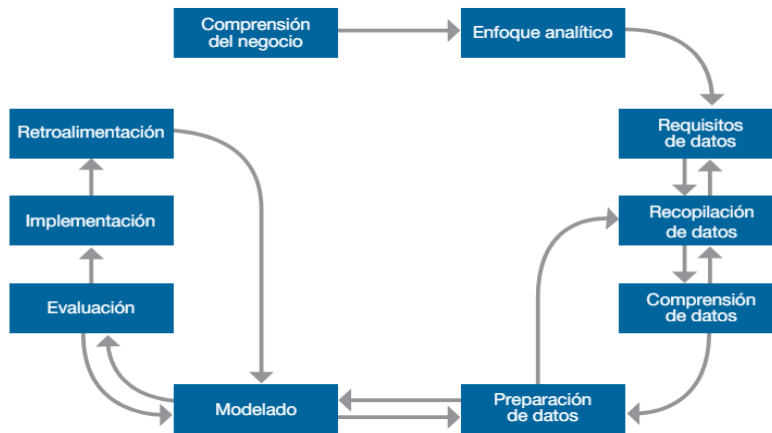
3.10. Metodología Fundamental para Ciencia de Datos de IBM

Una metodología es una práctica general que guía los procesos y actividades en un campo, independientemente de tecnología o equipo específico, y no se limita a técnicas o métodos de

construcción. Más bien, proporciona un marco para pensar y una forma práctica de resolver problemas y alcanzar metas en una situación determinada.

Figura 5

Metodología Fundamental para la Ciencia de Datos



Nota: Etapas de la metodología fundamental para ciencia de datos de IMB. Fuente: (Rollins, 2015)

3.10.1. Etapa 1: Comprensión del Negocio

En la primera fase se han establecido todas las cuestiones y objetivos necesarios para la realización del proyecto, sentando las bases para una adecuada resolución de problemas, por lo que es muy importante que los patrocinadores trabajen juntos durante todo el proyecto. Garantizar que las actividades se lleven a cabo de manera eficaz y se alcancen los resultados esperados (Rollins, 2015).

3.10.2. Etapa 2: Enfoque Analítico

En este proceso, el problema se transforma en algo que las estadísticas y el aprendizaje automático no comprenden, y la organización puede encontrar la mejor manera de lograr el resultado deseado.

3.10.3. Etapa 3: Requisitos de Datos

Método analítico para determinar la información requerida, los métodos analíticos utilizados requieren contenido, estructura y presentación de datos específicos, tomados del conocimiento de la industria.

3.10.4. Etapa 4: Recopilación de Datos

En la primera etapa de recopilación de datos, se identifican y recopilan datos relacionados con el área problemática de las fuentes de datos disponibles (estructurados, no estructurados y semiestructurados). Al incorporar más datos, los modelos predictivos pueden representar mejor eventos raros.

3.10.5. Etapa 5: Comprensión de Datos

Después de la recopilación de datos primarios, se utilizaron estadísticas descriptivas y visualización para comprender el contenido de los datos, evaluar la calidad y obtener una comprensión inicial.

3.10.6. Etapa 6: Preparación de Datos

Esta sección incluye todas las actividades relacionadas con la creación de estructuras de datos utilizadas en su posterior procesamiento. Las actividades de preparación de datos incluyen: limpieza de datos, combinación de datos de múltiples fuentes y transformación de datos.

3.10.7. Etapa 7: Modelado

En el proceso de modelado, se realiza una reconstrucción inicial de un conjunto de datos preparado para crear modelos predictivos, explicaciones o clasificaciones basadas en métodos de investigación aplicados. A medida que este enfoque cambia, también cambia el procesamiento de datos y la especificación del modelo, analizando diferentes algoritmos y sus diferentes enfoques para determinar el mejor modelo en función de las variables disponibles.

3.10.8. Etapa 8: Evaluación

Antes de implementar este modelo, se realiza una evaluación exhaustiva para comprender su calidad y garantizar que el modelo realmente resuelva los problemas existentes. Esta evaluación

incluye el cálculo de diversas pruebas de diagnóstico y la creación de tablas y gráficos adicionales para interpretar la efectividad y calidad del modelo en la resolución del problema.

3.10.9. Etapa 9: Implementación

Una vez que el prototipo se desarrolla y prueba con éxito, se implementa en un entorno de producción o en un entorno de prueba similar. Inicialmente, la configuración será limitada hasta que se pueda evaluar el rendimiento. La incorporación de modelos a los procesos operativos requiere el uso de métodos adicionales para garantizar un funcionamiento sin problemas.

3.10.10. Etapa 10: Retroalimentación

Al recopilar los resultados del modelo implementado, se obtiene retroalimentación sobre el rendimiento del modelo y su impacto en el entorno en el que se implementó.

3.11. Metodología de Desarrollo Mobile-D

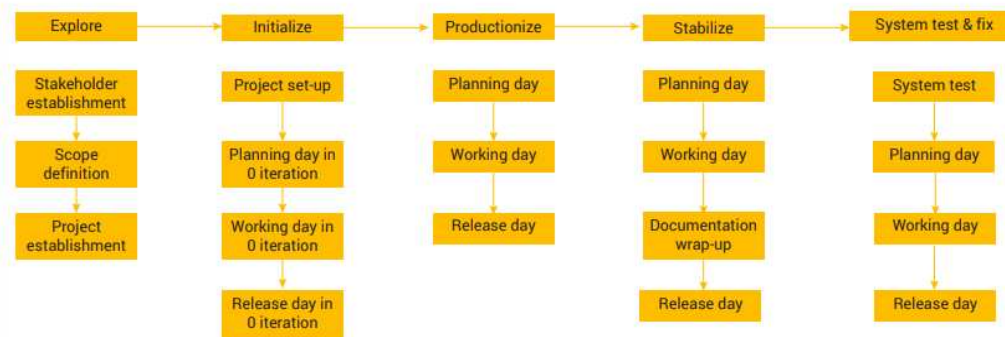
Es una metodología enfocada en pequeñas empresas de desarrollo, que, debido a los cortos tiempos de desarrollo, minimiza los costos de producción, esto la hace asequible para pequeñas organizaciones con personal y recursos limitados (Rodríguez & Socorro, 2018).

Es una manera fácil de crear una aplicación móvil, que permite una comunicación continua entre el equipo de trabajadores y el cliente, como respuesta rápida a los cambios que puedan ocurrir durante la creación del proyecto, reduciendo así el tiempo de desarrollo del producto (Molina et al., 2021).

La forma clásica de crear aplicaciones móviles, desarrollada en colaboración con tres de las empresas de software más populares. Mobile-D ha sido evaluado con éxito para la certificación CMMI (Capability Model Integration) Nivel 2 (Baldoce, 2017) y en la Figura 6 podemos ver el cambio en la longevidad de este enfoque.

Figura 6

Ciclo de Vida de la Metodología Mobile-D



Nota: Replicado de la revista de tecnología, metodologías ágiles en el desarrollo de aplicaciones para dispositivos móviles (pg. 23). Fuente: (Amaya, 2013).

3.11.1. Fase 1: Exploración

El objetivo de esta fase es planificar y crear un proyecto, sentar las bases para su desarrollo, definir las principales cuestiones, como identificar los objetivos, el tipo de aplicación y el desarrollo de la plataforma que los interesados han decidido, crear un meta. Incrementa y organiza el trabajo.

3.11.2. Fase 2: Iniciación

Durante el proceso de desarrollo del proyecto, se hacen planes para evitar problemas en futuras aplicaciones, se preparan todos los recursos materiales y técnicos para las actividades de producción y se determinan las herramientas de desarrollo a utilizar.

3.11.3. Fase 3: Producción

El propósito de esta fase es establecer los requisitos del producto utilizando métodos iterativos e incrementales. En esta fase se elabora un plan de trabajo que orienta el desarrollo.

3.11.4. Fase 4: Estabilización

Esta fase ocurre después de que la fase anterior completa el proceso de iteración. Implica la integración de funciones desarrolladas en diferentes módulos o subsistemas. Si se encuentra un

error, se deben realizar las correcciones necesarias para garantizar la finalización adecuada del producto.

3.11.5. Fase 5: Pruebas

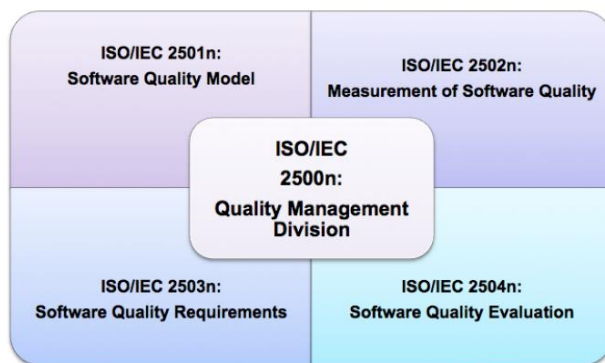
En esta etapa se completa el desarrollo, se realiza una fase de pruebas y se realizan iteraciones hasta obtener una versión estable que cumple con los requisitos establecidos.

3.12. ISO/IEC 25000

La familia ISO/IEC 25000 se basa en una revisión de normas anteriores, incluida ISO/IEC 9126, que especifica estándares de calidad para productos informáticos, e ISO/IEC 14598, que se ocupa del proceso de evaluación de esos productos. . Hay cinco divisiones en esta familia de normas ISO/IEC 25000, como se muestra.

Figura 7

Divisiones de Calidad



Fuente: (ISO 25000, 2019)

La norma ISO/IEC 25000 especifica requisitos para productos, procesos y sus requisitos de inspección e incluye modelos de calidad que incorporan estándares de calidad del cliente y se definen durante el desarrollo. Es un conjunto de estándares diseñados para crear una forma común de medir la calidad de los productos de software (Ordóñez, 2018).

3.12.1. ISO/IEC 25010:2011

System and software engineering- Systems and software Quality Requirements and Evaluation (SQuaRE)-System and software quality models

El modelo de calidad es la piedra angular en la creación de un sistema de evaluación de la calidad de los productos software. Este modelo identifica las características de calidad que deben considerarse al evaluar las características de un producto de software determinado. La calidad de un producto de software se puede interpretar como el grado en que el producto satisface las necesidades de los usuarios y proporciona valor. Estos requisitos (funcionalidad, rendimiento, seguridad, mantenibilidad, etc.) se reflejan en el modelo de calidad, que divide la calidad del producto en funciones y subfunciones.

Figura 8

Características y Subcaracterísticas de Calidad Según SQuaRE



Fuente: (ISO 25000, 2019)

3.12.2. ISO/IEC 25022:2016

Systems and software engineering-System and software Quality Requeriments and Evaluation (SQuaRE)- Masurement of quality in use

ISO/IEC 25022:2016 define medidas de calidad para los atributos definidos en ISO/IEC 25010 y está destinado a usarse en conjunto. No asigna rangos de valores de medición a niveles nominales o grados de cumplimiento, ya que estos valores se determinan para cada sistema o producto en función del entorno de uso y las necesidades del usuario. Los usuarios principales de

ISO/IEC 25022:2016 son aquellos que realizan actividades de especificación y evaluación como parte de:

- **Desarrollo:** Incluye análisis de requisitos, diseño y pruebas mediante aceptación durante el proceso del ciclo de vida.
- **Gestión de la calidad:** Es una revisión sistemática de un producto o sistema. Por ejemplo, se realizan estimaciones al evaluar la calidad del uso como parte del aseguramiento y control de calidad.
- **Suministro:** Un acuerdo para proporcionar sistemas, productos de software o servicios de software en virtud de un contrato o acuerdo con un comprador, por ejemplo, verificando la calidad en pruebas de clasificación.
- **Adquisición:** Incluyendo la selección de productos y pruebas de aceptación al comprar o adquirir sistemas, productos de software o servicios de software de proveedores.
- **Mantenimiento:** Mejora del producto en función de las medidas de calidad en el uso.

3.12.3. ISO/IEC 25023:2016

Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Measurement of system and software product quality

Definir métricas de calidad para cuantificar la calidad de los productos de software y sistemas basándose en las características y subcaracterísticas definidas por ISO/IEC 25010 destinadas a usarse juntas.

ISO/IEC 25023:2016 no asigna rangos de valores de medición a niveles nominales o grados de conformidad, ya que estos valores se basan en factores como la naturaleza del sistema, el producto o parte del producto, y la software que define la categoría, el nivel de integridad y los requisitos del usuario. Algunas propiedades pueden tener un rango de valores ideal que no depende de las necesidades específicas del usuario, sino que depende de factores generales como los factores

cognitivos humanos. Los usuarios principales de ISO/IEC 25023:2016 son aquellos que realizan actividades de evaluación y especificación de requisitos de calidad.

3.12.4. ISO/IEC 25040:2011

Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Evaluation process

Contiene requisitos y recomendaciones para evaluar la calidad de los productos de software e ilustra conceptos generales. Proporciona una descripción del proceso de evaluación de la calidad del producto de software y define los requisitos de aplicación para el proceso. Este proceso se puede utilizar para una variedad de propósitos y métodos, como evaluar la calidad de software predesarrollado, software comercial o software personalizado, y se puede utilizar durante o después del proceso de desarrollo.

Está destinado a personas responsables de evaluar productos de software y se aplica a desarrolladores, adquirentes y evaluadores independientes de productos de software. No pretende evaluar otros aspectos del producto de software (como requisitos funcionales, requisitos de proceso, requisitos comerciales, etc.). Este estándar define el proceso de evaluación de productos de software. Este proceso de evaluación generalmente implica cinco pasos.

1. Establecer los requisitos de la evaluación
 - Establecer el propósito de la evaluación: Esta tarea documenta el propósito por el cual la organización desea evaluar la calidad de su producto de software (garantizar la calidad del producto, decidir si el producto es aceptable, determinar la viabilidad del proyecto que se está desarrollando, comparar la calidad del producto con otros Productos).
 - Obtener los requisitos de calidad del producto: En esta tarea se identifican las partes interesadas en el producto software (desarrolladores, potenciales beneficiarios,

usuarios, proveedores, etc.) y, mediante un modelo de calidad específico, se aclaran los requisitos de calidad del producto.

- **Identificar las partes del producto que se deben evaluar:** Las partes del producto software que se incluyen en la evaluación deben ser identificadas y registradas. La etapa del ciclo de vida en que se lleva a cabo la evaluación, así como el propósito de esta, determinan el tipo de producto a evaluar (diagramas de diseño, especificaciones de requisitos, documentación de las pruebas, etc.).
- **Definir el rigor de la evaluación:** Basándose en factores como el peligro para la seguridad, el peligro económico o el peligro ambiental, se debe establecer el rigor de la evaluación en función del propósito y del uso previsto del producto software. Qué métodos se emplean y los resultados esperados de la evaluación se determinarán en función del rigor.

2. Especificar la evaluación

- **Seleccionar los módulos de evaluación:** El evaluador elige las métricas de calidad, técnicas y herramientas (módulos de evaluación) para satisfacer todos los requisitos de la evaluación en esta tarea. Las comparaciones confiables con criterios que posibilitan la toma de decisiones deben ser posibles con respecto a su valor.
- **Definir los criterios de decisión para las métricas:** Se deben establecer los criterios de toma de decisiones para las métricas elegidas. Se pueden relacionar estos criterios con los requisitos de calidad y luego con los criterios de evaluación para determinar la calidad del producto. Benchmarks, límites de control estadísticos, datos históricos, requisitos del cliente, etc. pueden ser utilizados para establecer estos límites.
- **Definir los criterios de decisión de la evaluación:** Se deben establecer criterios para las distintas características que se han evaluado a partir de las métricas de calidad y las

subcaracterísticas. La evaluación general de la calidad del producto software es posible gracias a estos resultados más abstractos.

3. Diseñar la evaluación

- **Planificar las actividades de la evaluación:** Las actividades de evaluación deben planificarse teniendo en cuenta la disponibilidad de los recursos, tanto materiales como humanos, que puedan ser necesarios. El presupuesto, los métodos de evaluación, los estándares adaptados, las herramientas de evaluación y otros factores deben considerarse en la planificación. Durante el proceso de evaluación, el plan de evaluación se revisará y actualizará con información adicional según sea necesario.

4. Ejecutar la evaluación

- **Realizar las mediciones:** Para obtener los valores de las métricas elegidas y especificadas en el plan de evaluación, se deben realizar las mediciones sobre el producto software y sus componentes. Se registrarán correctamente todos los resultados obtenidos.
- **Aplicar los criterios de decisión para las métricas:** Se aplican los criterios de decisión para las métricas seleccionadas sobre los valores obtenidos en la medición del producto
- **Aplicar los criterios de decisión de la evaluación:** Los criterios de decisión a nivel de características y subcaracterísticas de calidad deben aplicarse en esta última tarea; esto da como resultado la evaluación de la medida en que el producto software cumple con los requisitos de calidad establecidos.

5. Concluir la evaluación

- **Revisar los resultados de la evaluación:** Para mejorar la interpretación de la evaluación y la detección de errores, el evaluador y el cliente de la evaluación (en caso de existir) realizan una revisión conjunta de los resultados obtenidos mediante esta tarea.

- **Crear el informe de evaluación:** El informe de evaluación incluye los requisitos de la evaluación, los resultados, las limitaciones y restricciones, el personal evaluador, etc. después de revisar los resultados.
- **Revisar la calidad de la evaluación y obtener feedback:** El evaluador examinará los hallazgos de la evaluación, así como la validez del procedimiento de evaluación, de las métricas y de los indicadores utilizados. El proceso de evaluación de la organización y los métodos utilizados deben mejorar con el feedback de la revisión.
- **Tratar los datos de la evaluación:** Una vez realizada la tasación, el tasador deberá procesar correctamente los datos y elementos de la tasación de acuerdo con el contrato celebrado por el cliente, especialmente si se trata de un tercero. Esto incluye restaurar, archivar o eliminar según sea necesario.

3.13. Herramientas

3.13.1. Google Colab.

Es una plataforma en línea proporcionada por Google que ofrece un entorno basado en la nube, esta plataforma permite a los usuarios escribir, ejecutar y compartir código Python en forma de cuadernos Jupyter.

3.13.2. Kaggle.

Es una plataforma en línea que reúne a una comunidad de científicos de datos y entusiastas del aprendizaje automático para colaborar en proyectos. Proporciona una plataforma para escribir código en forma de kernels basados en Jupyter, lo que permite la exploración de datos y la escritura de código en diversos lenguajes.

3.13.3. Python.

Es un lenguaje de programación de alto nivel, interpretado y generalizado, conocido por su simplicidad, flexibilidad y versatilidad. Fue creado por Guido Van Rossum y lanzado en 1991.

Se trata de un lenguaje de programación de código abierto que ofrece la capacidad de ejecutarse en múltiples plataformas. Sus usuarios lo elogian como el más elegante y amigable para la programación (Molina et al., 2016).

3.13.4. Librerías

- **split-folders:** Es una herramienta de Python que se utiliza para dividir un conjunto de datos en subconjunto ya sea para el entrenamiento, validación y prueba de manera sencilla y conveniente.
- **visualkerass:** Esta es una biblioteca de Python utilizada para visualizar modelos de redes neuronales construidas por Keras, es una biblioteca de aprendizaje profundo que permite la representación gráfica de la arquitectura de redes neuronales.
- **Skimage:** Es un acrónimo de la biblioteca scikit-image en Python, la cual es un conjunto de bibliotecas utilizada para el aprendizaje automático, se enfoca en el procesamiento de imágenes y proporciona una amplia gama de herramientas y algoritmos para trabajar con imágenes.
- **Numpy:** Es una biblioteca fundamental de Python para computación numérica, ya que proporciona un conjunto de herramientas y estructura de datos.
- **Os:** Es un módulo estándar de Python que proporciona una interfaz con el sistema operativo en el que se ejecuta el programa.
- **Matplotlib:** Es una biblioteca de Python que es usada para la creación de gráficos y visualizadores de datos de alta calidad.
- **Tensorflow:** Una biblioteca de código abierto desarrollada por Google para implementar y entrenar modelos de aprendizaje automático, especialmente modelos de redes neuronales artificiales.

- **Keras:** Es una biblioteca de alto nivel para la construcción y entrenamiento de redes neuronales artificiales en Python, su objetivo es proporcionar una interfaz para utilizar con modelos de aprendizaje profundo.
- **Sklearn:** Es una biblioteca Python de código abierto para aprendizaje automático y minería de datos, que se utiliza para crear, entrenar y evaluar modelos de aprendizaje automático.

3.13.5. Android Studio.

Un entorno de desarrollo integrado (IDE) diseñado para desarrollar aplicaciones de Android.

El sitio de desarrollo se lanzó en Google I/O en 2013 como una herramienta para arrancar. Este IDE es desarrollado por Google, la empresa propietaria de Android, por eso se le llama el IDE oficial para aplicaciones de Android (Luján, 2019).

3.13.6. Java.

Es un lenguaje sencillo orientado a objetos que permite el desarrollo de aplicaciones en diversos campos como seguridad, animación, aplicaciones cliente-servidor, interfaces gráficas, desarrollo de aplicaciones móviles, etc. (UNAM, 2023).

3.13.7. Firebase.

Se trata de datos en tiempo real relacionados con el desarrollo de aplicaciones móviles y la gestión de datos a través de ellas. Cuando la aplicación se conecta a Firebase, lo hace a través de websockets, por lo que es más rápida que HTTP porque solo necesita una conexión de socket. Todos los datos se sincronizan automáticamente a través de ese único websocket.

3.13.8. Dataset

Es un conjunto de datos o una colección estructurada de datos que se organiza y representa en un formato específico para facilitar el acceso, análisis y la interpretación.

3.13.8.1. Datos de entrenamiento.

El aprendizaje automático es un conjunto de datos que se utiliza para entrenar un algoritmo o modelo estadístico. Se utilizan para enseñar a los modelos cómo realizar tareas específicas, como clasificación, reconocimiento o predicción.

3.13.8.2. Datos de validación

Se trata de conjuntos de datos específicos que se utilizan para evaluar el rendimiento de modelos estadísticos o de aprendizaje automático, entrenados con variables específicas para mejorar el ajuste del modelo.

3.14. Métricas de Evaluación

3.14.1. Accuracy.

Métrica utilizada para evaluar el rendimiento de los modelos de clasificación en aprendizaje automático y aprendizaje matemático.

La precisión es la proporción de predicciones correctas en comparación con todas las predicciones del modelo especificado.

$$\text{Exactitud} = \frac{\text{NPC}}{\text{NTP}}$$

NPC: número de predicciones correctas

NTP: número total de predicciones

3.14.2. Precisión.

Métrica empleada en la evaluación de modelos de clasificación, representa la proporción de instancias positivas que son correctamente identificadas por el modelo, respecto al total de instancias clasificadas como positivas por el modelo.

(Nills, 2020). La precisión es una medida de la exactitud de una predicción positiva. En otras palabras, significa que, si un resultado se predice como positivo, ¿qué tan seguro puede estar de que sea realmente positivo? Se calcula utilizando la siguiente fórmula.

$$\text{Precisión} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

TP: verdaderos positivos

FP: falsos positivos

3.14.3. Recall.

Sensibilidad o Tasa de Verdaderos Positivos; esta métrica es utilizada en la evaluación de modelos de clasificación y representa la proporción de instancias positivas que fueron correctamente identificadas por el modelo, en relación con el total real de instancias positivas en el conjunto de datos.

(Nills, 2020). Recall, o la tasa de verdaderos positivos, es la medida de cuántos verdaderos positivos se predicen de todos los positivos en el conjunto de datos; se define con la siguiente fórmula.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

TP: verdaderos positivos

FN: falsos negativos

3.14.4. F1 score.

Es una métrica que combina la precisión y el recall en un solo número, proporcionando una medida más equilibrada del rendimiento de un clasificador binario.

$$\text{F1 - score} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

$$\text{F1 - score} = \frac{2 \times \text{TP}}{2 \times \text{TP} + \text{FP} + \text{FN}}$$

TP: verdaderos positivos

FP: falsos positivos

FN: falsos negativos

3.14.5. Matriz de Confusión.

Esta matriz muestra el número de predicciones correctas en comparación con el número de predicciones incorrectas, lo que significa el número de rechazos positivos y resultados positivos (predicciones correctas) y el número de positivos negativos y falsos (predicción incorrecta) (Nills, 2020).

Según Huerta et al. (2020) La matriz de confusión es una de las métricas más simples y fáciles para estimar la precisión, ya que es una tabla que describe el desempeño de la estructura.

Tabla 1

Matriz de Confusion

| | |
|----|----|
| VP | FN |
| FP | VN |

Fuente: Romero (2023)

- **VP:** Representa la cantidad de casos en los que el modelo predijo correctamente la clase positiva.
- **Verdaderos Negativos (VN):** Representa la cantidad de casos en los que el modelo predijo correctamente la clase negativa.
- **Falsos positivos (FP):** Representa la cantidad de casos en los que el modelo predijo incorrectamente la clase positiva cuando en realidad era negativa.
- **Falsos negativos (FN):** Representa la cantidad de casos en los que el modelo predijo incorrectamente la clase negativa cuando en realidad era positiva.

Capítulo 4

Metodología de la Investigación

4.1. Tipo de Investigación

La presente investigación es de tipo **Aplicada**, según (Lozada, 2014) La investigación aplicada tiene como objetivo crear conocimiento y contribuir directamente a los problemas de la sociedad o la industria. En el presente estudio se desarrolla la aplicación móvil “SickCorn” en la detección de *Puccinia sorghi* (roya común) y *Spiroplasma kunkeli* (puka poncho) en las plantas de maíz.

La investigación se centra en mejorar un problema en una situación. Su propósito es aplicar conocimientos de una o más áreas específicamente para implementar soluciones prácticas que satisfagan necesidades específicas, ya sea en el sector público o con fines de lucro. (DuocUC, 2024)

4.2. Enfoque de Investigación

En el presente estudio, el enfoque de investigación es **cuantitativo** y según Hernández et al. (2014), El método de medición es secuencial y confirmatorio e implica medir parámetros en un contexto específico. Luego, estas mediciones se analizan utilizando métodos estadísticos y se extraen conclusiones. Este método utiliza la recopilación de datos para probar hipótesis basadas en mediciones numéricas para establecer patrones de comportamiento y confirmar teorías.

4.3. Diseño de Investigación

El diseño de investigación es **No Experimental**, Este tipo de investigación se realiza sin controlar variables ya que se centra en los acontecimientos que ocurren en su entorno para su análisis. (Escamilla, 2008)

Según Hernández et al. (2014), la investigación no experimental Es un enfoque sistemático y empírico que no utiliza variables independientes como en el pasado. Las suposiciones sobre las

relaciones entre variables se hacen sin intervención directa o influencia intencional, ya que las relaciones se realizan en su contexto natural.

El tipo de diseño de investigación es el diseño de panel longitudinal, ya que se pueden utilizar datos móviles durante el desarrollo del maíz, y este diseño se utiliza para recopilar datos en varios momentos para examinar el desarrollo que ocurre durante el desarrollo del maíz y evaluar los cambios.

Según Hernández et al. (2014), los diseños longitudinales, recolectan datos en diferentes momentos o periodos para hacer inferencias respecto al cambio, sus determinantes y consecuencias.

4.4. Nivel de Investigación

El nivel de investigación del presente trabajo es **descriptivo**, ya que se pretende medir la precisión de la detección con la aplicación móvil extrayendo las características de las enfermedades en la hoja de la planta de maíz. Según Hernández et al. (2014), La investigación descriptiva busca identificar rasgos y características importantes de un objeto en particular y busca medir o recopilar información sobre variables de forma individual o colectiva.

4.5. Población

La población utilizada en el presente trabajo es una tarea de 3500 plantas de maíz del distrito de Talavera.

Según Hernández et al. (2014), Una población o el mundo en su conjunto coincide en determinadas características en todos los casos.

4.6. Muestra

El muestreo es de tipo no probabilístico por conveniencia, ya que solo deben seleccionarse plantas de maíz que estén enfermas con *Puccinia sorghi* (roya común) y *Spiroplasma kunkeli* (puka poncho) y plantas sanas en el distrito de Talavera.

Según Otzen & Manterola (2017), El método de muestreo no probabilístico se basa en la selección de los sujetos de investigación en base a ciertas características y criterios que el investigador estaba considerando en su momento.

Según Otzen & Manterola (2017), El muestreo por conveniencia permite la selección de casos disponibles para su inclusión en función de la disponibilidad del sujeto y la proximidad al investigador.

Tamaño de la muestra

La población del presente estudio es finita, la cual es una tarea con 3500 plantas de maíz, por lo tanto, se realizará los siguientes cálculos para determinar el tamaño de muestra a utilizar:

$$n = \frac{N \times Z^2 \times \alpha^2}{(N - 1) \times E^2 + Z^2 \times \alpha^2}$$

- n: tamaño de la muestra
- N: tamaño de la población
- Z: el valor estadístico correspondiente al nivel de confianza
- α : desviación estándar
- E: margen de error deseado

Por lo tanto, realizamos los cálculos:

$$N = 3500 \quad z = 1.96 \quad \alpha = 0.5 \quad E = 0.05$$

$$n = \frac{3500 \times 1.96^2 \times 0.5^2}{(3500 - 1) \times 0.05^2 + 1.96^2 \times 0.5^2}$$

$$n = 346.2$$

Para el presente proyecto de investigación el tamaño de muestra es de 346 imágenes de plantas de maíz y el muestreo será no probabilística por conveniencia.

4.7. Técnicas e Instrumentos de Recopilación de Datos

Los datos se obtuvieron mediante **la técnica de la observación directa**, digitalizando las imágenes de las enfermedades: *Puccinia sorghi* (roya común) y *Spiroplasma kunkeli* (puka poncho), utilizando una cámara de un dispositivo móvil, con la cual se tomó fotos de diferentes ángulos, las cuales están en formato JPG.

Según (Zapata, 2006), El método de análisis es el método mediante el cual el investigador presenta con precisión los datos que se estudian sin modificaciones.

La observación directa es un método mediante el cual los investigadores observan y recopilan datos a través de sus propias observaciones (Tamayo, 2007, pág. 193).

Según Ñaupas et al. (2013), La observación directa y no participante se refiere al contacto directo entre el investigador y los sujetos. En este tipo de observación, el investigador utiliza telescopios, videgrabadores, cámaras fotográficas, cuadernos, fichas de campo, etc. para potenciar o complementar la observación. Involucrado en la observación de la situación utilizando herramientas de observación. Obtener conocimiento científico de ello.

4.8. Confiabilidad de los Datos

La confiabilidad de los datos será determinada por expertos en la materia, en este caso agrónomos especializados en sanidad vegetal. Estos especialistas se especializan en agronomía, la rama de la ciencia agrícola dedicada al estudio, práctica y producción de cultivos. Los agrónomos desempeñan un papel importante en la salud de los cultivos, lo que ayuda a la detección, el diagnóstico y el manejo tempranos de enfermedades.

4.9. Técnicas y Análisis de Datos

El análisis de datos implica analizar un conjunto de datos para sacar conclusiones válidas. Los datos se procesaron para poder alimentar la red neuronal combinada, por lo tanto, los datos se analizaron en formato de histograma, ya que nos proporciona información sobre la distribución de intensidad de píxeles de la imagen, por lo que se encontró que el modelo era adecuado.

4.10. Operacionalización de Variables

La investigación es de nivel aplicativo con un enfoque cuantitativo y una intervención del tipo de diseño no experimental, la operacionalización de variables se define de acuerdo con el estudio planteado, a continuación, se muestra la tabla de operacionalización de variables.

Tabla 2

Operacionalización de Variables

| VARIABLES | DEFINICIÓN CONCEPTUAL | DEFINICIÓN OPERACIONAL | DIMENSIONES | INDICADORES | MEDIDA |
|--|---|--|-------------------------|---|--------------|
| APLICACIÓN MOVIL SICKCORN CON MODELO DE DETECCION DE IMÁGENES DE LA HOJA DEL MAÍZ | Una aplicación móvil con un modelo de detección de imágenes es una herramienta que utiliza la tecnología de visión artificial y aprendizaje automático para identificar y clasificar las características a partir de imágenes capturadas con la cámara del dispositivo móvil (Casa & Tumbaco, 2021) | La aplicación móvil para dispositivos Android captura una imagen por medio de la cámara del dispositivo móvil, esta imagen se procesa para mejorar la calidad y prepararla para la clasificación de puccinia sorgi, spiroplasma kunkeli y planta sana por el modelo de clasificación con las redes neuronales convolucionales que retorna un resultado que es mostrado por la aplicación móvil | Modelo VGG16 | Taza de aprendizaje | $0 > TA < 1$ |
| | | | Épocas de entrenamiento | $1 > E < 1000$ | |
| | | | Cantidad de neuronas | Conteo por capa $Neuronas_{Total} =$ $Neuronas_{conv}$ $+ Neuronas_{pool}$ $+ Neuronas_{fully}$ | |
| | | | Modelo MobilenetV2 | Taza de aprendizaje | $0 > TA < 1$ |
| | | | Épocas de entrenamiento | $1 > E < 1000$ | |
| | | | Cantidad de neuronas | Conteo por capa $Neuronas_{Total} =$ $Neuronas_{conv}$ $+ Neuronas_{pool}$ $+ Neuronas_{fully}$ | |
| | | | Modelo implementado CNN | Taza de aprendizaje | $0 > TA < 1$ |
| | | | Épocas de entrenamiento | $1 > E < 1000$ | |
| | | | Cantidad de neuronas | Conteo por capa $Neuronas_{Total} =$ $Neuronas_{conv}$ $+ Neuronas_{pool}$ $+ Neuronas_{fully}$ | |

| | | | | | |
|--|--|---|-----------------------------|--|--|
| DETECCIÓN DE PUCINIA SORGHY (ROYA COMÚN) Y SPIROPLASMA KUNKELI (PUKA PONCHO) EN LAS PLANTAS DE MAIZ | La detección de enfermedades es el proceso de identificar y diagnosticar enfermedades en plantas y animales (CGIAR, 2018). | Se determina las enfermedades en los cultivos de maíz que presentan algún tipo de síntoma basado en la Puccinia sorghi y Spiroplasma kunkeli. | Rendimiento de la detección | El nivel de exactitud en la detección de Puccinia sorghi y spiroplasma kunkeli en las plantas de maíz. | $E = \frac{NPC}{NTP}$ |
| | | | | El nivel de precisión en la detección de Puccinia sorghi y spiroplasma kunkeli en las plantas de maíz. | $P = \frac{TP}{TP + FP}$ |
| | | | | El nivel de sensibilidad en la detección de Puccinia Sorghi y Spiroplasma Kunkeli. | $R = \frac{TP}{TP + FN}$ |
| | | | | El nivel de precisión y sensibilidad. | $F1 = \frac{2 \times TP}{2 \times TP + FP + FN}$ |

Calidad de la aplicación móvil ISO/IEC 25000 (SickCorn)

Característica seguridad

Característica eficiencia del desempeño

Característica usabilidad

Capítulo 5

Resultados

5.1. Evaluación de los Modelos

En este capítulo, los resultados del análisis de muestras han confirmado la detección de *Puccinia sorghi* (roya común), *Spiroplasma kunkeli* (puka poncho) y hojas sanas. Estos resultados están respaldados por el análisis de diferentes mediciones, lo que nos permite evaluar el desempeño de diferentes tipos de indicadores. Las métricas utilizadas en la evaluación son la matriz de confusión, la exactitud, la precisión, la sensibilidad y el F1-Score, que se utiliza para calcular el rendimiento del modelo.

5.1.1. Evaluación del Modelo Implementado

Para realizar la evaluación, se empleó una muestra de 346 imágenes de las clases *Puccinia sorghi* (roya común), *Spiroplasma kunkeli* (puka poncho) y hoja sana. Se puede entender mejor cómo funciona el modelo para detectar enfermedades en plantas de maíz a través de la evaluación de estas muestras.

Tabla 38

Resultado del Modelo Implementado

| Modelo | Clases | Verdadero | Verdadero | Falso | Falso |
|--------------|-------------|-----------|-----------|----------|----------|
| | | Positivo | Negativo | Positivo | Negativo |
| Implementado | Puccinia | 102 | 5 | 6 | 4 |
| | sorgui | | | | |
| | Spiroplasma | 116 | 11 | 8 | 9 |
| | kunkeli | | | | |
| | Sana | 73 | 3 | 4 | 5 |

- **Verdadero Positivo (VP):** Evento favorable, en el que la aplicación detecta las enfermedades en las plantas de maíz.
- **Verdadero negativo (VN):** Evento en que la aplicación detecta la planta sana como planta enferma.
- **Falso Positivo (FP):** Son casos en que la aplicación detecta las enfermedades en las plantas de maíz como planta sana.
- **Falso Negativo (FN):** Corresponde a los casos en que la aplicación no detecta la planta enferma ni la planta sana.

5.1.1.1. Matriz de Confusión del Modelo Implementado

En la matriz de confusión de tres por tres, que corresponde a las dos clases de enfermedades, se muestra que: Spiroplasma kunkeli (puka poncho), Puccinia sorghi (roya común) y la clase sana.

La matriz de confusión se estructura de la siguiente manera: en el eje horizontal se presentan las etiquetas de predicción, mientras que en el eje vertical se encuentran las etiquetas verdaderas de las clases. En esta matriz, los valores en el eje diagonal representan las predicciones correctas que fueron clasificadas de manera adecuada, lo que indica que el modelo clasificó correctamente las imágenes correspondientes a las clases de enfermedades.

Figura 9

Matriz de Confusión del Modelo Implementado

| Clases | Predicciones | | |
|---------------------|-----------------|------|---------------------|
| | Puccinia Sorghi | Sana | Spiroplasma Kunkeli |
| Puccinia Sorghi | 110 | 1 | 5 |
| Sana | 0 | 85 | 0 |
| Spiroplasma Kunkeli | 6 | 4 | 134 |

Nota. Adaptado de la revista de ciencia y tecnología de la universidad de Palermo de (Navarro & Martínez, 2022).

En la Figura 9. se puede observar que se tuvo un acierto de 110 imágenes clasificadas correctamente como Puccinia sorghi, además de que la hoja sana fue clasificada correctamente y la clase Spiroplasma kunkeli tuvo un acierto de 134 imágenes.

5.1.1.2. Precisión del Modelo Implementado

En esta sección, se determinó el nivel de precisión del modelo de clasificación para las enfermedades, Puccinia sorghi (roya común), Spiroplasma kunkeli (puka poncho) y hoja sana en las plantas de maíz.

$$P0 = \frac{TP}{TP + FP} = \frac{102}{102 + 6} = 0.944$$

$$P1 = \frac{TP}{TP + FP} = \frac{116}{116 + 8} = 0.935$$

$$P2 = \frac{TP}{TP + FP} = \frac{73}{73 + 4} = 0.948$$

En la evaluación se obtuvo un nivel de precisión del 94.4% para la clase puccinia sorghi, 93.5% en spioplasma y 94.8% para la clase sana con el modelo implementado.

5.1.1.3. Exactitud (Accuracy) del Modelo Implementado

La exactitud es una métrica esencial que nos permite evaluar la proporción de clasificaciones correctas realizadas por el modelo en comparación con el total de casos evaluados. Por consiguiente, examinamos el nivel de exactitud del modelo de clasificación para las patologías Puccinia sorghi (roya común), Spiroplasma Kunkeli (puka poncho) y la planta sana.

$$E0 = \frac{325}{346} = 0.939$$

En la evaluación se obtuvo una exactitud del 93.9%, lo que significa que el modelo clasifica correctamente las enfermedades evaluadas con un pequeño margen de error.

5.1.1.4. Sensibilidad (Recall) del Modelo Implementado

Se ha determinado el nivel de sensibilidad del modelo de clasificación para las patologías *Puccinia sorghi* (roya común), *Spiroplasma Kunkeli* (puka poncho) y planta sana.

$$R0 = \frac{TP}{TP + FN} = \frac{102}{102 + 4} = 0.962$$

$$R1 = \frac{TP}{TP + FN} = \frac{116}{116 + 9} = 0.928$$

$$R2 = \frac{TP}{TP + FN} = \frac{73}{73 + 5} = 0.935$$

Se obtuvo como resultado un nivel de sensibilidad del 96.2% para la clase *puccinia sorghi*, 92.8% para la clase *spiroplasma kunkeli* y 93.5% para sana, por lo que el modelo tiene la capacidad de identificar las instancias positivas, es decir, puede identificar las plantas enfermas y la planta sana.

5.1.1.5. F1-score del Modelo Implementado

Se evalúa el equilibrio entre la precisión y la sensibilidad, para determinar el punto medio entre ambas evaluaciones.

$$F0 = \frac{2 \times TP}{2 \times TP + FP + FN} = \frac{2 \times 102}{2 \times 102 + 6 + 4} = 0.953$$

$$F1 = \frac{2 \times TP}{2 \times TP + FP + FN} = \frac{2 \times 116}{2 \times 116 + 8 + 9} = 0.931$$

$$F2 = \frac{2 \times TP}{2 \times TP + FP + FN} = \frac{2 \times 73}{2 \times 73 + 4 + 5} = 0.941$$

El resultado obtenido del modelo alcanza un valor del 95.3% para la clase *puccinia sorghi*, 93.1% para la clase *Spiroplasma kunkeli* y 94.1% para la clase sana por lo que se tiene un equilibrio entre la precisión y sensibilidad en la clasificación.

5.1.2. Evaluación del Modelo VGG16

Se realiza la evaluación del modelo VGG16 entrenado con los datos de la muestra, para verificar la correcta funcionalidad en la detección de *Puccinia sorghi* (roya común), *Spiroplasma kunkeli* (puka poncho) y hoja sana en la planta de maíz, para la cual se realiza la evaluación con 346 datos correspondientes a la muestra.

Tabla 39

Resultado del Modelo VGG16

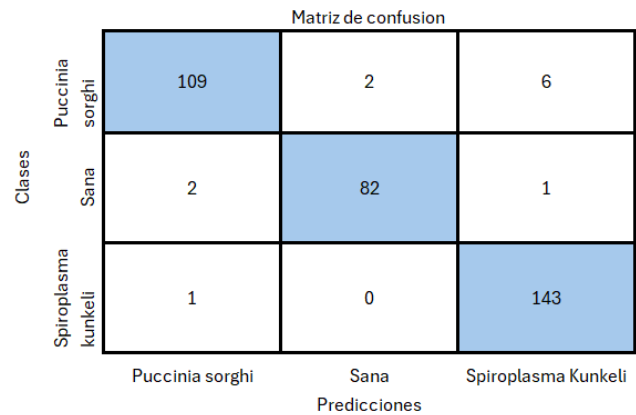
| Modelo | Clases | Verdadero | Verdadero | Falso | Falso |
|--------|---------------------|-----------|-----------|----------|----------|
| | | Positivo | Negativo | Positivo | Negativo |
| VGG16 | Puccinia sorghi | 104 | 6 | 4 | 3 |
| | Spiroplasma kunkeli | 128 | 5 | 6 | 5 |
| | Sana | 73 | 4 | 5 | 3 |

5.1.2.1. Matriz de Confusión del Modelo VGG16

Se realiza la evaluación de la matriz de confusión del modelo VGG16 con los 346 datos perteneciente a la muestra.

Figura 10

Matriz de Confusión del Modelo VGG16



Nota. Adaptado de la revista de ciencia y tecnología de la universidad de Palermo de (Navarro & Martínez, 2022).

Se puede observar que la clasificación de la clase *Puccinia sorghi* tuvo un acierto de 109 imágenes, además, la clase sana tuvo un acierto de 82 imágenes y la clase *spiroplasma kunkeli* tuvo un acierto de 143 imágenes.

5.1.2.2. Precisión del Modelo VGG16

La precisión es la identificación de las enfermedades identificadas correctamente, para que los resultados obtenidos refuercen la confiabilidad del modelo VGG16.

$$P0 = \frac{TP}{TP + FP} = \frac{104}{104 + 4} = 0.962$$

$$P1 = \frac{TP}{TP + FP} = \frac{128}{128 + 6} = 0.955$$

$$P2 = \frac{TP}{TP + FP} = \frac{73}{73 + 5} = 0.935$$

La métrica obtuvo un valor del 96.2% para la clase *puccinia sorghi*, 95.5% para *spiroplasma kunkeli* y 93.5% para la clase sana, por lo que el modelo identifica las enfermedades y la hoja sana correctamente.

5.1.2.3. Exactitud (Accuracy) del Modelo VGG16

En esta evaluación, se mide la proporción de clasificaciones correctas entre todas las predicciones realizadas

$$E = \frac{335}{346} = 0.962$$

En los cálculos de esta métrica se obtuvo un valor del 96.2% de clasificaciones correctas entre las clases *Puccinia sorghi* (roya común), *Spiroplasma kunkeli* (puka poncho) y hoja sana de la planta de maíz.

5.1.2.4. Sensibilidad (Recall) del Modelo VGG16

La sensibilidad, también conocida como recall, es una métrica importante que nos permitió evaluar la capacidad del modelo para identificar correctamente los casos positivos de cada clase.

$$R0 = \frac{TP}{TP + FN} = \frac{104}{104 + 3} = 0.971$$

$$R1 = \frac{TP}{TP + FN} = \frac{128}{128 + 5} = 0.962$$

$$R2 = \frac{TP}{TP + FN} = \frac{73}{73 + 3} = 0.960$$

Se obtuvo un valor del 97.1% para puccinia sorghi, 96.2% para spiroplasma kunkeli y 96% para la clase sana, por lo que el modelo identifica correctamente los casos positivos de las clases Puccinia sorghi, Spiroplasma kunkeli y hoja sana.

5.1.2.5. F1-score del modelo VGG16

La evaluación utilizando esta métrica es esencial porque nos permite calcular la proporción de clasificaciones correctas que el modelo ha realizado entre la sensibilidad y la precisión.

$$F0 = \frac{2 \times TP}{2 \times TP + FP + FN} = \frac{2 \times 104}{2 \times 104 + 4 + 3} = 0.967$$

$$F1 = \frac{2 \times TP}{2 \times TP + FP + FN} = \frac{2 \times 128}{2 \times 128 + 6 + 5} = 0.958$$

$$F2 = \frac{2 \times TP}{2 \times TP + FP + FN} = \frac{2 \times 73}{2 \times 73 + 5 + 3} = 0.948$$

En la evaluación de esta métrica se obtuvo un valor del 96.7% para puccinia sorghi, 95.8% para spiroplasma kunkeli y 94.8% para la clase sana, por lo que se tiene una buena asimetría entre la precisión y la sensibilidad.

5.1.3. Evaluación del Modelo MobileNetV2

La evaluación del modelo MobilenetV2 se realiza con la finalidad de medir el rendimiento del modelo entrenado en la clase Puccinia sorghi (roya común), Spiroplasma kunkeli (puka

poncho) y hoja sana en las plantas de maíz, utilizando los 346 datos de la muestra. La evaluación se realiza mediante la precisión, sensibilidad, exactitud y la matriz de confusión.

Tabla 40

Resultados del Modelo MobilenetV2

| Modelo | Clases | Verdadero | Verdadero | Falso | Falso |
|-------------|-------------|-----------|-----------|----------|----------|
| | | Positivo | Negativo | Positivo | Negativo |
| MobilenetV2 | Puccinia | 101 | 5 | 6 | 5 |
| | sorgui | | | | |
| | Spiroplasma | 112 | 14 | 10 | 8 |
| | kunkeli | | | | |
| | Sana | 70 | 5 | 6 | 4 |

5.1.3.1. Matriz de Confusión del Modelo MobilenetV2

La matriz de confusión proporciona una representación del rendimiento del modelo al mostrar la distribución de las predicciones de las diferentes clases

Figura 11

Matriz de Confusión del Modelo MobileNetV2

Matriz de confusion

| | | | | |
|--------|------------------------|-----------------|------|---------------------|
| Clases | Puccinia sorgui | 113 | 1 | 3 |
| | Sana | 0 | 84 | 1 |
| | Spiroplasma kunkeli | 4 | 1 | 139 |
| | | Puccinia sorgui | Sana | Spiroplasma Kunkeli |
| | | Predicciones | | |

Nota. Adaptado de la revista de ciencia y tecnología de la universidad de Palermo de (Navarro & Martínez, 2022).

5.1.3.2. Precisión del Modelo MobilenetV2

Se realiza la evaluación de la precisión para medir la proporción de verdaderos positivos entre todas las instancias que el modelo ha clasificado como positivas.

$$P0 = \frac{TP}{TP + FP} = \frac{101}{101 + 6} = 0.943$$

$$P1 = \frac{TP}{TP + FP} = \frac{112}{112 + 10} = 0.918$$

$$P2 = \frac{TP}{TP + FP} = \frac{70}{70 + 6} = 0.921$$

El valor de la precisión del modelo MobilenetV2 es de 94.3% para puccinia sorghi, 91.8% para spiroplasma kunkeli y 92.1% para la clase sana, lo que indica la porción de predicciones correctas.

5.1.3.3. Exactitud (Accuracy) del Modelo MobilenetV2

La evaluación de la exactitud mide la proporción de predicciones correctas entre todas las predicciones realizadas

$$E = \frac{315}{346} = 0.910$$

El valor de la exactitud (accuracy) obtenido por el modelo es de 91.0% lo que indica la proporción de las predicciones correctas, tanto los verdaderos positivos como los verdaderos negativos.

5.1.3.4. Sensibilidad (Recall) del Modelo MobilenetV2

Esta métrica también llamada recall o exhaustividad mide la proporción de los verdaderos positivos que son correctamente identificados por el modelo

$$R0 = \frac{TP}{TP + FN} = \frac{101}{101 + 5} = 0.952$$

$$R1 = \frac{TP}{TP + FN} = \frac{112}{112 + 8} = 0.933$$

$$R2 = \frac{TP}{TP + FN} = \frac{70}{70 + 4} = 0.945$$

La sensibilidad o recall obtenido con el modelo es de 95.2% para la clase puccinia sorghi, 93.3% para la clase spiroplasma y 94.5% para la clase sana, por lo que indica que el modelo es capaz de identificar las instancias positivas.

5.1.3.5. F1-score del Modelo MobilenetV2

Se realiza la evaluación de esta métrica para determinar el balance entre la precisión y la sensibilidad del modelo.

$$F0 = \frac{2 \times TP}{2 \times TP + FP + FN} = \frac{2 \times 101}{2 \times 101 + 6 + 5} = 0.948$$

$$F1 = \frac{2 \times TP}{2 \times TP + FP + FN} = \frac{2 \times 112}{2 \times 112 + 10 + 8} = 0.925$$

$$F2 = \frac{2 \times TP}{2 \times TP + FP + FN} = \frac{2 \times 70}{2 \times 70 + 6 + 4} = 0.933$$

El valor obtenido por medio de esta métrica es de 94.8% en la clase puccinia sorghi, 92.5% para spiroplasma kunkeli y 93.3% para la clase sana, por lo que la media armónica de la precisión y la sensibilidad es equilibrada.

5.1.4. Evaluación de la Taza de Error de los Modelos

La taza de error de los modelos en la identificación de enfermedades: Puccinia sorghi (roya común), Spiroplasma kunkeli (puka poncho) y hoja sana, es la proporción de las predicciones incorrectas, las cuales fueron evaluadas para determinar cómo los modelos están fallando.

Tabla 41*Taza de Error*

| Modelo | Taza de Error |
|---------------|----------------------|
| Implementado | 9% |
| VGG16 | 3.8% |
| MobileNetV2 | 6.1% |

Los modelos VGG16, MobilenetV2 y el implementado para la identificación de Puccinia sorghi (roya común), Spiroplasma kunkeli (puka poncho) y hoja sana, demostraron buenos resultados en la clasificación. Por lo general, VGG16 tiene la tasa de error más baja con 3.8%, lo que indica que tiene un mejor rendimiento que los otros dos modelos. MobilenetV2 es el segundo modelo que tiene la tasa de error más baja y el modelo implementado tiene la tasa de error más alta entre los tres modelos.

5.2. Evaluación de la Aplicación Móvil

La evaluación de la aplicación móvil se realiza para determinar el nivel de precisión, sensibilidad, recall y F1-score, con el modelo VGG16, ya que es el modelo que obtuvo mejores resultados en el rendimiento, además de tener una menor tasa de error.

Tabla 42*Resultados de la Aplicación Móvil*

| Clases | Verdadero Positivo | Verdadero Negativo | Falso Positivo | Falso Negativo |
|---------------------|---------------------------|---------------------------|-----------------------|-----------------------|
| Puccinia sorgui | 101 | 5 | 8 | 3 |
| Spiroplasma kunkeli | 119 | 10 | 7 | 8 |
| Sana | 75 | 3 | 4 | 3 |

5.2.1. Precisión de la Aplicación Móvil

La evaluación se está métrica busca medir la proporción de verdaderos positivos entre todos los casos.

$$P0 = \frac{TP}{TP + FP} = \frac{101}{101 + 8} = 0.926$$

$$P1 = \frac{TP}{TP + FP} = \frac{119}{119 + 7} = 0.944$$

$$P2 = \frac{TP}{TP + FP} = \frac{75}{75 + 4} = 0.949$$

El nivel de precisión de la aplicación móvil es de 92.6% en la clase puccinia sorgi, 94.4% para la clase spiroplasma kunkeli y 94.9% de la clase sana por lo que el modelo tiene pocos falsos positivos.

5.2.2. Exactitud (Accuracy) de la Aplicación Móvil

La evaluación de la métrica de exactitud mide la proporción de las clasificaciones correctas sobre el total de las predicciones.

$$E = \frac{316}{346} = 0.913$$

El nivel de exactitud obtenido por la aplicación móvil mide el desempeño del modelo por lo cual el valor obtenido es de 91.3%.

5.2.3. Sensibilidad (Recall) de la Aplicación Móvil

La métrica sensibilidad mide la proporción de las clasificaciones correctas entre los casos reales positivos.

$$R0 = \frac{TP}{TP + FN} = \frac{101}{101 + 3} = 0.971$$

$$R1 = \frac{TP}{TP + FN} = \frac{119}{119 + 8} = 0.937$$

$$R2 = \frac{TP}{TP + FN} = \frac{75}{75 + 3} = 0.961$$

El nivel de sensibilidad obtenido es de 97.1% en la clase puccinia sorgui, 93.7% en la clase spiroplasma kunkeli y 96.1% en la clase sana por lo que el modelo tiene pocos falsos negativos.

5.2.4. F1-score de la Aplicación Móvil

Esta métrica proporciona una medida equilibrada entre la sensibilidad y la precisión, siendo la media armónica de ambas.

$$F = \frac{2 \times TP}{2 \times TP + FP + FN} = \frac{2 \times 101}{2 \times 101 + 8 + 3} = 0.948$$

$$F = \frac{2 \times TP}{2 \times TP + FP + FN} = \frac{2 \times 119}{2 \times 119 + 7 + 8} = 0.940$$

$$F = \frac{2 \times TP}{2 \times TP + FP + FN} = \frac{2 \times 75}{2 \times 75 + 4 + 3} = 0.955$$

El resultado del F1-score es de 94.8% para clase puccinia sorgui, 94% para la clase spiroplasma kunkeli y 95.5% para la clase sana, por lo que indica un buen balance entre la precisión y la sensibilidad

5.3. Resultados de la ISO/IEC 25000

5.3.1. Resultados de la Evaluación de la Subcaracterísticas de Calidad

El estándar ISO/IEC 25000 establece un marco para evaluar la calidad del software definiendo un conjunto de subcaracterísticas que cubren aspectos específicos de cada característica principal; La siguiente tabla presenta los resultados de la evaluación y el seguimiento de las subcaracterísticas de calidad.

Tabla 43

Resultados de las Subcaracterísticas de Calidad

| Sub característica | Característica | Nivel de rating | Observación de la evaluación |
|---------------------------|-----------------------|------------------------|---|
| Integridad | Seguridad | ACEPTABLE | Se concluye que la aplicación móvil "SickCorn" garantiza la exactitud de los datos almacenados de los usuarios. |

| | | | |
|--|-------------------------|-----------|--|
| Autenticidad | Seguridad | ACEPTABLE | Se concluye que la aplicación móvil “SickCorn” valida la identidad de usuario, impidiendo accesos no autorizados. |
| Utilización de recursos | Eficiencia de desempeño | PLANEADO | Se concluye que la aplicación móvil “SickCorn” tiene un buen desempeño en el uso de memoria. |
| Capacidad | Rendimiento | PLANEADO | Se concluye que la aplicación móvil “SickCorn” tiene un buen desempeño en el tiempo de respuesta del análisis. |
| Comportamiento temporal | Eficiencia de desempeño | PLANEADO | Se concluye que la aplicación móvil “SickCorn” se desempeña de forma rápida con los recursos necesarios |
| Operabilidad | Usabilidad | PLANEADO | Se concluye que la aplicación móvil “SickCorn” realiza las funciones establecidas en los requerimientos. |
| Inteligibilidad | Usabilidad | PLANEADO | Se concluye que la aplicación móvil “SickCorn” posee la capacidad de permitir al usuario entender si es adecuada para sus necesidades. |
| Protección frente a errores de usuario | Usabilidad | ACEPTABLE | Se concluye que la aplicación móvil “SickCorn” posee la capacidad para proteger a los usuarios de cometer errores. |
| Aprendizaje | Usabilidad | ACEPTABLE | Se concluye que la aplicación móvil “SickCorn” es sencillo al uso del usuario. |

| | | | |
|---------------|------------|----------|--|
| Accesibilidad | Usabilidad | PLANEADO | Se concluye que la aplicación móvil “SickCorn” está disponible para los usuarios. |
| Estética | Usabilidad | PLANEADO | Se concluye que la aplicación móvil “SickCorn” posee la capacidad de agradar al usuario. |

5.3.2. Resultados de la Evaluación de la Característica de Calidad

El estándar ISO/IEC 25000 define un conjunto de características de calidad del software que se evalúan para medir la calidad de la aplicación móvil SickCorn y por lo tanto se informa los resultados de la evaluación y seguimiento de las características de calidad.

Tabla 44

Resultados de la Característica de Calidad

| Características | Nivel de rating | Observaciones de la evaluación |
|--------------------------|------------------------|--|
| Seguridad | ACEPTABLE | La aplicación móvil “SickCorn” mantiene la integridad de los datos y la verificación del usuario |
| Eficiencia del desempeño | PLANEADO | La aplicación móvil “SickCorn” tiene la capacidad de realizar sus funciones de manera efectiva |
| Usabilidad | PLANEADO | La aplicación móvil “SickCorn” tiene la facilidad de que los usuarios puedan utilizarla. |

Esta calificación se basa en los criterios establecidos en el estándar ISO/IEC 25000 para evaluar la calidad del software y refleja el compromiso de la aplicación móvil SickCorn de garantizar un uso seguro y eficiente.

Capítulo 6

Discusión

De acuerdo a los resultados obtenidos, al evaluar el objetivo general propuesto, en el cual se obtuvo el 92.6% de nivel de precisión en la detección de puccinia sorgui y 94.4% de precisión en la detección de spiroplasma kunkeli con la aplicación móvil “SickCorn” en las plantas de maíz del distrito de Talavera.

Por otro lado, Salinas y Alva (2023), en su trabajo de investigación, sostienen que, con una cantidad suficiente de datos y el uso de técnicas adecuadas, los modelos de redes neuronales convolucionales proporcionan resultados más precisos. Esto se debe a que, al contar con una mayor cantidad de datos, el modelo tiende a generalizar mejor, disminuyendo el nivel de overfitting. Es por ello por lo que, en la presente investigación, se recopilaron los datos por medio de la técnica de observación directa. Posteriormente, se aplicaron técnicas de preprocesamiento de datos, tales como el filtro gaussiano, el filtro de nitidez, segmentación por el algoritmo Canny y el resaltado HSV, con la finalidad de mejorar la calidad de las imágenes obtenidas. También se utilizó la técnica de data augmentation (aumento de datos), debido a que la cantidad de datos era insuficiente para entrenar los modelos. Esta técnica se empleó para incrementar la cantidad y diversidad de datos.

De la misma manera, Córdoba y Periche (2022) en su investigación realizaron la evaluación de algunos modelos en la clasificación de hojas de maíz con gusano, roya común, Tizón de la hoja de maíz y sana, en la cual realizaron la evaluación del rendimiento de los modelos DesNet201, MobilenetV2, ResNet50 y EfficientNetVB0 obteniendo como mejor modelo DesNet201 que tiene una precisión de 97% para la clase de Gusano, para la Roya común 95%, Tizón de la hoja 98% y planta saludable 73 % de precisión, de forma similar a la presente investigación en la cual se realizó la evaluación del rendimiento del modelo implementado CNN, MobilenetV2 y VGG16, en la cual se obtuvo el modelo con mejor rendimiento es VGG16, obteniendo un nivel de precisión de la clase

Puccinia sorghi, de 96.2% en la planta sana 93.5% y la clase *Spiroplasma kunkeli* (puka poncho) 95.5%.

De la misma manera, Caiza & Lema (2020) sostienen que el uso de la metodología de desarrollo Mobile-D permite el desarrollo ágil de software pequeño en poco tiempo con menor cantidad de recursos, con el cual desarrollaron una aplicación móvil para la detección y medición de altura en plantas de maíz utilizando Deep learning. Debido a que la aplicación móvil permite tener la experiencia de manipular tecnologías nuevas e innovadoras al alcance del usuario, es por ello por lo que en esta investigación también se hace uso de dicha metodología, debido a que se combina los principios de las metodologías ágiles con prácticas específicas para el desarrollo, lo que permite responder a los cambios del entorno asegurando la calidad del producto.

Asimismo, los resultados guardan relación con la investigación de Ordoñez y Bustamante (2018), en la cual los autores lograron un nivel de precisión del 94.34%, una sensibilidad del 100%, una exactitud del 94.94% y un índice de especificidad del 66.67% con su aplicación móvil que permite la detección y sugerencia de tratamientos para enfermedades en cultivos de maíz. De la misma forma, dicho estudio concuerda con el presente al realizar el análisis del rendimiento de la aplicación móvil, en la que se obtuvo un nivel de precisión del 92.6% en la clase *puccinia sorghi*, 94.4% para la clase *spiroplasma kunkeli* y 94.9% en la clase sana, además obtuvo una sensibilidad del 97.1% en la clase *puccinia sorghi*, 93.7% para *spiroplasma kunkeli* y 96.1 para sana. La aplicación móvil obtuvo una exactitud del 91.3% y un F1-score del 94.8% en la clase *puccinia sorghi*, 94.0% para *spiroplasma kunkeli* y 95.5% para sana. Los resultados de ambas investigaciones demuestran que el uso de las redes neuronales convolucionales es ampliamente utilizado en tareas de visión por computadora, siendo una herramienta fundamental para desarrollar soluciones innovadoras.

Conclusiones

- La recolección de datos se realizó por medio de la técnica de observación directa, utilizando un dispositivo móvil para la captura de las imágenes. Para continuar con el preprocesamiento de los datos recolectados se utilizó la técnica de histograma, el filtro gaussiano, el filtro de nitidez, la segmentación por el algoritmo Canny y el resaltado HSV para preparar los datos que se utilizaron para entrenar los modelos.
- Al concluir el preprocesamiento de los datos se realizó el entrenamiento de los modelos, los cuales fueron evaluados por medio de las métricas de rendimiento: precisión, exactitud, sensibilidad y F1-score, obteniendo mejores resultados del modelo VGG16 en cuanto al modelo implementado y al modelo MobilenetV2.
- Se implementó la aplicación móvil “SickCorn” mediante la metodología Mobile-D, para la detección de *Puccinia sorghi* (Roya común), *Spiroplasma kunkeli* (Puka poncho) en las plantas de maíz del distrito de Talavera.
- Se concluye, respecto al objetivo general, que la aplicación móvil “SickCorn” tiene una precisión del 94.7% en la identificación de *Puccinia sorghi* (roya común) y *Spiroplasma kunkeli* (puka poncho) en las plantas de maíz del Distrito de Talavera.

Recomendaciones

- Se recomienda en futuras investigaciones incrementar la cantidad de datos para el entrenamiento de los modelos.
- Por otro lado, en cuanto a la aplicación móvil, sería recomendable incrementar la cantidad de módulos en la cual se abarque el tratamiento de las enfermedades como sus métodos de control.
- Asimismo, sería recomendable agregar las zonas geográficas en las cuales se tuvo mayor incidencia de las enfermedades

JELH

Referencias Bibliográficas

- Amaya B, Y. D. (14 de Noviembre de 2013). *Dialnet*.
<https://dialnet.unirioja.es/servlet/articulo?codigo=6041502>
- Amazon. (2023). *A WS, que es la vision artificial*. <https://aws.amazon.com/es/what-is/computer-vision/>
- Artola M, Á. (2019). *Deposito de Investigacion Universidad de Sevilla*
<https://hdl.handle.net/11441/89506>
- ASPROMAD. (2015). *Plataforma del Estado Peruano*.
https://bioseguridad.minam.gob.pe/wpcontent/uploads/2017/02/maiz_13regiones_2015.pdf
- AWS. (2023). AWS: <https://aws.amazon.com/es/what-is/data-preparation/>
- Agudelo, F., & Sarria V , G. A. (2006-2007). *Instituto Colombiano Agropecuario*.
<https://www.ica.gov.co/getattachment/f1c1f3f1-d775-4216-a5d0-d9d4a67b7943/Publicacion-8.aspx>
- Baz A, A., Ferreira A, I., Álvarez R, M., & García B, R. (2011). *Repositorio uniovi, ingeniería de telecomunicación*. http://isa.uniovi.es/docencia/SIGC/pdf/telefonía_movil.pdf
- Baldoceca C, J. C. (Noviembre de 2017). *Repositorio Universidad Inca Garcilaso de la Vega*.
<http://repositorio.uigv.edu.pe/handle/20.500.11818/1800>
- Barandiarán G, M. A. (Abril de 2020). *Repositorio INIA*: <https://hdl.handle.net/20.500.12955/1643>
- Basogain O, X. (1998). *UPVEHU Redes Neuronales Artificiales*.
https://ocw.ehu.eus/pluginfile.php/40137/mod_resource/content/1/redes_neuro/contenidos/pdf/libro-del-curso.pdf
- Baz , A. A., Ferreira A, I., & Alvarez R, M. (2011). *MEMORIA Universidad de Oviedo* .
http://isa.uniovi.es/docencia/SIGC/pdf/telefonía_movil.pdf
- Bisetti R, S. N. (2023). *Repositorio Universidad Agraria la Molina*.
<https://hdl.handle.net/20.500.12996/5865>
- Bonilla C, C. (01 de Junio de 2020). *Deposito de Investigacion Universidad de Sevilla* .
<https://hdl.handle.net/11441/115221>
- Bono C, R. (14 de Septiembre de 2012). *Repositorio Universidad de Barcelona* .
<http://hdl.handle.net/2445/30783>
- Calero M, R. G. (Marzo de 2022). *Biblioteca General Universisdad Tecnica de Cotopaxi*.
<http://repositorio.utc.edu.ec/handle/27000/8436>

- Calvo, J., & Taboares, W. (2014). *Repositorio Institucional de Innovacion Agraria (INIA)*.
<https://hdl.handle.net/20.500.12955/112>
- Calvo, L. (27 de Diciembre de 2023). *GoDaddy Tecnologia*.
<https://www.godaddy.com/resources/es/crearweb/que-es-una-app-y-para-que-se-utiliza>
- Canchachi H, D. F., & Mendoza A, J. T. (2020). *Universidad César Vallejo Repositorio Institucional*. <https://hdl.handle.net/20.500.12692/50550>
- Castro, C., & McNab, A. (2004). *CIMMYT*. <http://hdl.handle.net/10883/715>
- Ccopa C, L. O. (2019). *Library*. <https://library.co/document/y9gvl4lq-aplicacion-movil-gestion-denuncias-efectivos-policiales-policia-nacional.html>
- Chiluisa, G. D. (20 de Julio de 2021). *Repositorio Digital Universidad Técnica del Norte*.
<http://repositorio.utn.edu.ec/handle/123456789/11319>
- CIMMYT. (2004). *Repositorio cimmyt*.
<https://repository.cimmyt.org/xmlui/bitstream/handle/10883/715/25905.pdf>
- Orbera, T., Serrat, M., & Ortega, E. (Junio de 2014). *Medigraphic*.
<https://www.medigraphic.com/pdfs/biotecapl/ba-2014/ba141b.pdf>
- Del Carpio, A. M., & Nuñez, A. E. (2022). *Repositorio Institucional RI-UNPHU*.
<https://repositorio.unphu.edu.do/handle/123456789/4825>
- Diaz, G. I. (Septiembre de 2017). *Repositorio Universidad de Guayaquil*.
<https://repositorio.ug.edu.ec/bitstreams/b98677d9-8a67-4fe5-9e74-218c02ee88b5/download>
- Diaz H, R., & Vilcas V, D. R. (21 de Noviembre de 2022). *Repositorio Universidad Nacional Amazonica de Madre de Dios*. <http://hdl.handle.net/20.500.14070/889>
- DuocUC. (06 de Marzo de 2024). *DuocUC Bibliotecas: Investigación Aplicada*.
<https://bibliotecas.duoc.cl/investigacion-aplicada>
- Durán S, J. (2017). *Deposito de Investigacion Universidad de Sevilla*.
<https://hdl.handle.net/11441/69564>
- Dzul E, M. (2008). *Sistema de Universidad Virtual (Universidad Autonoma del Estado de Hidalgo)*.
https://www.uaeh.edu.mx/docencia/VI_Presentaciones/licenciatura_en_mercadotecnia/fundamentos_de_metodologia_investigacion/PRES38.pdf
- Enciso O, S. E., Mamani V, E., & Ordoñez R, E. (15 de 10 de 2023). *C&T Riqchary (Revista de investigación en Ciencia y Tecnología) UNAMBA*.
<https://doi.org/10.57166/riqchary/v5.n1.2023.104>

- Enriquez, J. G., & Casas, S. I. (2013). *Dialnet*
<https://dialnet.unirioja.es/descarga/articulo/5123524.pdf>
- Fassio, A., Carriquiry, A. I., Tojo, C., & Romero, R. (Diciembre de 1998). *Instituto Nacional de Investigacion Agropecuaria Uruguay*.
<http://inia.uy/en/Publicaciones/Documentos%20compartidos/111219240807135855.pdf>
- Garcia, C. (21 de Julio de 2020). *agraria*. <https://agraria.pe/noticias/el-maiz-es-el-cultivo-mas-importante-en-extension-para-el-pe-22033>
- Grobman, A. (13 de Febrero de 2012). *universia (Vida Universitaria)*.
<https://www.universia.net.pe/actualidad/vida-universitaria/maiz-se-origino-peru-mexico-911136.html>
- Grobman, A., Salhuana, W., Sevilla, R., & Mangelsdorf, P. (1961). *National Academies*.
<https://nap.nationalacademies.org/catalog/21228/races-of-maize-in-peru-their-origins-evolution-and-classification>
- Hernandez M, A. (01 de Diciembre de 2021). *Repositorio Institucional del Tecnológico Nacional de México*. <https://rinacional.tecnm.mx/jspui/handle/TecNM/5144>
- Hernández S, R., Fernández C, C., & Baptista L, M. (2014). *Metodología de la investigación*. McGRAW HILL Interamericana Editores, S.A. DE. C. V. <https://doi.org/04/14>
- Hernández S, R., Fernández C, C., & Baptista L, P. (2014). *Metdología de la Investigación*. McGraw Hill Education. <https://doi.org/978-1-4562-2396-0>
- Hernández, A. (2003). *Dialnet*. <https://dialnet.unirioja.es/servlet/articulo?codigo=793097>
- Huerta, E., Gonzalez H, V., Rodriguez R, H., & Amabilis S, E. (2020). *RINDERESU (Revista Internacional de Desarrollo Regional Sustentable)*.
<http://rinderesu.com/index.php/rinderesu/article/view/46/50>
- INEI. (2022). *Instituto Nacional de Estadística e Informática*.
<https://m.inei.gob.pe/prensa/noticias/produccion-de-maiz-amilaceo-crecio-258-y-totalizo-78-mil-285-toneladas-durante-julio-del-2022-13924/>
- inia. (6 de Abril de 2004). *Repositorio inia*. https://www.inia.gob.pe/wp-content/uploads/investigacion/programa/sistProductivo/variedad/maiz-amilaceo/INIA_606.pdf
- INIA. (2020). *Repositorio inia*. <https://pgc-snia.inia.gob.pe:8443/jspui/handle/20.500.12955/1310>
- ISO 25000. (2019). *ISO 25000*. Obtenido de ISO 25000: <https://iso25000.com/index.php/normas-iso-25000>
- Jara C, T. W. (2014). *Repositorio INIA*. <https://hdl.handle.net/20.500.12955/112>

- Jaramillo T, P. A. (2017). *Repositorio RiUTPL*.
<http://dspace.utpl.edu.ec/handle/20.500.11962/20730>
- Larrañaga, P., Inza, I., & Moujahid, A. (s.f.). *UPV/EHU Redes Neuronales*. Recuperado el 9 de Marzo de 2024, de UPV/EHU Redes Neuronales:
<http://www.sc.ehu.es/ccwbayes/docencia/mmcc/docs/t8neuronales.pdf>
- Lisandro, D., Galdamez, N., T. P., & Pesado, P. (2013). *Repositorio institucional de la UNLP*.
<http://sedici.unlp.edu.ar/handle/10915/32397>
- Lozada P, W. A., Suarez B, M. J., & Avendaño F, E. (2021). Aplicación de redes neuronales convolucionales para la detección del tizón tardío *Phytophthora infestans* en papa *Solanum tuberosum*. *Revista U.D.C.A Actualidad & Divulgación Científica*, 24(2), 9.
<https://doi.org/0123-4226>
- Lozada, J. (2014). *Dialnet*. <https://dialnet.unirioja.es/servlet/articulo?codigo=6163749>
- Luján C, D. (2019). Desarrollo de aplicacion Android con Android Studio: Conoce Android Studio. En J. D. Luján Castillo, *Desarrollo de aplicacion Android con Android Studio: Conoce Android Studio* (pág. 286). AlphaEditorial.
- Martínez L, J. (2018). *Archivo digital UPM*. <https://oa.upm.es/53050/>
- Martínez L, J. (2018). *Archivo Digital UPM (Universidad Politecnica de Madrid)*.
<https://oa.upm.es/53050/>
- Medina C, J. M., & Urteaga M, J. A. (7 de Julio de 2021). *Repositorio Institucional UPN*.
<https://hdl.handle.net/11537/28251>
- midagri. (2021). *Sistema Integrado de Estadística Agraria*.
<https://sica.midagri.gob.pe/portal/publicacion/boletines-diarios?download=1639:agricola-2021>
- milenium. (2020). *Informática Milenium*.
<https://www.informaticamilenium.com.mx/es/temas/que-es-una-app.html>
- Miranda T, A. J. (2023). *Repositorio Universidad Internacional de Andalucía*.
<http://hdl.handle.net/10334/8167>
- Modi, A. (17 de Febrero de 2021). *CYNOTEK*. https://cynoteck.com/es/author/amit_modi/
- Molina R, J. R., Honores T, J. A., Pereira S, N., & Pardo L, H. P. (Septiembre de 2021). *Repositorio Unidversidad de Coruña*.
https://ruc.udc.es/dspace/bitstream/handle/2183/28449/Pedreira-Souto_Nieves_2021_Metodolog%C3%ADas_aplicaciones_m%C3%B3viles.pdf?sequence=3&isAllowed=y

- Molina R, J. R., Loja M, N. M., & Loaiza S, E. L. (2016). *Revistas UNLA (Universidad Tecnica de Machala)*. <https://doi.org/10.18294/relais.2016.201-207>
- Müller, M., Charypar, D., & Gross, M. (2003). Particle Based Fluid Simulation for Interactive Applications. *ACM SIGGRAPH / Eurographics Symposium on Computer Animation*.
- Nacelle, A. (2009). *Nucleo de Ingenieria Biomedica*.
<http://www.nib.fmed.edu.uy/Seminario%202009/Monografias%20seminario%202009/Nacelle-Redes%20NeuronalesImplementacion.pdf>
- Navarro L, T. P., & Piña D, P. C. (Marzo de 2021). *Repositorio INIA*.
<https://hdl.handle.net/20.500.12955/1310>
- Navarro O, D. J., & Martinez L, S. A. (2022). *Dialnet*.
<https://dialnet.unirioja.es/servlet/articulo?codigo=8735079>
- NPIC. (17 de Septiembre de 2021). *National Pesticide Information Center*.
<http://npic.orst.edu/pest/index.es.html>
- Ñaupas P, H., Mejía M, E., Novoa R, E., & Villagómez P, A. (2013). *Metodología de la Investigación*. Ediciones de la U - Transversal . <https://doi.org/978-958-762-188-4>
- Ordóñez , P. A. (Agosto de 2018). *Repositorio Institucional de la UNLP*.
<http://sedici.unlp.edu.ar/handle/10915/82406>
- Otzen, T., & Manterola, C. (2017). *SCIELO*. <http://dx.doi.org/10.4067/S0717-95022017000100037>
- Otzen, T., & Manterola, C. (Marzo de 2017). *Scielo*. <http://dx.doi.org/10.4067/S0717-95022017000100037>
- Paez, L. (21 de Mayo de 2021). *Crehana Futureof People*.
<https://www.crehana.com/blog/transformacion-digital/tipos-sistemas-operativos/>
- Peña R, J. C., Camargo M, S., & Osorio, J. O. (Mayo de 2016). *Repositorio udistrital*.
<https://repository.udistrital.edu.co/bitstream/handle/11349/4789/Prototipo%20de%20aplicativo%20movil%20para%20la%20gestion%20de%20Vales%20de%20compra%20y%20entrega%20de%20productos%20en%20Corabastos.pdf;jsessionid=9945C4CAD29740A0EB60D60E16B2FF14?squence>
- Péres C, R. D. (2018). *Repositorio Institucional Universidad de Pamplona*.
<http://repositoriodspace.unipamplona.edu.co/jspui/handle/20.500.12744/5240>
- Pino D, R., De la Fuente G, D., Parreño F , J., & Priore, P. (2004). *Dialnet*.
<https://dialnet.unirioja.es/descarga/articulo/1032033.pdf>
- Pino D, R., Gomez G, A., & Martínez, N. (2001). *Google Libros*.
<https://books.google.es/books?hl=es&lr=&id=RKqLMCw3IUkC&oi=fnd&pg=PA1&dq=La+I>

A+es+un+campo+de+la+ciencia+y+la+ingenier%C3%ADa+que+se+ocupa+de+la+comprensi%C3%B3n+desde+el+punto+de+vista+inform%C3%A1tico+de+lo+que+se+denomina+com%C3%BAmente+comportamiento

Quevedo W, S. (Junio de 2013). *Repositorio midagri*.

https://repositorio.midagri.gob.pe/bitstream/20.500.13036/693/1/Manual...maiz_blanco_Urubamba.pdf

Rollins, J. B. (Junio de 2015). *IBM*. <https://www.ibm.com/downloads/cas/6RZMKDN8>

Salhuana M, W. (Mayo de 1942). *Servicio de Investigacion Agricola (USD-ARS)*.

https://www.ars.usda.gov/ARSTUserFiles/50301000/Races_of_Maize/Diversidad%20y%20razas%20de%20maiz%20en%20Peru.pdf

Salhuana, W. (2020). *United States Department of Agriculture*.

https://www.ars.usda.gov/ARSTUserFiles/50301000/Races_of_Maize/Diversidad%20y%20razas%20de%20maiz%20en%20Peru.pdf

Salinas G, R. (Septiembre de 2021). *Repositorio Universidad Autonoma del Estado de Morelos*.

<http://riaa.uaem.mx/handle/20.500.12055/2138>

Seidaliyeva, U., Akhmetov, D., Iipbayeva, L., & Matson, E. (2020). *ResearchGate*.

<https://www.researchgate.net/scientific-contributions/Daryn-Akhmetov-2177682432>

Serrato, C. (7 de Junio de 2021). *INMEDIATUM*. <https://inmediatum.com/blog/estrategia/cuales-son-los-componentes-basicos-de-la-inteligencia-artificial/>

Serratos H, j. A. (Enero de 2009). *funsepa origen del maíz*.

<http://www.funsepa.net/guatemala/docs/el-origen-y-la-diversidad-del.pdf>

Solenthaler, B., Schläfli, J., & Pajarola, R. (2007). A Unified Particle Model for Fluid–Solid Interactions: Research Article. *Computer Animation and Virtual Worlds*.

Sugata, T., & Yang, C. (25 de Agosto de 2017). Leaf App: Leaf recognition with deep convolutional neural networks. *IOPSCIENCE*, 273(012004), 7.

<https://doi.org/10.1088/1757-899X/1012004>

Takahiro, H., Seiichi, K., & Yoichiro, K. (2007). Smoothed Particle Hydrodynamics in Complex Shapes. *Proceedings of the 23rd Spring Conference on Computer Graphics*. Budmerice, Slovakia.

Tamayo. (2007). *Virtual URBE (Universidad Privada Rafael Bellosó Chacín)*.

<https://virtual.urbe.edu/tesispub/0092769/cap03.pdf>

Torrey, L., & Shavlik, J. (2010). *IGI Global*. <https://www.igi-global.com/chapter/transfer-learning/36988>

- UNAM. (Diciembre de 2023). *Universidad Nacional Autonoma de Mexico: Lenguaje de programacion JAVA*. Obtenido de Universidad Nacional Autonoma de Mexico: Lenguaje de programacion JAVA: <https://docencia.tic.unam.mx/presenciales/Lenguaje-de-programacion-java.html>
- Unisem. (2022). *unisem articulos*. <https://semillastodoterreno.com/articulos/page/2>
- Van Dokkum, K., Koomen, K., Ten Katen, H., & Pinkster, J. (2008). *Ship Stability, 3rd edition*. DokMar: Enkhuizen.
- Varón A , F., & Sarria Villa, A. G. (2006-2007). *Ministerio de Agricultura y Desarrollo Rural*. <https://www.ica.gov.co/getattachment/f1c1f3f1-d775-4216-a5d0-d9d4a67b7943/Publicacion-8.aspx>
- Veintimilla P, D. J., & Sigüencia C, Y. F. (Julio de 2014). *BIBDigital Escuela Politecnica Nacional*. <http://bibdigital.epn.edu.ec/handle/15000/7853>
- Verdeguer M, A. (1985). *Ministerio de Agricultura, Pesca y Alimentación*. https://www.mapa.gob.es/ministerio/pags/biblioteca/hojas/hd_1985_04.pdf
- Villalón P, D. (2021). *Universidad Politecnica de Valencia*. <http://hdl.handle.net/10251/174515>
- Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). *NeurIPS Proceedings*. https://proceedings.neurips.cc/paper_files/paper/2014/hash/375c71349b295fbe2dedca9206f20a06-Abstract.html
- Zapata. (2006). *Virtual URBE (Universidad Privada Rafael Belloso Chacín)*. [http://virtual.urbe.edu/tesispub/0106891/cap03.pdf?fbclid=IwAR1LkHQ6gUgV7SZ22dEaAQueO1ojjEMgex4zU5M0Iix-tO2VUGBjadEH02I#:~:text=Seg%C3%BAAn%20Zapata%20\(2006\)%2C%20las,de%20operaci%C3%B3n%20que%20permita%20manipular%E2%80%9D](http://virtual.urbe.edu/tesispub/0106891/cap03.pdf?fbclid=IwAR1LkHQ6gUgV7SZ22dEaAQueO1ojjEMgex4zU5M0Iix-tO2VUGBjadEH02I#:~:text=Seg%C3%BAAn%20Zapata%20(2006)%2C%20las,de%20operaci%C3%B3n%20que%20permita%20manipular%E2%80%9D)

Anexo

Anexo 1

Matriz de Consistencia

| PROBLEMA DE INVESTIGACIÓN | OBJETIVOS DE INVESTIGACIÓN | VARIABLES DE INVESTIGACIÓN | DISEÑO DE INVESTIGACIÓN | POBLACIÓN Y MUESTRA DE ESTUDIO |
|---|---|---|---|--|
| <p>PROBLEMA GENERAL</p> <p>¿Cuál es el nivel de precisión de la aplicación móvil (SICKCORN) en la detección de Puccinia sorghi (roya común) y Spiroplasma kunkeli (puka poncho) en las plantas de maíz, distrito de Talavera?</p> <p>PROBLEMAS ESPECÍFICOS</p> <p>¿De qué manera se realiza la recolección y preprocesamiento de datos para la detección de Puccinia sorghi (roya común) y Spiroplasma kunkeli (puka poncho) en las plantas de maíz, distrito de Talavera?</p> <p>¿De qué manera se realizará la evaluación del modelo para la detección de Puccinia sorghi (roya común) y Spiroplasma kunkeli (puka poncho) en las plantas de maíz, distrito de Talavera?</p> <p>¿Cómo implementar la aplicación móvil (SICKCORN), para la detección de Puccinia sorghi (roya común) y Spiroplasma kunkeli (puka poncho) en las plantas de maíz, distrito de Talavera?</p> | <p>OBJETIVO GENERAL</p> <p>Determinar el nivel de precisión de la aplicación móvil (SICKCORN) en la detección de Puccinia sorghi (roya común) y Spiroplasma kunkeli (puka poncho) en las plantas de maíz, distrito de Talavera</p> <p>OBJETIVOS ESPECÍFICOS</p> <p>Recolectar y preprocesar los datos para la detección de Puccinia sorghi (roya común) y Spiroplasma kunkeli (puka poncho) en las plantas de maíz, distrito de Talavera.</p> <p>Evaluar el modelo en la detección de Puccinia sorghi (roya común) y Spiroplasma kunkeli (puka poncho) en las plantas de maíz, distrito de Talavera.</p> <p>Desarrollar la aplicación móvil (SICKCORN) para la detección de Puccinia sorghi (roya común) y Spiroplasma kunkeli (puka poncho) en las plantas de maíz, distrito de Talavera</p> | <p>VARIABLE INDEPENDIENTE</p> <p>Aplicación móvil (SickCorn) con modelo de detección de imágenes de la hoja del maíz</p> <p>VARIABLE DEPENDIENTE</p> <p>Detección de Puccinia sorghi (roya común) y Spiroplasma kunkeli (puka poncho) en plantas de maíz.</p> | <p>TIPO DE INVESTIGACIÓN</p> <p>Aplicada</p> <p>DISEÑO DE INVESTIGACIÓN</p> <p>No experimental</p> <p>ENFOQUE DE INVESTIGACIÓN</p> <p>Cuantitativo</p> | <p>POBLACIÓN</p> <p>Una tarea de terreno con 3500 plantas de maíz del distrito de Talavera</p> <p>MUESTRA</p> <p>La muestra para la presente investigación es de 346 imágenes seleccionadas por el muestreo no probabilístico por conveniencia</p> |

Anexo 2

Constancia de Validación de datos

CONSTANCIA DE VALIDACIÓN DE DATOS

Yo, Fernando Meneses Lujan, con D.N.I. N° 31169951, especialista en Sanidad Vegetal, ostento el grado de Especialista en sanidad vegetal, y ejerzo la carrera profesional en Ing. Agrónomo, Por medio de la presente hago constar que he revisado, con fines de validación, las imágenes de las enfermedades "Spiroplasma kunkeli" (Puca poncho) y "Puccinia sorghi" (Roya común) en las plantas de maíz que será utilizado como datos en la investigación de Jhackelyn Esthefany Leon Hurtado, identificado con D.N.I. N° 71242212

Luego de hacer las verificaciones pertinentes, puedo formular las siguientes apreciaciones:

Evaluación de los datos

| N° | Consideraciones | Valor | |
|----|---|-------|----|
| | | Si | No |
| 1 | Las imágenes de la carpeta Corn_(maize)___SpiroplasmaKunkeli pertenecen a la enfermedad "Spiroplasma Kunkeli" (Puca poncho) en las plantas de maíz. | X | |
| 2 | Las imágenes de la carpeta Corn_(maize)___PucciniaSorghii pertenecen a la enfermedad "Pucinia Sorghi" (Roya Común) en las plantas de maíz. | X | |
| 3 | Las imágenes de la carpeta Corn_(maize)___Desconocido, son distintas enfermedades al Spiroplasma kunkeli y al Puccinia sorghi | X | |

Andahuaylas, 19 de marzo de 2024.



Ing. Fernando Meneses Lujan
ANALISTA SANEAMIENTO DE PLANTAS Y SUELO

6.1. Implementación de los modelos de redes neuronales convolucionales

Se utiliza esta metodología para realizar la implementación y el análisis del modelo implementado por el investigador, VGG16 y MobilenetV2 en la clasificación de *Puccinia sorghi* (roya común), *Spiroplasma kunkeli* (puka poncho) y planta sana en las plantas de maíz, con la finalidad de evaluar la precisión de los modelos y obtener el que tenga mejores resultados para implementarlo en la aplicación móvil “SickCorn”.

6.1.1. Etapa 1 Comprensión del Negocio

6.1.2. Etapa 2 Enfoque Analítico

6.1.3. Etapa 3 Requisitos de Datos

6.1.4. Etapa 4 Recopilación de Datos

6.1.5. Etapa 5 Comprensión de los Datos

6.1.6. Etapa 6 Preparación de Datos

6.1.7. Etapa 7 Modelado

6.1.8. Etapa 8 Evaluación

6.1.9. Etapa 9 Implementación

6.1.10. Etapa 10 Retroalimentación

Se busca la retroalimentación de los profesionales, con el conocimiento del tema o al poder comparar con otros archivos científicos similares.

En la investigación de Bulawit & Palaoag (2023) llevaron a cabo su investigación, en la cual emplearon una red neuronal convolucional (CNN) preentrenada con la que desarrolló una herramienta de reconocimiento en Android para reconocer enfermedades del maíz, por lo que se creó un conjunto de datos de las hojas de maíz sanas y tres enfermedades del maíz (roya común, mancha gris de la hoja y tizón de la hoja) utilizando un conjunto de datos de Plant village e imágenes del campo. Los investigadores utilizaron la técnica de aumento de datos y se realizó una división del

75% y 25%. El modelo se entrenó utilizando la técnica de aprendizaje por transferencia usando el modelo VGG 16. El modelo pudo identificar enfermedades del maíz, obteniendo una precisión del 93.42% y F1-score del 93.53%. La aplicación móvil se desarrolló utilizando la metodología de desarrollo de sistemas dinámicos (DSDM), además, la aplicación se evaluó por medio del estándar de calidad de software ISO 25010.

6.2. Implementación de la ISO/IEC 25000

6.2.1. Requisitos de la Evaluación

6.2.1.1. Propósito de la Evaluación.

El propósito de la evaluación de calidad es obtener el nivel de calidad de la aplicación móvil “SickCorn” para poder identificar sus debilidades y fortalezas de una manera formalmente mensurable, en el marco de un proceso de mejora continua. Teniendo en cuenta el objetivo de la evaluación, los resultados y las conclusiones deberán indicar aspectos a mejorar en el producto de software.

6.2.1.2. Motivación de la Evaluación.

La motivación para la realización de esta evaluación de calidad sobre la aplicación móvil SickCorn es:

- Necesidad de conocer las debilidades y fortalezas que posee la aplicación, en el marco de un proceso de mejora continua
- Necesidad de estudiar características de calidad como usabilidad eficiencia del desempeño y seguridad

6.2.1.3. Tipo de Producto por Evaluar.

El producto de software seleccionado es una aplicación móvil desarrollada para Android versión 11.0 en adelante, denominada SickCorn. Es un producto terminado, más allá de que actualmente se le están realizando modificaciones con el diseño.

6.2.2. Modelo de Calidad

El modelo de calidad que se va a utilizar en la evaluación de calidad para el producto de software seleccionado se encuentra definido en la norma ISO/IEC 25010-System and software quality models, mientras que el proceso de evaluación se encuentra en la norma ISO/IEC 25040-Evaluation process. Teniendo en cuenta el tipo de producto de software a evaluar, se tendrá en cuenta la calidad externa para la evaluación de calidad, y se aplicarán parte de las métricas externas que define la norma ISO/IEC 25023-Measurement of system and software product quality.

6.2.2.1. Selección de las Características por Evaluar.

En el caso concreto de la aplicación móvil SickCorn, dentro de las características que más importan, es la eficiencia del desempeño, la seguridad y la usabilidad, se eligen estas para ejecutar la evaluación de calidad con sus respectivas sub características.

Importancia de la evaluación de la característica eficiencia del desempeño para aplicaciones móviles

Esta característica representa el desempeño relativo a la cantidad de recursos utilizado bajo determinadas condiciones a la que es expuesto la aplicación móvil como los tiempos de respuesta o la cantidad de recursos que necesita para poder funcionar correctamente.

Importancia de la subcaracterística comportamiento temporal para aplicaciones móviles

Esta sub característica hace referencia al tiempo de respuesta y procesamiento de una app de llevar a cabo sus funciones bajo condiciones determinadas en relación a un banco de pruebas establecidas.

Importancia de la subcaracterística utilización de recursos para aplicaciones móviles

La sub característica utilización de recursos hace referencia a la cantidad y tipo de recurso utilizado cuando la app lleva a cabo sus funciones bajo determinadas condiciones.

Importancia de la subcaracterística capacidad para aplicaciones móviles

La sub característica capacidad hace referencia al grado en que los límites máximos de un parámetro de una app cumplen con los requisitos.

Importancia de la evaluación de la característica usabilidad para aplicaciones móviles

La usabilidad es considerada uno de los factores más importantes dentro de la calidad de un producto de software, más específicamente de una aplicación móvil. Con la aparición, el uso masivo y el crecimiento de los dispositivos móviles, especialmente los smartphones, la medición de usabilidad en aplicaciones móviles se tornó un tema de investigación. Particularmente la usabilidad de una “app” se refiere a la facilidad con que los usuarios pueden utilizar la misma para alcanzar un objetivo concreto.

Importancia de la evaluación de la subcaracterística inteligibilidad para aplicaciones móviles

Esta sub característica hace referencia a la capacidad por parte de la app de hacer entender su adecuación para las necesidades del usuario.

Importancia de la evaluación de la subcaracterística aprendizaje para aplicaciones móviles

La sub característica aprendizaje hace referencia a la forma en que el producto de software permite al usuario aprender su uso. Es de suma importancia evaluar esta sub característica, ya que, en el peor de los casos, puede ocurrir que no exista la facilidad de aprendizaje, haciendo que la aplicación pierda usuarios por no poder realizar las tareas que brinda la app.

Importancia de la evaluación de la subcaracterística operabilidad para aplicaciones móviles

La sub característica operabilidad hace referencia a la manera en que el producto de software permite al usuario operarlo y controlarlo. Dicho esto, se observa claramente porque es importante que esta sub característica sea evaluada: el hecho de que la aplicación móvil no pueda ser controlada u operada con facilidad presupone un alto riesgo para la misma, haciendo que cada vez sean menos los usuarios que la usen.

Importancia de la evaluación de la subcaracterística protección frente a errores de usuario para aplicaciones móviles

Esta sub característica hace referencia a la capacidad del sistema para proteger a los usuarios de cometer errores. Si se producen, es importante que se den a conocer al usuario de forma rápida y clara, además de ofrecer algún mecanismo para recuperarse de ese error.

Importancia de la evaluación de la subcaracterística estética para aplicaciones móviles

La sub característica estética hace referencia a la capacidad de la interfaz de usuario de agradar y satisfacer la interacción con el usuario. Esta sub característica es muy importante para ser evaluada, ya que los usuarios se fijan mucho en la manera en que están predispuestas las funcionalidades dentro de la aplicación, o la forma en que se encuentran los iconos, colores, etc, haciendo más fácil su navegación por la aplicación y satisfaciendo sus necesidades.

Importancia de la evaluación de la subcaracterística accesibilidad para aplicaciones móviles

La sub característica accesibilidad hace referencia a la capacidad del producto para ser utilizado por usuarios con determinadas características, permitiendo así que

todas las personas puedan utilizar la aplicación móvil sin problemas, incluso aquellas con limitaciones físicas, visuales, auditivas o de otra índole, y sumando más usuarios.

Importancia de la evaluación de la característica seguridad para aplicaciones móviles

El comportamiento de una aplicación móvil preocupa al usuario desde el punto de vista de la seguridad, ya que estarán disponibles en dispositivos que pueden ser susceptibles de robo o de uso fraudulento. En este sentido, resulta imprescindible asegurar, entre otros factores, que la app mantenga la confidencialidad de los datos privados, la verificación del usuario, los diferentes perfiles que puede tener un usuario dentro de la misma aplicación o el autoguardado de un histórico de operaciones o actividades realizadas.

Importancia de la evaluación de la subcaracterística integridad para aplicaciones móviles

Es importante evaluar esta sub característica para determinar el grado en que un sistema, producto o componente almacena los datos sin alterar o producir cambios

Importancia de la evaluación de la subcaracterística autenticidad para aplicaciones móviles

Las medidas de autenticidad se utilizan para evaluar el grado en que se puede demostrar que la identidad de un sujeto o recurso es la que se atribuye, siendo muy importante evaluar esta sub característica con la finalidad de conocer si la aplicación móvil posee reglas o medidas de autenticidad adecuadas como para darse cuenta si un usuario logeado en el sistema es quien dice ser.

6.2.3. Requerimientos de Calidad

6.2.4. Especificación de la Evaluación

6.2.5. Criterios de Evaluación

6.2.6. Ejecución de la Evaluación

6.2.7. Reporte de la Evaluación

6.2.7.1. Resultado de la Evaluación de las Métricas de Calidad

6.2.7.2. Resultados de la Evaluación de la Subcaracterísticas de Calidad.

6.2.7.3. Resultados de la Evaluación de la Característica de Calidad.

6.2.7.4. Conclusión sobre la Calidad del Producto de Software

Al analizar los resultados obtenidos, se concluye que se han alcanzado los objetivos planificados con las métricas evaluadas en el software "SickCorn". Esto representa un avance significativo en términos de calidad y desempeño del software, en relación con la seguridad, se ha observado que la aplicación mantiene un nivel aceptable, garantizando la integridad de los datos y la autenticación del usuario, en cuanto a la eficiencia del desempeño, se confirma que la aplicación cumple con lo planeado, demostrando una capacidad efectiva para realizar sus funciones de manera óptima, en términos de usabilidad, la aplicación "SickCorn" ofrece una interfaz intuitiva y de fácil navegación, lo que facilita su utilización por parte de los usuarios.

Anexo 3

Código del Histograma

```
import cv2
import numpy as np
import matplotlib.pyplot as plt

def plot_histogram(image):
    # Calcular el histograma
    histogram = cv2.calcHist([image], [0], None, [256], [0, 256])
    plt.figure(figsize=(10, 5))

    plt.subplot(1, 2, 1)
    plt.imshow(image, cmap='gray')
    plt.title("Imagen")
    plt.axis('off')
    plt.subplot(1, 2, 2)
    plt.bar(np.arange(256), histogram[:,0], color='blue', width=1.0)
    plt.title("Histograma de la imagen")
    plt.xlabel("Intensidad de pixeles")
    plt.ylabel("Número de pixeles")
    plt.xlim([0, 256])

    plt.show()

image = cv2.imread('/content/drive/MyDrive/TESIS/DataPlantaVillage/Sana/IMG_20230411_060503.jpg', cv2.IMREAD_COLOR)
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)

plot_histogram(image)
```

Anexo 4

Código del Filtro de Nitidez

```
import cv2
import numpy as np
import matplotlib.pyplot as plt

def nitidez_image(image):
    kernel = np.array([[0.2, -1, 0],
                      [-1, 5, -1],
                      [0, -1, 0.2]])

    sharpened_image = cv2.filter2D(image, -1, kernel)
    return sharpened_image

image = denoised_image
sharpened_image = sharpen_image(image)

image_rgb = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
sharpened_image_rgb = cv2.cvtColor(nitidez_image, cv2.COLOR_BGR2RGB)

plt.figure(figsize=(12, 6))

plt.subplot(1, 2, 1)
plt.imshow(image_rgb)
plt.title('Imagen Original')

plt.subplot(1, 2, 2)
plt.imshow(sharpened_image_rgb)
plt.title('Imagen con Filtro de Nitidez')

plt.show()
```

Anexo 5

Código del Filtro de Gaussiano

```

import cv2
import numpy as np
import matplotlib.pyplot as plt

def denoise_image(image):
    denoised_image = cv2.GaussianBlur(image, (5, 5), 0)
    return denoised_image

denoised_image1 = cv2.imread('/content/drive/MyDrive/TESIS/imagesPrueba/puccinia/20231217_180041.jpg', cv2.IMREAD_COLOR)
denoised_image1 = cv2.cvtColor(denoised_image1, cv2.COLOR_BGR2RGB) # Convertir de BGR (OpenCV) a RGB (matplotlib)
denoised_image = rgb_to_grayscale(denoised_image1)
plt.figure(figsize=(10, 5))

plt.subplot(1, 2, 1)
plt.imshow(denoised_image1, cmap='gray')
plt.title('Imagen original')
plt.axis('off')

plt.subplot(1, 2, 2)
plt.imshow(image_gray, cmap='gray')
plt.title('Imagen sin fondo')
plt.axis('off')

```

Anexo 6

Código de la Conversión a Escala de Grises

```

import cv2
import matplotlib.pyplot as plt

def rgb_grayscale(image):
    gray_image = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)
    return gray_image

image_rgb = cv2.imread('/content/drive/MyDrive/TESIS/DataPlantaVillage/Corn (maize) PucciniaSorghii/20231217_180000 (1).jpg', cv2.IMREAD_COLOR)
image_rgb = cv2.cvtColor(image_rgb, cv2.COLOR_BGR2RGB)
image_gray = rgb_grayscale(image_rgb)
plt.figure(figsize=(10, 5))

plt.subplot(1, 2, 1)
plt.imshow(image_rgb)
plt.title('Imagen RGB')
plt.axis('off')

plt.subplot(1, 2, 2)
plt.imshow(image_gray, cmap='gray')
plt.title('Imagen en escala de grises')
plt.axis('off')

plt.show()

```

Anexo 7

Código del Resaltado HSV

```

def hsv_areas(image):
    hsv_image = cv2.cvtColor(image, cv2.COLOR_BGR2HSV)
    lower_yellow = np.array([20, 100, 100])
    upper_yellow = np.array([30, 255, 255])
    lower_brown = np.array([10, 100, 20])
    upper_brown = np.array([20, 255, 200])
    lower_red1 = np.array([0, 100, 100])
    upper_red1 = np.array([10, 255, 255])
    lower_red2 = np.array([160, 100, 100])
    upper_red2 = np.array([180, 255, 255])
    yellow_mask = cv2.inRange(hsv_image, lower_yellow, upper_yellow)
    brown_mask = cv2.inRange(hsv_image, lower_brown, upper_brown)
    red_mask1 = cv2.inRange(hsv_image, lower_red1, upper_red1)
    red_mask2 = cv2.inRange(hsv_image, lower_red2, upper_red2)
    red_mask = cv2.bitwise_or(red_mask1, red_mask2)
    combined_mask = cv2.bitwise_or(cv2.bitwise_or(yellow_mask, brown_mask), red_mask)
    gray_image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
    highlighted_image = cv2.bitwise_and(gray_image, gray_image, mask=combined_mask)
    result_image = gray_image.copy()
    result_image[combined_mask > 0] = 255
    return result_image, yellow_mask, brown_mask, red_mask

image_paths = [
    '/content/drive/MyDrive/TESIS/DataPlantaVillage/Corn (maize) PucciniaSorghii/20231217_180000 (1).jpg',
    '/content/drive/MyDrive/TESIS/DataPlantaVillage/Corn (maize) SpiroplasmaKunkeli/Polish_20240322_232012354.jpg',
    '/content/drive/MyDrive/TESIS/DataPlantaVillage/Sana/IMG_20230411_060503.jpg'
]
images = [cv2.imread(image_path) for image_path in image_paths]

plt.figure(figsize=(20, 15))

```

Anexo 8

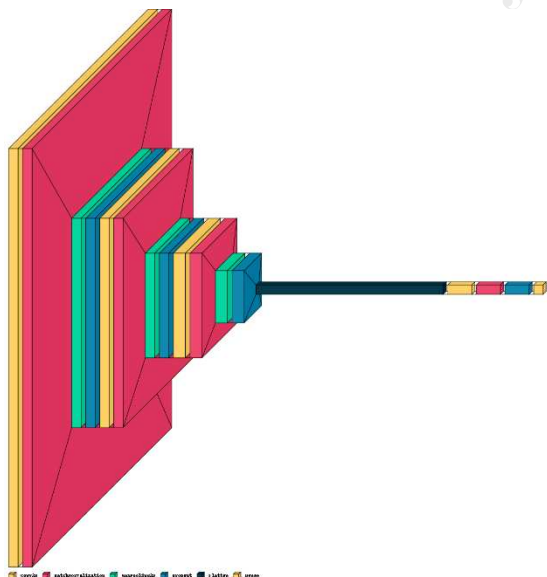
Código del Modelo Implementado

```
# Definir el modelo de la red convolucional
model = models.Sequential()
# Primera capa de convolución
model.add(layers.Conv2D(32, (3, 3), padding='same', activation='relu', input_shape=(224, 224, 1)))
model.add(layers.BatchNormalization())
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Dropout(0.25))
# Segunda capa de convolución
model.add(layers.Conv2D(64, (3, 3), padding='same', activation='relu', kernel_regularizer=regularizers.l2(0.001)))
model.add(layers.BatchNormalization())
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Dropout(0.5))
# Tercera capa de convolución
model.add(layers.Conv2D(256, (3, 3), padding='same', activation='relu', kernel_regularizer=regularizers.l2(0.001)))
model.add(layers.BatchNormalization())
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Dropout(0.25))
# Aplanar y añadir capas totalmente conectadas
model.add(layers.Flatten())
model.add(layers.Dense(512, activation='relu'))
model.add(layers.BatchNormalization())
model.add(layers.Dropout(0.5))
model.add(layers.Dense(3, activation='softmax'))
# Compilar el modelo
model.compile(optimizer=Adam(learning_rate=0.001),
              loss='categorical_crossentropy',
              metrics=['accuracy'])

# Definir callback de EarlyStopping
early_stopping = EarlyStopping(monitor='val_loss', patience=5, restore_best_weights=True)
# Entrenar el modelo con EarlyStopping
modelo = model.fit(train_gen, epochs=50, verbose=1, validation_data=val_gen, callbacks=[early_stopping])
```

Anexo 9

Estructura del Modelo Implementado



Anexo 10

Código de la Evaluación del Rendimiento del Modelo

```

y_pred_probs = modelo_final.predict(test_gen)

# Obtener las clases predichas
y_pred_classes = y_pred_probs.argmax(axis=-1)

# Obtener las clases verdaderas
y_true = test_gen.classes

from sklearn.metrics import accuracy_score, classification_report
# Calcular la precisión global
accuracy = accuracy_score(y_true, y_pred_classes)
print(f'Accuracy global: {accuracy:.2f}')

# Mostrar el accuracy por clases
class_report = classification_report(y_true, y_pred_classes)
print('Accuracy por clases:\n', class_report)

```

Anexo 11

Código de la Matriz de Confusión

```

# Obtener los nombres de las clases (sustituye 'class_names' con los nombres reales)
class_names = ["Puccinia", "Sano", "Spiroplasma"] # Reemplaza con tus nombres de clase reales

# Obtener las predicciones del modelo
y_pred_probabilities = modelo_final.predict(test_gen)

# Obtener las etiquetas predichas (clases)
y_pred = np.argmax(y_pred_probabilities, axis=1)

# Obtener las etiquetas reales
y_true = test_gen.classes

# Calcular y visualizar la matriz de confusión
cm = confusion_matrix(y_true, y_pred)

# Visualizar la matriz de confusión utilizando seaborn y matplotlib
plt.figure(figsize=(8, 6))
sns.heatmap(cm, annot=True, fmt="d", cmap="Blues", cbar=False, xticklabels=class_names, yticklabels=class_names)
plt.xlabel("Predicciones")
plt.ylabel("Clases")
plt.title("Matriz de Confusión")
plt.show()

```

Anexo 12

Código de la Conversión al Formato TFLite

```

# Convertir a TensorFlow Lite
# Crea un convertidor TFLite
converter = tf.lite.TFLiteConverter.from_keras_model(modelo_final)

# Opcionalmente, configura algunas opciones de conversión
# Ejemplo: activa la cuantización (quantization) para reducir el tamaño del modelo
converter.optimizations = [tf.lite.Optimize.DEFAULT]

# Convierte el modelo a TFLite
modelo_tflite = converter.convert()

# Guardar el modelo TFLite en un archivo
nombre_archivo_tflite = '/content/drive/MyDrive/TESIS/TresModelos/mobilenetv2_tflite.tflite'

with open(nombre_archivo_tflite, 'wb') as archivo_tflite:
    archivo_tflite.write(modelo_tflite)

```

Anexo 13

Código de Inicio de Sesión de la Aplicación Móvil

```

24 public class InicioActivity extends AppCompatActivity {
25     EditText email, password;
26     Button ingresar;
27     private FirebaseAuth mAuth;
28
29     @Override
30     protected void onCreate(Bundle savedInstanceState) {
31         super.onCreate(savedInstanceState);
32         setContentView(R.layout.activity_inicio);
33
34         mAuth = FirebaseAuth.getInstance();
35         email=findViewById(R.id.txtEmail);
36         password=findViewById(R.id.txtContra);
37         ingresar=findViewById(R.id.btnIniciar);
38         ingresar.setOnClickListener(new View.OnClickListener() {
39             @Override
40             public void onClick(View v) {
41                 if(email.getText().toString().isEmpty() || password.getText().toString().isEmpty()){
42                     toastE( msgE "Revise su Correo Electronico o Contraseña !!");
43                 }else
44                 {
45                     Inicio();
46                 }
47             }
48         });
49     }
50
51     @Override
52     public void onStart() {
53         super.onStart();
54         // Check if user is signed in (non-null) and update UI accordingly.
55         FirebaseUser currentUser = mAuth.getCurrentUser();
56         if(currentUser != null){
57             Intent i=new Intent( packageContext this, DeteccionActivity.class);
58             startActivity(new Intent[]{i});
59             finish();
60             //currentUser.reload();
61         }
62     }
63     private void limpiar()
64     {
65         email.setText("");
66         password.setText("");
67     }
68     public void ReestablecerContra(View view){
69         Intent i=new Intent( packageContext this, RecuperarCActivity.class);
70         startActivity(new Intent[]{i});
71         //finish();
72     }

```



```

73     public void Inicio()
74     {
75         mAuth.signInWithEmailAndPassword(email.getText().toString(), password.getText().toString())
76             .addOnCompleteListener( activity: this, new OnCompleteListener<AuthResult>() {
77                 @Override
78                 public void onComplete(@NonNull Task<AuthResult> task) {
79                     if (task.isSuccessful()) {
80                         // Sign in success, update UI with the signed-in user's information
81                         //Log.d(TAG, "signInWithEmail:success");
82                         FirebaseUser user = mAuth.getCurrentUser();
83                         // updateUI(user);
84                         //Toast.makeText(getApplicationContext(), "Bienvenido", Toast.LENGTH_SHORT).show();
85                         ToastG( msg: "BIENVENIDO");
86                         limpiar();
87                         // estas dos lineas nos ayuda con la redireccion
88                         Intent i = new Intent(getApplicationContext(), DetecccionActivity.class);
89                         startActivity(new Intent[]{i});
90                         //Finish();
91                     }
92                     } else {
93                         // If sign in fails, display a message to the user.
94                         //Log.w(TAG, "signInWithEmail:failure", task.getException());
95                         toastE( msg: "Revise su Correo Electronico o Contraseña !!");
96                         //Toast.makeText(getApplicationContext(), "Revise su Contraseña o Correo", Toast.LENGTH_SHORT).show();
97                         // updateUI(null);
98                     }
99                 }
100             });
101     }

```

Anexo 14

Código de Registro de la Aplicación Móvil

```

32     public class RegistroActivity extends AppCompatActivity {
33
34         3 usages
35         private FirebaseAuth mAuth;
36         2 usages
37         private FirebaseFirestore fStore;
38         5 usages
39         EditText nombres, apellidos, email, password, password2;
40
41         2 usages
42         Button btnRegistrar;
43         6 usages
44         private ProgressDialog dialogo;
45
46         @Override
47         protected void onCreate(Bundle savedInstanceState) {
48             super.onCreate(savedInstanceState);
49             setContentView(R.layout.activity_registro);
50
51             mAuth = FirebaseAuth.getInstance();
52             fStore = FirebaseFirestore.getInstance();
53
54             nombres= findViewById(R.id.txtNombres);
55             apellidos = findViewById(R.id.txtApellidos);
56             email= findViewById(R.id.txtEmailR);
57             password = findViewById(R.id.txtContraR);
58             password2=findViewById(R.id.txtContraR2);
59             btnRegistrar= findViewById(R.id.btnRegistrar);
60
61             dialogo = new ProgressDialog( context: this);

```

```

62         @Override
63         public void onClick(View v) {
64             String nom=nombres.getText().toString();
65             String ape=apellidos.getText().toString();
66             String esemail=email.getText().toString();
67             String cpassword=password.getText().toString();
68             String c2=password2.getText().toString();
69
70             if(nom.isEmpty() || ape.isEmpty() || c1.isEmpty() || e.isEmpty() || c2.isEmpty()){
71                 ToastE( msg: "Ingrese sus Datos en Todos los Campos");
72             }else if(!VLetras(nom)){
73                 nombres.setError("Nombre Incorrecto");
74                 ToastE( msg: "No Ingrese caracteres especiales ni números");
75             } else if (!VLetras(ape) ) {
76                 apellidos.setError("Apellido Incorrecto");
77                 ToastE( msg: "No Ingrese caracteres especiales ni números");
78             }else if( es_email.isEmpty() || !android.util.Patterns.EMAIL_ADDRESS.matcher(e).matches() ) {
79                 email.setError("Correo Inválido");
80             }else if(!c.equals(c2)){
81                 Toast.makeText(getApplicationContext(), txt: "Las contraseñas son diferentes", Toast.LENGTH_SHORT).show();
82                 password.setError("La contraseña es diferente");
83                 password2.setError("Su contraseña es diferente");
84             }else if(c.length()<=7 || c2.length()<=7){
85                 ToastE( msg: "Contraseña Insegura");
86                 ToastE( msg: "Su contraseña debe contener más de 7 dígitos");
87             }else{
88                 dialogo.setMessage("Registrando ..");
89                 dialogo.setCancelableOnTouchOutside(false);
90                 dialogo.show();
91                 RegistroU();
92             }
93         }

```

```

154 public void RegistroU()
155 {
156     mAuth.createUserWithEmailAndPassword(email.getText().toString(), password.getText().toString())
157     .addOnCompleteListener( activity: this, new OnCompleteListener<AuthResult>() {
158         @Override
159         public void onComplete(@NonNull Task<AuthResult> task) {
160             if (task.isSuccessful()) {
161                 FirebaseUser user= mAuth.getCurrentUser();
162                 String userID= user.getUID();
163                 DocumentReference documentReference = firestore.collection( collectionPath: "Users").document(userID);
164                 // Create a new user with a first and last name
165                 Map<String, Object> dataU= new HashMap<>();
166                 dataU.put("usuario", nombres.getText().toString());
167                 dataU.put("apellidos", apellidos.getText().toString());
168                 dataU.put("email", email.getText().toString());
169                 documentReference.set(dataU).addOnSuccessListener(new OnSuccessListener<Void>() { // aqui me dice error
170                     @Override
171                     public void onSuccess(Void unused) {
172
173                         Log.d( tag: "Registro", msg: "coleccion creada");
174                         dialogo.dismiss();
175                         ToastG( msg: "Registro Exitoso!");
176                         Limpiar();
177                         Intent in = new Intent( packageContext: RegistroActivity.this, InicioActivity.class);
178                         startActivities(new Intent[]{in});
179                         //finish();
180                     }
181                 }).addOnFailureListener(new OnFailureListener() {
182                     @Override
183                     public void onFailure(@NonNull Exception e) {
184                         Log.d( tag: "Registro", msg: "error al crear coleccion");
185                     }
186                 });
187             }
188         }
189     });
190 }

```

Anexo 15

Código del Análisis de las Imágenes de la Aplicación Móvil

```

59 protected void onCreate(Bundle savedInstanceState) {
60     super.onCreate(savedInstanceState);
61     setContentView(R.layout.activity_deteccion);
62
63     mAuth = FirebaseAuth.getInstance();
64     mStorage = FirebaseStorage.getInstance().getReference();
65     getPermission();
66
67     seleccionBtn = findViewById(R.id.seleccionBtn);
68     capturaBtn = findViewById(R.id.capturaBtn);
69     prediccionBtn = findViewById(R.id.prediccionBtn);
70     resultadotxt = findViewById(R.id.resultadotxt);
71     resultadotxt.setVisibility(View.INVISIBLE);
72     imagenView = findViewById(R.id.imageView);
73
74     salirBtn=findViewById(R.id.btnSalir);
75     informacionBtn=findViewById(R.id.btnInformacion);
76     btnPerfil = findViewById(R.id.btnPerfil);
77
78     seleccionBtn.setOnClickListener(new View.OnClickListener() {
79         @Override
80         public void onClick(View view) {
81             Intent intent = new Intent();
82             intent.setAction(Intent.ACTION_GET_CONTENT);
83             intent.setType("image/*");
84             startActivityForResult(intent, requestCode: 10);
85             resultadotxt.setText(" ");
86         }
87     });

```

```

88
89     capturaBtn.setOnClickListener(new View.OnClickListener() {
90         @Override
91         public void onClick(View view) {
92             Intent intent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
93             //startActivityForResult(intent, 12);
94             if (intent.resolveActivity(getPackageManager()) != null) {
95                 startActivityForResult(intent, requestCode: 12);
96             }
97             resultadotxt.setText(" ");
98         }
99     });

```

```

100 predicciónBtn.setOnClickListener(new View.OnClickListener() {
101     @Override
102     public void onClick(View v) {
103
104         if (bitmap == null) {
105             Toast.makeText(getApplicationContext(), "Seleccione una Imagen o Tome una Foto", Toast.LENGTH_SHORT).show();
106         } else {
107
108             try {
109                 Modelo2Tflite model1 = Modelo2Tflite.newInstance(context, DetecccionActivity.this);
110                 // Creates inputs for reference.
111                 TensorBuffer inputFeature1 = TensorBuffer.createFixedSize(new int[]{1, 224, 224, 3}, DataType.FLOAT32);
112                 bitmap = Bitmap.createScaledBitmap(bitmap, dstWidth: 224, dstHeight: 224, filter: true);
113                 ByteBuffer byteBuffer = ByteBuffer.allocateDirect(capacity: 4 * imageSize * imageSize * 3);
114                 byteBuffer.order(ByteOrder.nativeOrder());
115                 int[] values = new int[imageSize * imageSize];
116                 bitmap.getPixels(values, offset: 0, bitmap.getWidth(), x: 0, y: 0, bitmap.getWidth(), bitmap.getHeight());
117
118                 int pixel = 0;
119                 for (int i = 0; i < imageSize; i++) {
120                     for (int j = 0; j < imageSize; j++) {
121                         int val = values[pixel++];
122                         byteBuffer.putFloat(value: ((val >> 16) & 0xFF) * (1.f / 255.f));
123                         byteBuffer.putFloat(value: ((val >> 8) & 0xFF) * (1.f / 255.f));
124                         byteBuffer.putFloat(value: (val & 0xFF) * (1.f / 255.f));
125                     }
126                 }
127                 inputFeature1.loadBuffer(byteBuffer);
128
129                 // Runs model inference and gets result.
130                 Modelo2Tflite.Outputs outputs1 = model1.process(inputFeature1);
131                 TensorBuffer outputFeature1 = outputs1.getOutputFeature0AsTensorBuffer();
132                 float[] conf1 = outputFeature1.getFloatArray();
133                 int pos1 = 0;
134                 float maxConf1 = 0;
135                 for (int i = 0; i < conf1.length; i++) {
136                     if (conf1[i] > maxConf1) {
137                         maxConf1 = conf1[i];
138                         pos1 = i;
139                     }
140                 }
141                 if (pos1 == 0) {
142                     // malz
143                     //ModeloFinalNuevoTflite model = ModeloFinalNuevoTflite.newInstance(DetecccionActivity.this);
144                     Mobilenetv2Tflite model = Mobilenetv2Tflite.newInstance(context, DetecccionActivity.this);
145                     // Creates inputs for reference.
146                     TensorBuffer inputFeature0 = TensorBuffer.createFixedSize(new int[]{1, 224, 224, 3}, DataType.FLOAT32);
147                     bitmap = Bitmap.createScaledBitmap(bitmap, dstWidth: 224, dstHeight: 224, filter: true);
148                     inputFeature0.loadBuffer(byteBuffer);
149
150                     // Runs model inference and gets result.
151                     Mobilenetv2Tflite.Outputs outputs = model.process(inputFeature0);
152                     TensorBuffer outputFeature0 = outputs.getOutputFeature0AsTensorBuffer();
153
154                     float[] conf = outputFeature0.getFloatArray();
155                     int pos = 0;
156                     float maxConf = 0;
157                     String clase = "";
158
159                     for (int i = 0; i < conf.length; i++) {
160                         if (conf[i] > maxConf) {
161                             maxConf = conf[i];
162                             pos = i;
163                         }
164                     }
165
166                     if (pos == 0) {
167                         clase = "Puccinia Sorghi";
168                     } else if (pos == 1) {
169                         clase = "Hoja Sana";
170                     } else if (pos == 2) {
171                         clase = "Spiroplasma Kunkeli";
172                     } else {
173                         clase = "Planta Desconocida";
174                     }
175
176                     resultadoetxt.setVisibility(View.VISIBLE);
177                     resultadoetxt.setText(clase);
178
179                     //resultadoetxt.setText("clase: "+outputFeature0.getFloatArray()[0]); // por este fallaba
180                     // Releases model resources if no longer used.
181                     model.close();
182                 } else if (pos1 == 1) {
183                     // desconocido
184                     resultadoetxt.setVisibility(View.VISIBLE);
185                     resultadoetxt.setText("Planta Desconocida");
186                 }
187             } catch (Exception e) {
188                 e.printStackTrace();
189             }
190             model1.close();

```

```

@Override
public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {
    if(requestCode==11){
        if(grantResults.length>0){
            if(grantResults[0]!= PackageManager.PERMISSION_GRANTED){
                this.getPermission();
            }
        }
    }
    super.onRequestPermissionsResult(requestCode, permissions, grantResults);
}

@Override
protected void onActivityResult(int requestCode, int resultCode, @androidx.annotation.Nullable Intent data) {
    // esta funcion es para abrir galeria y cargar la imagen
    if(requestCode==10)
    {
        if(data!=null)
        {
            Uri uri=data.getData();
            // agregue esto para subir la imagen
            StorageReference filePath= mStorage.child( pathString: "imagenes").child(Uri.getLastPathSegment());
            filePath.putFile(uri).addOnSuccessListener(new OnSuccessListener<UploadTask.TaskSnapshot>() {
                @Override
                public void onSuccess(UploadTask.TaskSnapshot taskSnapshot) {
                    //Toast.makeText(getApplicationContext(),"Imagen cargada",Toast.LENGTH_SHORT).show();
                }
            });
        }
    }
}

```

Anexo 16

Código del Preprocesamiento de las Imágenes en la Aplicación Móvil

```

private Bitmap processImage(Bitmap inputBitmap) {
    // Carga la imagen en formato Mat de OpenCV
    Mat image = new Mat();
    org.opencv.android.Utils.bitmapToMat(inputBitmap, image);

    // Convierte la imagen a HSV
    Mat hsvImage = new Mat();
    Imgproc.cvtColor(image, hsvImage, Imgproc.COLOR_BGR2HSV);

    // Define los rangos de colores
    Scalar lowerYellow = new Scalar(20, 100, 100);
    Scalar upperYellow = new Scalar(30, 255, 255);
    Scalar lowerBrown = new Scalar(10, 100, 20);
    Scalar upperBrown = new Scalar(20, 255, 200);
    Scalar lowerRed1 = new Scalar(0, 100, 100);
    Scalar upperRed1 = new Scalar(10, 255, 255);
    Scalar lowerRed2 = new Scalar(160, 100, 100);
    Scalar upperRed2 = new Scalar(180, 255, 255);

    // Crear máscaras para cada color
    Mat yellowMask = new Mat();
    Mat brownMask = new Mat();
    Mat redMask1 = new Mat();
    Mat redMask2 = new Mat();

    Core.inRange(hsvImage, lowerYellow, upperYellow, yellowMask);
    Core.inRange(hsvImage, lowerBrown, upperBrown, brownMask);
    Core.inRange(hsvImage, lowerRed1, upperRed1, redMask1);
    Core.inRange(hsvImage, lowerRed2, upperRed2, redMask2);
}

```



```

// Combinar máscaras rojas
Mat redMask = new Mat();
Core.bitwise_or(redMask1, redMask2, redMask);

// Combinar todas las máscaras
Mat combinedMask = new Mat();
Core.bitwise_or(yellowMask, brownMask, combinedMask);
Core.bitwise_or(combinedMask, redMask, combinedMask);

// Convertir la imagen a escala de grises
Mat grayImage = new Mat();
Imgproc.cvtColor(image, grayImage, Imgproc.COLOR_BGR2GRAY);

// Resaltar áreas detectadas
Mat highlightedImage = new Mat();
Core.bitwise_and(grayImage, grayImage, highlightedImage, combinedMask);

// Crear la imagen de resultado
Mat resultImage = grayImage.clone();
resultImage.setTo(new Scalar(0, 255), combinedMask);

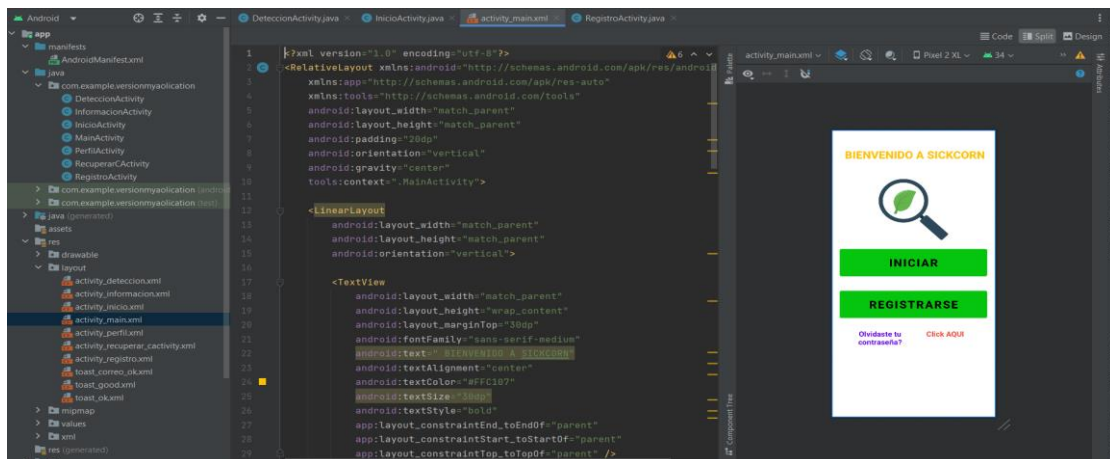
// Convertir el resultado a Bitmap
Bitmap resultBitmap = Bitmap.createBitmap(resultImage.cols(), resultImage.rows(), Bitmap.Config.ARGB_8888);
org.opencv.android.Utils.matToBitmap(resultImage, resultBitmap);

return resultBitmap;
}

```

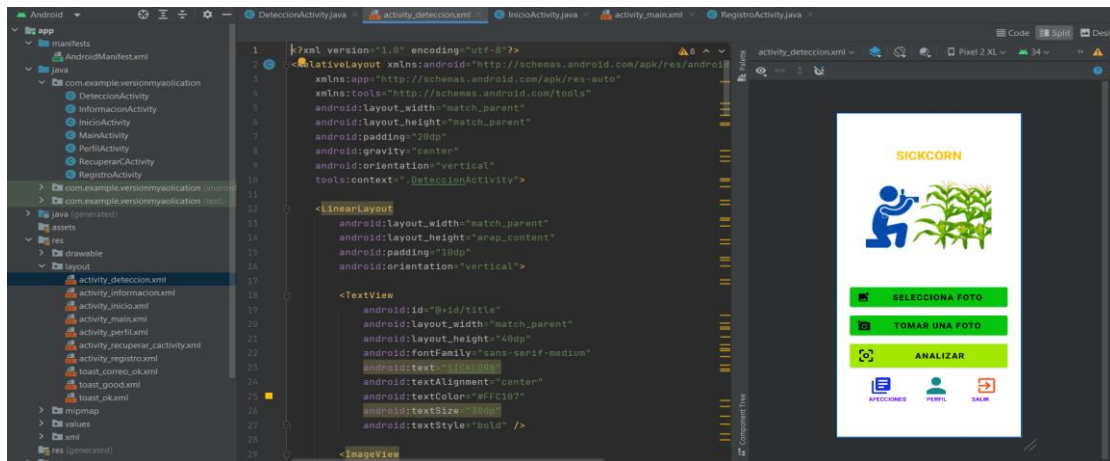
Anexo 17

Ventana Principal de la Aplicación Móvil



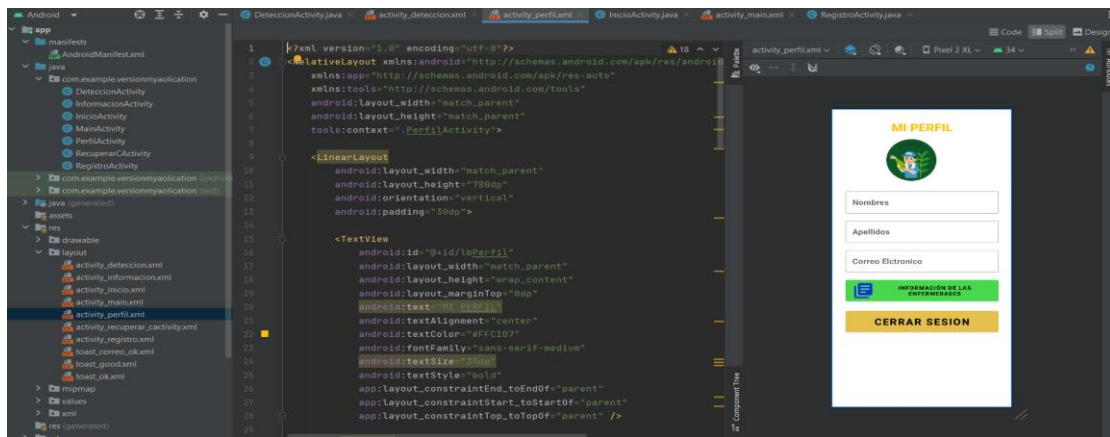
Anexo 18

Ventana de Detección de la Aplicación Móvil



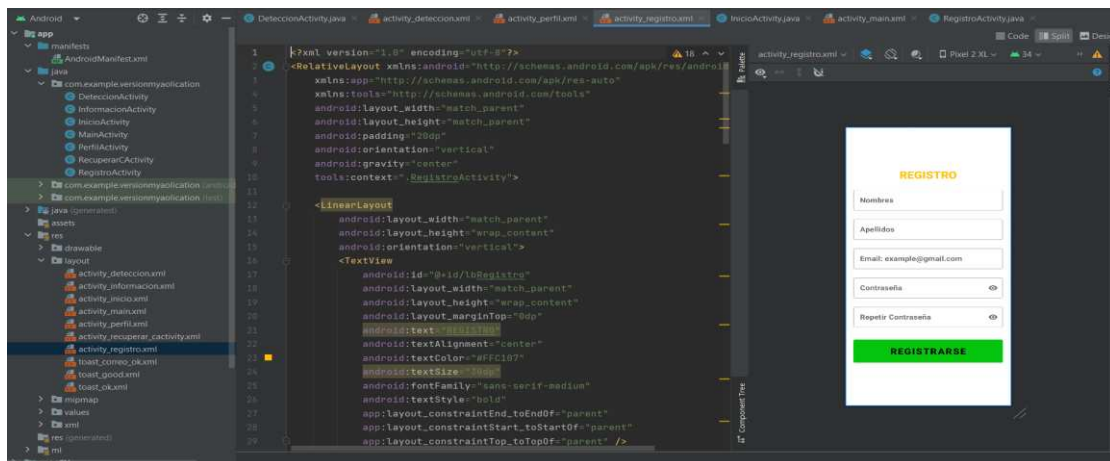
Anexo 19

Ventana del Perfil de Usuario



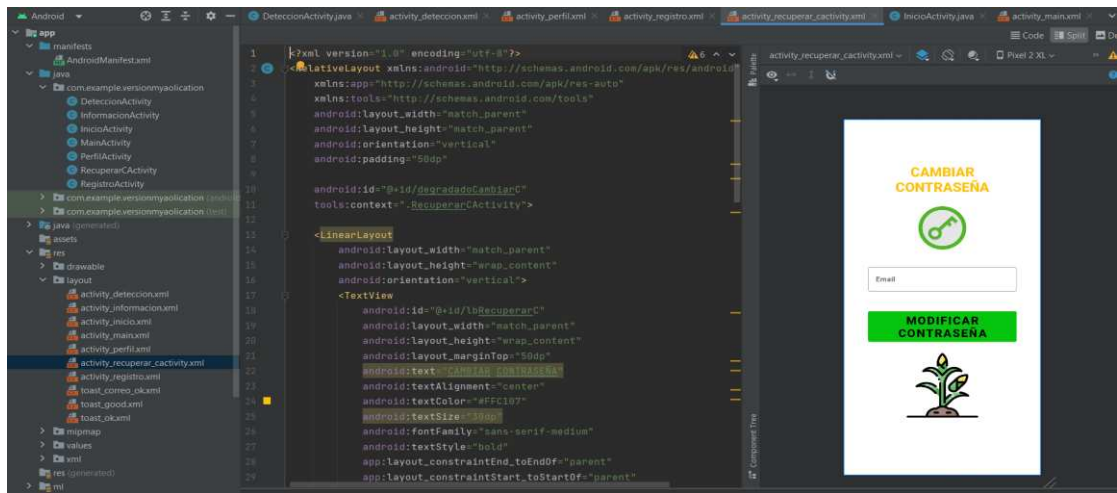
Anexo 20

Ventana Registro



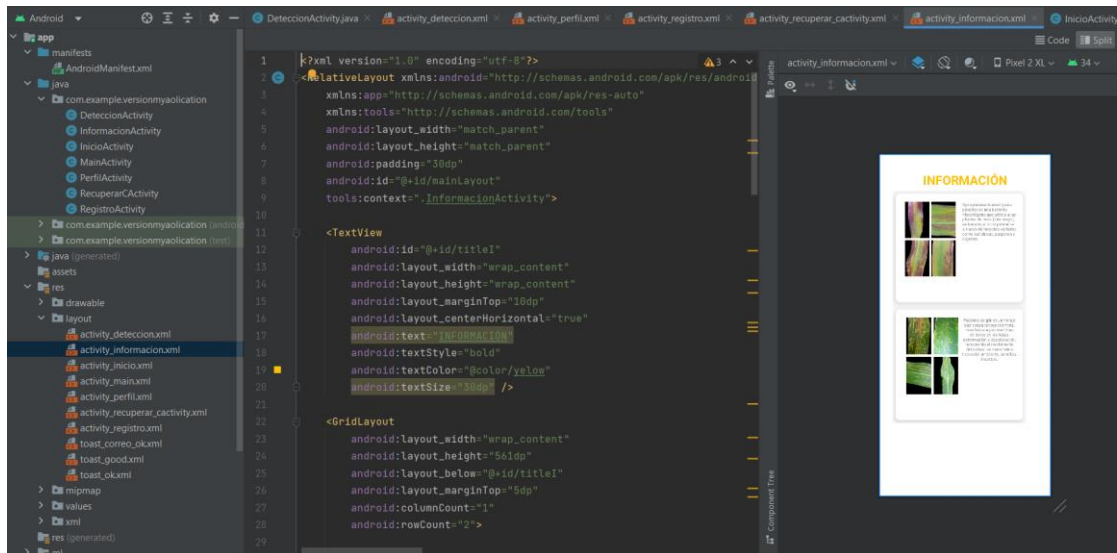
Anexo 21

Ventana Recuperar Contraseña



Anexo 22

Ventana Información





FORMATO DE AUTORIZACIÓN DE PUBLICACIÓN EN EL REPOSITORIO INSTITUCIONAL DE LA UNAJMA

Formulario de autorización para la publicación electrónica en la página web del Repositorio Institucional de la Universidad Nacional de José María Arguedas, de conformidad con la Ley N° 30035 Ley que regula el Repositorio Nacional Digital de Ciencia, Tecnología e Innovación de Acceso Abierto y Resolución de Comisión Organizadora N° 0203-2021 que aprueba el Reglamento que regula el Repositorio Institucional Digital de Ciencia, Tecnología e Innovación de Acceso Abierto de la Universidad Nacional de José María Arguedas.

1. DATOS DEL AUTOR:

Nombres y Apellidos: Jhackelyn Esthefany Leon Hurtado

DNI / Carnet de Extranjería: 71242212

Pasaporte N°:

Correo Institucional: 1007520191 @unajma.edu.pe

Teléfono/Celular: 924626909

Correo Electrónico: leonhurtadojhackelyn@gmail.com

2. GRADO, TÍTULO O ESPECIALIZACIÓN (Marque con una X)

| Bachiller | Título Profesional | Magíster | Doctor | Segunda Especialidad |
|-----------|--------------------|----------|--------|----------------------|
| | X | | | |

3. TIPO DE PUBLICACIÓN (Marque con una X)

| Tesis | Trabajo Académico | Trabajo de investigación | Trabajo de Suficiencia profesional |
|-------|-------------------|--------------------------|------------------------------------|
| X | | | |



4. DATOS DE LA PUBLICACION:

Título: Aplicación móvil “SickCorn” en la detección de Puccinia sorghi (roya común) y Spiroplasma kunkeli (Puka poncho) en las plantas de maíz, Distrito de Talavera.

Fecha de Publicación:

5. DATOS DEL ASESOR:

Nombres y Apellidos: Enrique Edgardo Condor Tinoco

DNI/Pasaporte/Cédula: 40998311

ORCID:

Firma

DOCUMENTO DE IDENTIDAD: 41618732

NOMBRES Y APELLIDOS: Jhackelyn Esthefany Leon Hurtado



LICENCIAS

Licencia Estándar

Bajo los siguientes términos, autorizo el depósito de mi trabajo de investigación en el Repositorio Institucional Digital de la UNAJMA.

Con la autorización de depósito de mi trabajo de investigación otorgo a la Universidad Nacional de José María Arguedas, una Licencia No Exclusiva para reproducir, distribuir, comunicar al público, transformar (únicamente mediante su traducción a otros idiomas) y poner a disposición del público mi trabajo de investigación (incluido el resumen), en formato físico o digital en cualquier medio, conocido o por conocerse, a través de los diversos servicios provistos por la Universidad, creados o por crearse, tales como el Repositorio Institucional Digital de la UNAJMA, Colección de Tesis, entre otros, en el Perú y en el extranjero, por el tiempo y veces que considere necesarias y libre de remuneraciones.

En virtud de dicha Licencia, la Universidad Nacional de José María Arguedas, podrá reproducir mi trabajo de investigación en cualquier tipo de soporte y en más de un ejemplar, sin modificar su contenido, solo con los propósitos de seguridad, respaldo y preservación.

Declaro que el trabajo de investigación es una creación de mi autoría y exclusiva titularidad, o coautoría con titularidad compartida, y me encuentro facultado a conceder la presente licencia y, asimismo, garantizo que dicho trabajo de investigación no infringe derechos de autor de terceras personas.

La Universidad Nacional de José María Arguedas consignará el nombre del/los autores/es del trabajo de investigación, y no le hará ninguna modificación más de la permitida en la presente licencia.

Autorizo el depósito (marque con una X).

SI autorizo que se deposite inmediatamente.

No autorizo.

Fecha: 26/09/2024

.....
Firma



Documento de identidad : 71242212

Nombres y apellidos: Jhackelyn Esthefany Leon Hurtado