

UNIVERSIDAD PERUANA UNIÓN
FACULTAD DE INGENIERÍA Y ARQUITECTURA
Escuela Profesional de Ingeniería de Sistemas



Una Institución Adventista

**Aplicación Web para Clasificar los Granos de la Quinoa Según la
Norma Técnica Peruana Aplicando Machine Learning**

Tesis para obtener el Título Profesional de Ingeniero de Sistemas

Por:

Paul Anderson Quispe Valero

Asesor:

Ing. Angel Rosendo Condori Coaquira

Juliaca, diciembre de 2020

DECLARACIÓN JURADA DE AUTORÍA DEL INFORME DE TESIS

Ing. Angel Rosendo Condori Coaquira, de la Facultad de Ingeniería y Arquitectura, Escuela Profesional de Ingeniería de Sistemas, de la Universidad Peruana Unión.

DECLARO:

Que el presente informe de investigación titulado: **“APLICACIÓN WEB PARA CLASIFICAR LOS GRANOS DE LA QUINUA SEGÚN LA NORMA TÉCNICA PERUANA APLICANDO MACHINE LEARNING”** constituye la memoria que presenta el(la) Bachiller Paul Anderson Quispe Valero para obtener el título de Profesional de Ingeniero de Sistemas, cuya tesis ha sido realizada en la Universidad Peruana Unión bajo mi dirección.

Las opiniones y declaraciones en este informe son de entera responsabilidad del autor, sin comprometer a la institución.

Y estando de acuerdo, firmo la presente declaración en Juliaca, a los 03 días del mes de diciembre del año 2020



Ing. Angel Rosendo Condori
Coaquira



ACTA DE SUSTENTACIÓN DE TESIS

En Puno, Juliaca, Villa Chullunquiari, a 23 día(s) del mes de diciembre del año 2020, siendo las 17:00 horas, se reunieron en el Salón de Grados y Títulos de la Universidad Peruana Unión, Filial Juliaca, bajo la dirección del Señor Presidente del jurado: Mtro. Lenin Henry Lenturion Julca, el secretario: Mg. Abel Angel Sullon Macalupa y los demás miembros: Ing. David Mamari Pari - Ing. Eder Gutierrez Quispe y el asesor: Ing. Angel Rosendo Gordon Loagira con el propósito de administrar el acto académico de sustentación de la tesis titulada: Aplicación Web para Clasificar los Granos de la Quinua según la Norma Técnica Peruana aplicando Machine Learning de el(los)/a(las) bachiller(es): a) Paul Anderson Quispe Valero b) _____ conducente a la obtención del título profesional de Ingeniero de Sistemas (Nombre del Título Profesional)

con mención en: _____

El Presidente inició el acto académico de sustentación invitando al (los)/a(la)(las) candidato(a)s hacer uso del tiempo determinado para su exposición. Concluida la exposición, el Presidente invitó a los demás miembros del jurado a efectuar las preguntas, y aclaraciones pertinentes, las cuales fueron absueltas por el(los)/a(las) candidato(a)s. Luego, se produjo un receso para las deliberaciones y la emisión del dictamen del jurado.

Posteriormente, el jurado procedió a dejar constancia escrita sobre la evaluación en la presente acta, con el dictamen siguiente:

Candidato (a): Paul Anderson Quispe Valero

CALIFICACIÓN	ESCALAS			Mérito
	Vigesimal	Literal	Cualitativa	
<u>APROBADO</u>	<u>18</u>	<u>A-</u>	<u>Muy Bueno</u>	<u>Sobresaliente</u>

Candidato (b): _____

CALIFICACIÓN	ESCALAS			Mérito
	Vigesimal	Literal	Cualitativa	

(*) Ver parte posterior

Finalmente, el Presidente del jurado invitó al(los)/a(la)(las) candidato(a)s a ponerse de pie, para recibir la evaluación final y concluir el acto académico de sustentación procediéndose a registrar las firmas respectivas.

Presidente

Asesor

Candidato/a (a)

Secretario

Miembro

Candidato/a (b)

DEDICATORIA

Primero que nada, gracias a Dios por darme la vida y estar siempre conmigo. Segundo, de manera muy especial, para mi familia, la investigación y todo lo que puedo hacer se atribuye a su apoyo incondicional, la inculcación de virtudes y valores; para mis hermanas y amigos cercanos, que me ha brindado su comprensión.

Toda esta investigación fue posible gracias a ellos.

AGRADECIMIENTOS

A Dios quien me sustentó y me cuidó todo este tiempo, tanto en la salud, espiritual y material.

Quien me enseñó que sobre todo en lo espiritual, lograré mis metas.

A la Universidad Peruana Unión por habernos inculcado principios y valores cristianos durante la permanencia en la Universidad.

A los docentes de la Escuela de Profesional de Ingeniería de Sistemas de la Universidad Peruana Unión quienes con paciencia y respeto impartieron sus conocimientos.

INDICE GENERAL

DEDICATORIA	IV
AGRADECIMIENTOS	V
ÍNDICE DE FIGURAS	VIII
ÍNDICE DE TABLAS	IX
RESUMEN	X
CAPÍTULO I. EL PROBLEMA	11
1.1. DESCRIPCIÓN DEL PROBLEMA	11
1.2. JUSTIFICACIÓN DE LA INVESTIGACIÓN	15
1.3. PRESUNCIÓN FILOSÓFICA	17
1.4. OBJETIVOS	18
1.5. OBJETIVO GENERAL	18
1.6. OBJETIVOS ESPECÍFICOS	18
CAPÍTULO II. MARCO TEÓRICO	19
1.7. REVISIÓN DE LA LITERATURA	19
1.8. MARCO TEÓRICO	22
1.9. QUINUA	22
1.10. CLASIFICACIÓN DE QUINUA	22
1.11. NORMA TECNICA PERUANA	23
2.1.2. <i>INTELIGENCIA ARTIFICIAL</i>	26
2.1.3. <i>INCEPTION</i>	30
2.1.4. <i>PROCESO AGIL UNIFICADO</i>	32
2.1.5. <i>METODOLOGÍA FUNDAMENTAL PARA LA CIENCIA DE DATOS</i>	33
CAPÍTULO III. MATERIALES Y MÉTODOS	38
1.12. METODOLOGÍA DE LA INVESTIGACIÓN	38
3.1. ARQUITECTURA DE LA SOLUCIÓN	39
1.13. HERRAMIENTAS	41
3.1.1. SOFTWARE	41
1.14. HARDWARE	41
1.15. MÉTODOS	42
3.1.2. TRANSFER LEARNING	42
1.16. METODOLOGÍA FUNDAMENTAL PARA LA CIENCIA DE DATOS	44
3.1.3. COMPRENSIÓN DEL NEGOCIO	44
3.1.4. REQUISITOS DE DATOS	44
3.1.5. RECOPIACIÓN DE DATOS	45
3.1.6. COMPRENSIÓN DE DATOS	45
3.1.7. PREPARACIÓN DE DATOS	46
3.1.8. MODELADO	47
3.1.9. EVALUACIÓN	56
3.1.10. IMPLEMENTACIÓN	56
3.2. METODOLOGÍA AUP PARA EL DESARROLLO DE LA APLICACIÓN WEB	56
3.3. FASE I: INICIACIÓN	56

3.3.1.	REQUERIMIENTOS.....	57
3.3.1.1.	REQUERIMIENTOS FUNCIONALES.....	57
3.3.1.2.	REQUERIMIENTOS NO FUNCIONALES.....	58
3.4.	FASE II: ELABORACIÓN.....	59
3.4.1.	ACTORES.....	59
3.4.2.	DIAGRAMA DE CASOS DE USO.....	60
3.4.2.1.	ESPECIFICACIÓN DE CASOS DE USO.....	60
3.4.3.	ARQUITECTURA DE LA APLICACIÓN WEB PROPUESTA.....	65
3.5.	FASE III: CONSTRUCCIÓN.....	67
3.5.1.	MODELO DE BASE DE DATOS.....	67
3.5.2.	DIAGRAMA DE COMPONENTES.....	68
3.6.	APLICACIÓN WEB.....	69
3.6.1.	INTERFAZ LOGIN.....	69
3.6.2.	INTERFAZ MENÚ.....	70
3.6.3.	INTEFAZ GRAIN.....	71
CAPÍTULO IV. RESULTADOS Y DISCUSIÓN.....		74
4.1.	RESULTADO N° 1: FORMAR UN BANCO DE IMÁGENES DE MUESTRAS DE QUINUA.....	74
4.2.	RESULTADO N° 2: IDENTIFICAR CARACTERÍSTICAS DE LA QUINUA Y SUS IMPUREZAS PARA LA CLASIFICACIÓN.....	75
4.3.	RESULTADO N° 3: MODELAR, ENTRENAR Y VALIDAR EL MODELO COMPUTACIONAL.....	76
4.4.	RESULTADO N° 4: IMPLEMENTAR EL MODELO COMPUTACIONAL EN UNA APLICACIÓN WEB.....	79
CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES.....		81
5.1.	CONCLUSIONES.....	81
5.2.	RECOMENDACIONES.....	82
REFERENCIAS BIBLIOGRAFICAS.....		83
ANEXOS.....		84

ÍNDICE DE FIGURAS

Figura 1. Proceso de muestro de los granos de quinua.	11
Figura 2. Producción de los granos de quinua 2002-2013.	12
Figura 3. Exportación de granos de quinua 2002-2013.	13
Figura 4. Producción de grano de quinua a nivel nacional (Perú).	13
Figura 5. Tamaño de los granos de quinua.	24
Figura 6. Inteligencia artificial.	26
Figura 7. Ramas de la inteligencia artificial.	27
Figura 8. Distinción de spam mediante machine learning.	28
Figura 9. Procesamiento de imágenes.	29
Figura 10. Modelo inception.	30
Figura 11. Módulo de inception.	31
Figura 12. Ciclo de vida de AUP.	32
Figura 13. Fases de la metodología fundamental para la ciencia de datos.	33
Figura 14. Arquitectura solución.	39
Figura 15. Pre-trained generic CNN.	42
Figura 16. Transfer learning.	43
Figura 17. Organización de adquisición de imágenes.	45
Figura 18. reorganización de DATA_SET.	47
Figura 19. Estructura de carpetas de IA.	48
Figura 20. Entrenamiento de inception.	55
Figura 21. Actores de AGRAIN.	59
Figura 22. Diagrama de Usuario y Administrador.	60
Figura 23. Arquitectura MVC.	65
Figura 24. Carpetas de la aplicación web.	66
Figura 25. Modelo de la base de datos.	67
Figura 26. Componentes de AGRAIN.	68
Figura 27. Login de la aplicación AGRAIN.	69
Figura 28. Menú del admin (administrador).	70
Figura 29. Menú principal de user(usuario).	70
Figura 30. CRUD grain (granos andinos).	71
Figura 31. CRUD type grains (tipos de granos).	71
Figura 32. CRUD sample grain (muestra del grano).	72
Figura 33. CRUD analisis grain (análisis de granos).	72
Figura 34. Reporte del análisis de la muestra de los granos de quinua.	73
Figura 35. Organización de folders.	74
Figura 36. Grano de quinua.	75
Figura 37. Imagen digital procesada de grano de quinua.	75
Figura 38. Organización de carpetas para inception.	76
Figura 39. Modelo de la implementación completa.	79
Figura 40. Análisis detallado de quinua por grano.	79

ÍNDICE DE TABLAS

Tabla 1. Clasificación de granos de quinua según el grado.	25
Tabla 2. Clasificación de modelos CNN.	31
Tabla 3. Descripción del software.	41
Tabla 4. Especificación de hardware.	42
Tabla 5. Matriz de confusión.	56
Tabla 6. Condiciones funcionales de la aplicación web.	57
Tabla 7. Condiciones no funcionales de la aplicación web.	58
Tabla 8. Definición de actores.	59
Tabla 9. Especificación de caso de uso para la clasificación de granos quinua.	61
Tabla 10. Especificación de caso de uso para el registro de proveedor.	62
Tabla 11. Especificación de caso de uso para el registro de la muestra.	63
Tabla 12. Especificación de caso de uso para generar el reporte del análisis.	64
Tabla 13. Resultado de entrenamiento de Inception.	77
Tabla 14. Etiquetas estándar de los granos andinos.	80
Tabla 15. Tamaño estándar de granos andinos.	80

RESUMEN

El procesamiento de granos de quinua es complejo además que el análisis físico no genera información para una posterior retro alimentación, y es necesario tener un clasificador de granos, que permita clasificar grano por grano reduciendo el tiempo del análisis y es fundamental tener información acerca del grano de quinua y su clasificación, con la relevancia que está ganando machine learning y uno de sus sub campos deep learning, es posible entrenar modelos de redes neuronales convolucionales que permiten analizar imágenes digitales clasificando los granos de quinua.

La presente investigación, utiliza el modelo Inception perteneciente a las redes neuronales convolucionales para analizar imágenes de granos de quinua para poder identificar parámetros basados en la NTP 206.56:2014, brindando un análisis en un corto tiempo creando un banco de información sobre el grano de quinua y su procedencia. Se utiliza la metodología para la ciencia de datos para la implementación de Inception y la abstracción de datos. Para el desarrollo web se emplea la metodología de proceso ágil unificado para analizar, desarrollar e implementar el modelo Inception en la aplicación web.

El modelo Inception, clasifica los granos de quinua, mediante una imagen de entrada diferenciando cada grano según sea clasificado usando 7 etiquetas(labels) y está integrado a la aplicación web, como resultado se obtuvo una interfaz de usuario para el uso del modelo computacional y la generación de reportes según la muestra por cada grano de quinua exportado en formato pdf, adicionalmente generando un banco de información para futuros usos.

Palabras Clave: Inception, Deep Learning, Redes Neuronales Convolucionales, Proceso Ágil Unificado

CAPÍTULO I. EL PROBLEMA

1.1. DESCRIPCIÓN DEL PROBLEMA

El proceso de clasificación del grano de quinua y la norma estimula que el grano de quinua, destinado para la comercialización o consumo humano, tiene el estándar de clasificación. Este proceso como toda actividad tiene un tiempo determinado elevado, y de acuerdo al tamaño del grano de quinua puede ocasionar errores de análisis en cuanto a actividad humana (CECAGRO, 2016).



Figura 1. Proceso de muestreo de los granos de quinua.

Fuente: (CECAGRO, 2016).

El proceso de muestreo como se muestra en la figura 1 tiene una duración de 4 a 5 horas de análisis por muestras de 100 gr, donde las impurezas en los granos de quinua son muy frecuentes, por lo que es necesario hacer un análisis de las impurezas que contiene un grupo amplio de granos de quinua, así como identificar los tipos de impurezas, además del estado físico del grano de quinua.

Las operaciones para la obtención del grano de quinua se detallan en limpieza, selección y clasificación además de desaponificado, secado y envasado y almacén, manifestando de importancia la selección y clasificación del grano de quinua (Kamiya Matsuoka, Rida Gamero, & Torres Artica, 2013).

El volumen producido de granos de quinua creció de una forma significativa en el sector agropecuario alcanzando las 52 toneladas de producción de quinua (Instituto Nacional de Estadística e Informática, 2014). La figura 2 se muestra la producción de quinua en el Perú donde muestra un índice de crecimiento.

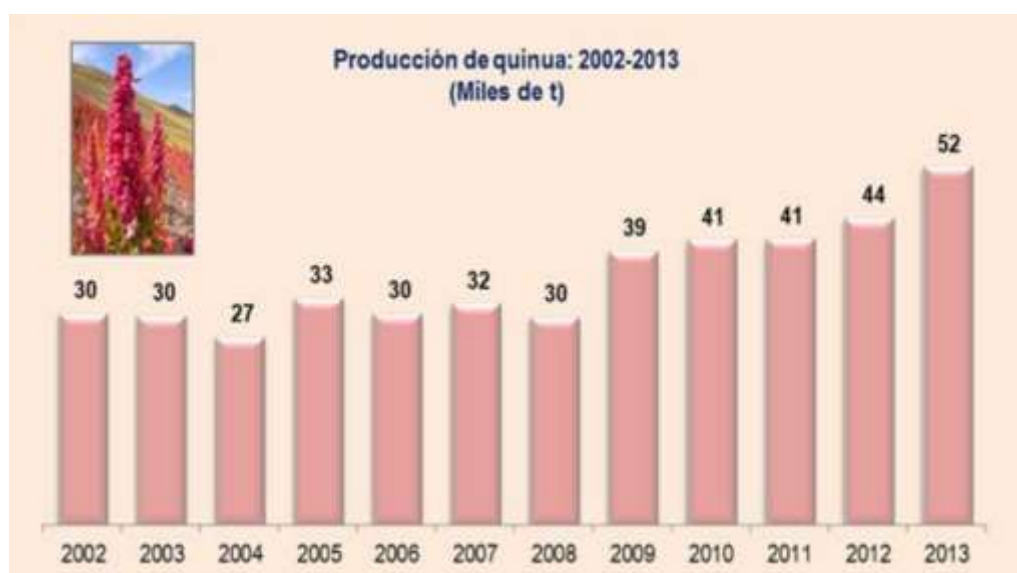


Figura 2. Producción de los granos de quinua 2002-2013.

Fuente: (INEI, 2013).

A medida que la producción de quinua crece, la exportación va en aumento siendo el grano de quinua el más solicitado, mostrado en la figura 3 un crecimiento de exportación muy elevado.

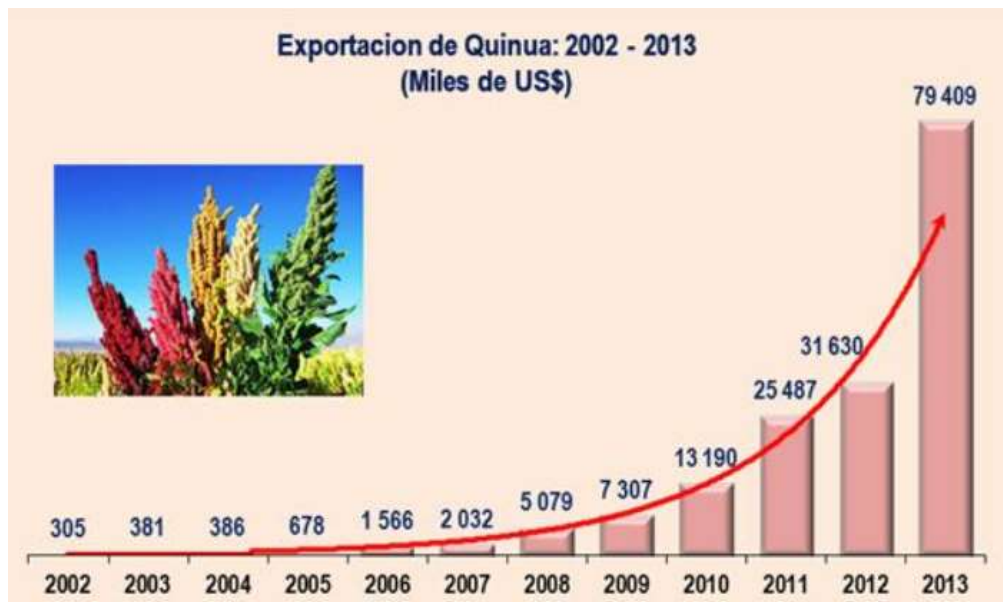


Figura 3. Exportación de granos de quinua 2002-2013.

Fuente: (INEI, 2013).

Según el Ministerio de Agricultura y Riego (2017) muestra el informe de análisis de la producción nacional de la Quinua, donde Puno tiene el mayor índice de producción de 39.6 toneladas a nivel Perú como se aprecia en la figura 4.

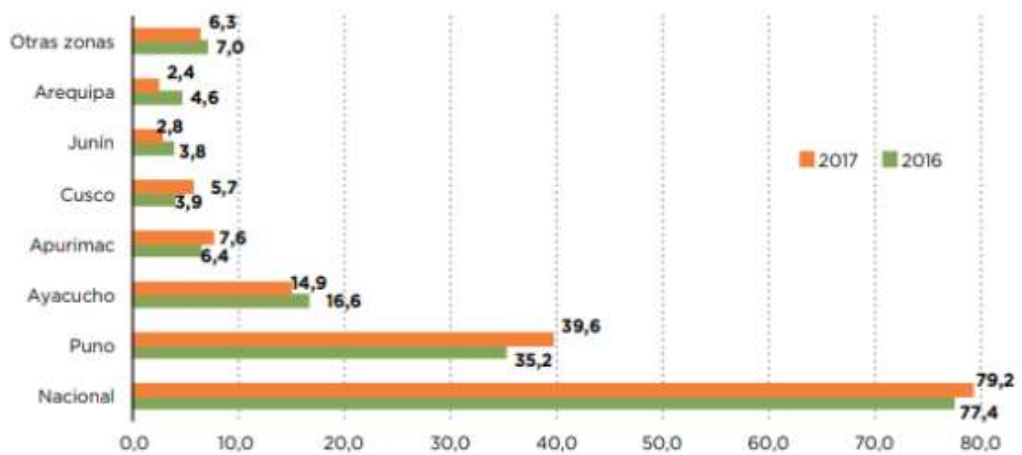


Figura 4. Producción de grano de quinua a nivel nacional (Perú).

Fuente: (MINAGRI, 2017).

La producción del grano de quinua es de unos 79,2 millones a nivel nacional, el procesamiento de este grano andino demanda fortificar su producción, así como su procesamiento.

Es imprescindible tener muestras de granos de quinua procesada para desarrollar una aplicación web que permita clasificarlos, y poder aplicar la NTP 205.062:2014 de granos andinos utilizando las características físicas del grano de quinua.

En la presente Investigación se desarrolla una aplicación web aplicando machine learning, en detalle específico su sub rama deep learning permitiendo clasificar los granos de quinua según la norma técnica peruana NTP 205.062:2014, obteniendo un amplio banco de datos.

1.2. JUSTIFICACIÓN DE LA INVESTIGACIÓN

En la actualidad el procesamiento de imágenes digitales con la automatización es utilizado en diferentes ámbitos como el tecnológico, científico y médico como lo define Nuñez (2017). Por ello este proyecto ayudará a formar una introducción acerca del procesamiento de imágenes aplicado con la sub rama deep learning ofreciendo un enfoque aplicado al ámbito agrícola del procesamiento de clasificación del grano de quinua.

Para determinar el tamaño del grano de quinua, tipo, clases de defecto e impurezas en una muestra, usan un método manual según la norma técnica peruana de quinua, y esta se emplea a las distintas variedades y ecotipos de quinua, destinados solo a la comercialización o consumo humano donde según Trade Map (2019) se exporta 51 mil 107 toneladas de granos de quinua, representando el 44.5% de exportaciones de grano de quinua a nivel mundial.

El presente proyecto busca automatizar el proceso del análisis de la muestra de la quinua implementando machine learning, detallando que no se busca reemplazar el proceso de análisis que también es necesario en la industria alimentaria (CECAGRO, 2016), determinando la clasificación de quinua, número de granos y así como el tipo de grano, grado del grano(tamaño) y estado físico.

La presente investigación desea brindar una herramienta a productores y recolectores de los granos de quinua, obteniendo información del análisis del grano de quinua, proveniente de distintos lugares y en un segundo plano para darle un segundo análisis de quinua procesada.

Creando una fuente de datos sobre el grano de quinua y sus ecotipos, según la norma técnica peruana 205.062.2014, de esta forma el aprendizaje de la aplicación web será constante y seguirá aprendiendo dinámicamente y clasificando los granos de quinua y sus características físicas.

La necesidad de tener este tipo de aplicación web para clasificar quinua es tener una fuente de información de los granos andinos, obteniendo un resultado de acuerdo a la arquitectura de redes neuronales convolucionales aplicada para el aprendizaje del banco de imágenes de quinua y de esta forma poder determinar un análisis completo de cada grano de quinua, almacenando la información de forma permanente en una base de datos. Para crear un banco de información en beneficio de la producción de quinua (productores) y mejorar su producción de quinua según el análisis determinado.

1.3. PRESUNCIÓN FILOSÓFICA

La epistemología busca argumentar cuestiones como ¿Qué es verdadero?, ¿De qué manera conocemos?, esto aporta fiabilidad al conocimiento y a la validez de los orígenes de las cuales conseguimos la información.

Existen diversas corrientes epistemológicas como el escepticismo que manifiesta que los seres humanos no pueden alcanzar el conocimiento confiable y que cualquier búsqueda de la verdad es en inexistente. Esto fue expresado por Gorgias (c. 483-376 a.C.) este sofista griego afirmó que nada existe y que, si así fuera, no podríamos conocerlo. Otras corrientes como el Empirismo donde mencionan que el conocimiento se obtiene por medio de los sentidos, y está relacionada con la experiencia humana.

Una fuente de conocimiento a lo largo de la historia humana ha sido la revelación, tiene una primordial significación en el campo de la fe. Es un conocimiento que se distingue de todos los demás orígenes de conocimiento porque presume una existencia sobrenatural (Knight, 2012).

Como Adventista del Séptimo Día y como egresado de la Universidad Peruana Unión, creo fielmente en que la fuente de todo conocimiento es Dios creador del firmamento, la tierra y todo lo que en ella se encuentra, es por esto solo con la ayuda divina podemos revelar los misterios del universo y así podemos comprender si temor a equivocarnos el conocimiento buscado.

“Porque el Señor otorga la sabiduría; conocimiento y ciencia emergen de sus labios.” (Proverbios 2:6)

1.4. OBJETIVOS

1.5. Objetivo general

Desarrollar una aplicación web que permita clasificar los granos de la quinua según la norma técnica peruana aplicando machine learning.

1.6. Objetivos específicos

- Formar un banco de imágenes de muestras de granos de quinua.
- Identificar características de los granos de quinua según la norma técnica 205.062.2014.
- Entrenar y validar el modelo la red neuronal convolucional de clasificación.
- Implementar el modelo la red neuronal convolucional en una aplicación web.

CAPÍTULO II. Marco teórico

1.7. REVISIÓN DE LA LITERATURA

La tesis por Perez (2016), titulada “Selección de propiedades de interés en la separación de granos de café por medio de un sistema de visión por computadora.” Se implementó un sistema y la distinción de imágenes, desarrollando una aplicación informática en Matlab además de utilizar la NTP 209.027:2001 para la clasificación del café, utilizando parámetros físicos. Implementando tres modelos máquinas de soporte vectorial (Support Vector Machine -SVM), arboles de decisión (Decision Tree -DT) y K-vecino más cercano (K-Nearest Neighbor). Los 3 tipos de clasificadores muestran una precisión 89% y 92.3%.

En el procesamiento de imágenes con OpenCV según Maza (2017) realiza clasificación de cacao con la librería OpenCV desarrollando un sistema prototipo para la selección de cacao procesando las imágenes del cacao, la investigación “Procesamiento de imágenes usando OPENCV aplicado en RASPBERRY PI para la clasificación del cacao” aplicado en la placa de un ordenador RASBERRY, obteniendo la reducción del tiempo de procesamiento de cacao tomando en cuenta las propiedades y la decantación de cacao según sus tres categorías definidas.

Una de las técnicas de aprendizaje para el reconocimiento de imágenes realizada en la investigación “Desarrollo de una aplicación de reconocimiento en imágenes utilizando Deep Learning con OpenCV” donde define el modelo de deep learning con alto índice de identificación según Rodríguez (2017) detallando la creación y el entrenamiento de las redes neuronales enfocados en la visión artificial, utilizando las librerías como OpenCV y TensorFlow esta última detallada como una librería de inteligencia artificial para construir y entrenar redes neuronales, compuestas por distintas arquitecturas. Se propone un problema total de identificación de objetos donde la herramienta implementada ayudara para la resolución de una parte de este. Se buscó un problema con limitaciones eventuales significativas para estudiar el uso de los reconocedores y valorar los requisitos de introducir otras técnicas y mejoras que admitan su utilización en aplicaciones con fuertes limitaciones eventuales. En las investigaciones citas se demuestra el avance de la clasificación de imágenes y procesamiento las soluciones que brindan es la utilización de modelos de redes neuronales convolucionales que resulta de importancia para la clasificación de granos de quinua, como el desarrollo de la aplicación web de vital importancia para la integración del modelo de redes neuronales.

En la tesis desarrollada por Cáceres (2014), titulada “Software automático de reconocimiento de patrones para la discriminación de granos de arroz pulido basado en procesamiento de imágenes digitales” se desarrolla una aplicación del software automático basado en procesamiento digital de imágenes que aminora el tiempo en el conteo general de granos de arroz pulido y en el proceso de exclusión por su clase y tipo de imperfección, así como se reduce el margen de error en la medida de longitud. La investigación usa la metodología Open UP que utiliza perspectivas iterativas e incrementales, en un entorno de ciclo de vida estructurado. Open UP adopta una filosofía pragmática y ágil que se enfoca en la naturaleza colaborativa de desarrollo de software. aporta evidencias valiosas concluyendo que la aplicación del software automático fundamentado en el procesamiento digital de imágenes minimizando el tiempo en el conteo general de granos de arroz pulido y en la operación de exclusión por su clase y tipo de imperfección, así como se reduce el margen de error en la medida de longitud.

1.8. MARCO TEÓRICO

Se presenta el marco teórico comenzando por definir el grano de quinua y las ramas de la inteligencia artificial detallando en específico las redes neuronales convolucionales en específico la arquitectura Inception.

1.9. QUINUA

El grano de quinua (*Chenopodium quinua* Willd) denominado también grano andino es una planta herbácea de ciclo periódico y correspondiente a la familia de los Chenopodiaceae. Sus dimensiones varían desde 1 m a 3.5 m. (INDECOPI, 2014)

1.10. Clasificación de quinua

La clasificación del grano de quinua:

- Quinua ecológica (orgánica o biológica): Es una quinua cuyo sistema de producción, beneficiado, almacenamiento y comercialización está regido por normas de agricultura orgánica, cuya finalidad esencial está fijada al desarrollo del sembrado sostenible.
- Quinua convencional (no ecológica): Es aquella quinua que no tiene los requisitos fijados y que esta detallado en la quinua ecológica, donde podría encontrar la utilización intensiva del suelo y uso de agroquímicos.
- Quinua tradicional (natural): Es aquella quinua cosechada por el agricultor sin la utilización de maquinaria agraria y productos agroquímicos.
- Quinua procesada (beneficiada): Son los granos que han sido introducidos a procesos de saneamiento, escarificado, lavado, secado, despedrado, clasificación y exclusión de granos de color, obteniendo un producto calificado para la adquisición y el consumo del hombre.

1.11. NORMA TECNICA PERUANA

Realizada por el comité técnico de normalización de productos agroindustriales de exportación, dentro de este comité existe un sub comité técnico de normalización de granos andinos. Reemplaza su NTP antecesor 205.062:2009 por la NTP 205.062:2014 basándose en la primera norma y estructurada de acuerdo a las guías peruanas GP 001:1995 y GP 002:1995 (Stadrita, 2015).

2.1.1.1. Campo de aplicación

La NTP se utiliza en las actividades de procesamiento de los granos de quinua, iniciando en el acopio de los granos de quinua, procesamiento, envasado y almacenamiento del producto final (Soto, Valdivia, & Solano, 2019).

2.1.1.2. Definiciones y requisitos

Las operaciones de la planta solas definiciones relativas como son: la limpieza, escarificado, lavado, secado, despedrado y clasificación.

Los requisitos tienen una amplia lista donde las más principales según Soto y Valdivia (2019) son:

- la ubicación de la planta de procesamiento
- Iluminación
- Ventilación
- instalaciones eléctricas
- instalaciones sanitarias
- eliminación de desechos sólidos.

2.1.1.3. Clasificación por el tamaño del grano

Se determina por el diámetro promedio definido en la medida de milímetros(mm) según se aprecia en la figura 5.

Tamaño de los granos	Diámetro promedio de los granos, expresado en mm	Malla
Grandes	Mayor a 1.70 mm	85 % retenido en la malla ASTM 12 (1,68 mm)
Medianos	Entre 1.70 a 14.0 mm	85 % retenido en la malla ASTM 14 (1,41 mm)
Pequeños	Menores a 14.0 mm	85 % que pasa por la malla ASTM 14 (1,41 mm)

Figura 5. Tamaño de los granos de quinua.

Fuente: (INDECOPY, 2014).

Donde se denomina a los granos de quinua según las dimensiones del grano de quinua, las dimensiones designan el grado.

- Grano de quinua grano grande grado 1.
- Grano de quinua grano mediano grado 2.
- Grano de quinua grano pequeño grado 3.

2.1.1.4. Disposiciones relativas a las tolerancias

La clasificación de los granos de quinua es en base a su grado. Según lo especifica la tabla número 1 donde se establece las tolerancias aceptables.

Tabla 1. Clasificación de granos de quinua según el grado.

Parámetros	Unidad	Grado 1		Grado 2	
		Min.	Max.	Min.	Max.
Sensoriales					
Granos enteros	%	97		94	
Granos quebrados	%		1.0		2.0
Granos dañados	%		0.5		0.5
Granos germinados	%		0.25		0.5
Granos recubiertos (vestidos)	%		0.0		0.0
Granos inmaduros (verdes)	%		0.5		0.7
Granos contrastantes	%		0.5		2.0
Impurezas totales	%		0.25		0.3
Piedrecillas en 100g de muestra	U/100g		vació		vació
Insectos (enteros, partes o larvas)	%		vació		vació

Fuente: (INDECOPY, 2014).

2.1.2. INTELIGENCIA ARTIFICIAL

La inteligencia artificial consiste en "Inteligencia" y "Artificial", la inteligencia es un término complejo definido como lógica, autoconciencia, aprendizaje, conocimiento emocional, conciencia, planificación, creatividad, y estas definiciones. Lo artificial es algo simulado, pero no es real y esto no quiere decir que es falso en sentido de ser estafado y una manera es pensar en un césped artificial que funciona igual a un césped real (Ramar, 2019).

La inteligencia artificial (IA) existe desde hace años; sin embargo, donde se puede avanzar es un tema de discusión. Con las tecnologías en desarrollo, actualmente existe una gran demanda de aprendizaje humano integral en aspectos computacionales, capaz de cambiar su propia creencia de comportamiento. Tener la capacidad de decidir, aprender e inculcarse en función de los eventos anteriores y actuar con mucha diligencia.

La IA se refiere a múltiples herramientas y tecnologías que se pueden combinar de diversas maneras para sentir, conocer y realizar con la capacidad de asimilar la experiencia y adecuarse con el tiempo, representado en la figura 6.

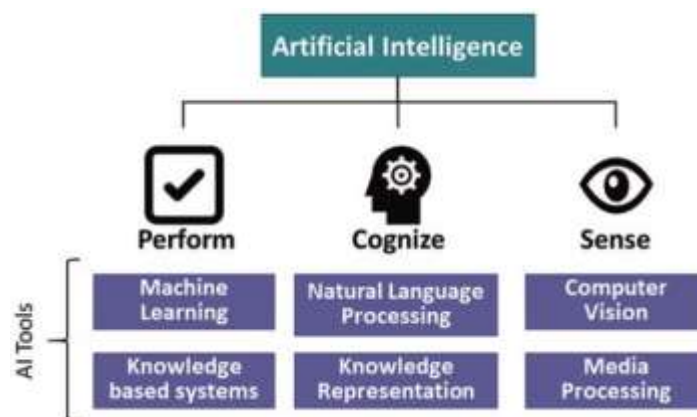


Figura 6. Inteligencia artificial.

Fuente: (Luna Gonzales, 2018).

Dentro de la Inteligencia Artificial se han desarrollado campos de conocimiento como son el Machine Learning y Deep Learning como se representa en la figura 7.

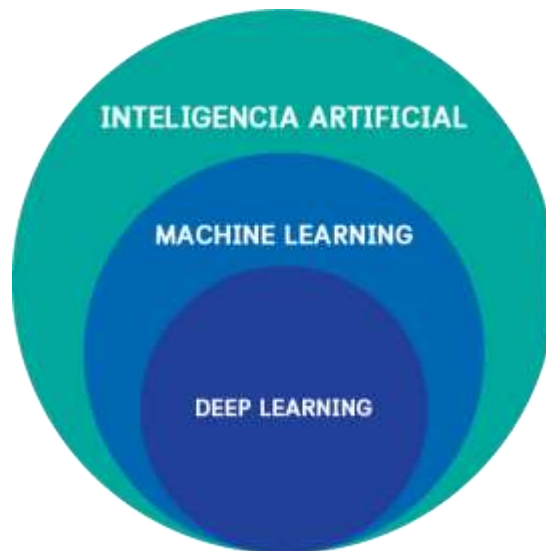


Figura 7. Ramas de la inteligencia artificial.

Fuente: (IAAR, 2019).

2.1.2.1. Machine learning

Aprendizaje automatizado o aprendizaje de máquinas (del inglés, “Machine Learning”) es el subcampo de las ciencias de la computación y una rama de la inteligencia artificial, cuya finalidad es desarrollar técnicas que permitan que las computadoras asimilen información. De manera más precisa, de esta manera crear programas aptos para generalizar conductas a partir de un dato suministrado en forma de patrones (Bonaccorso, 2017).

Machine learning existió durante al menos 60 años atrás. A raíz de la investigación de inteligencia artificial, los iniciales sistemas de aprendizaje automático utilizaban reglas de “if...else” declaraciones codificadas para procesar datos y realizar un arbitraje de decisión (Beyeler, 2017).

Un ejemplo más simple es un depurador de correo no deseado cuya labor es distinguir los correos electrónicos entrantes y mover los mensajes no deseados a un apartado de correo no deseado, representado en la figura 8.

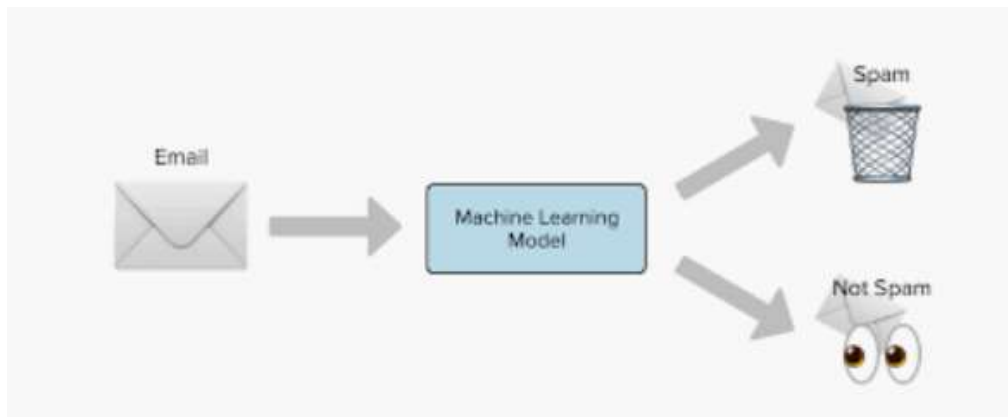


Figura 8. Distinción de spam mediante machine learning.

Fuente: (Packt, 2017).

2.1.2.2. Deep learning

Es una rama de machine learning o también llamado subcampo, deep learning emplea unas diferentes estructuras de redes neuronales para adquirir un aprendizaje, y posteriores capas de representaciones aún más valiosas de los datos (Briega, 2018), mapeando la entrada objetivo mediante una red neuronal organizada por masivas capas libres con una estructura de jerarquía.

Donde Ketkar (2017) afirma que deep learning es la jerarquía de conceptos directamente a partir de datos brutos, otros términos técnicos como aprendizaje de la representación. Es una semejanza del aprendizaje humano pasando los datos brutos a una conclusión sin el paso explícito donde se identifican las características.

2.1.2.3. Arquitecturas de deep learning

La organización de información de una red neuronal esta extravagantemente diseñada como el cerebro humano, nuestro cerebro de la misma forma que hace sinapsis y refuerza algunas regiones de nuestro cerebro se especializan de acuerdo a las entradas que se le presentan y de la misma manera existen arquitecturas de redes neuronales que se adecuan para nutrirse de información de entrada que se les brinda según se requiera.

Las arquitecturas más populares según son:

- **Redes neuronales prealimentadas:** las que inicialmente se desarrollaron y son el modelo más simple los representantes de esta arquitectura son el perceptrón y el perceptrón multicapa. Utilizándose en problemas de clasificación simple.

- **Redes neuronales convolucionales:** son demasiado semejantes a las redes neuronales comunes, el perceptrón multicapa a la vez también tiene una función de pérdida, la única diferencia entre el perceptrón multicapa y las redes neuronales convolucionales, la diferencia son los datos de entrada son imágenes digitales, adquiriendo eficiencia y reducción de las variables de la red.
- **Redes neuronales recurrentes:** la persistencia de datos hace a esta red tan especial, siendo capaces de adquirir aprendizaje a largo plazo, su estructura es de cuatro capas y estas interactúan entre sí brindándole una memoria a largo plazo.

2.1.2.3.1. Redes neuronales convolucionales – CNN

Son los modelos más utilizados para el procesamiento de imágenes digitales y visión por computadora según Beysolow (2017) diseñados de manera que tienen una semejanza con la estructura de la corteza visual animal. Las CNN constan de tres tipos diferentes de capas:

- Capa convolucion, definida como nombre de la red.
- Capa pooling, reduce la cantidad de variables.
- Capa fully connected, brinda el resultado de la red.

La figura 9 representa el procesamiento de una imagen con las CNN.

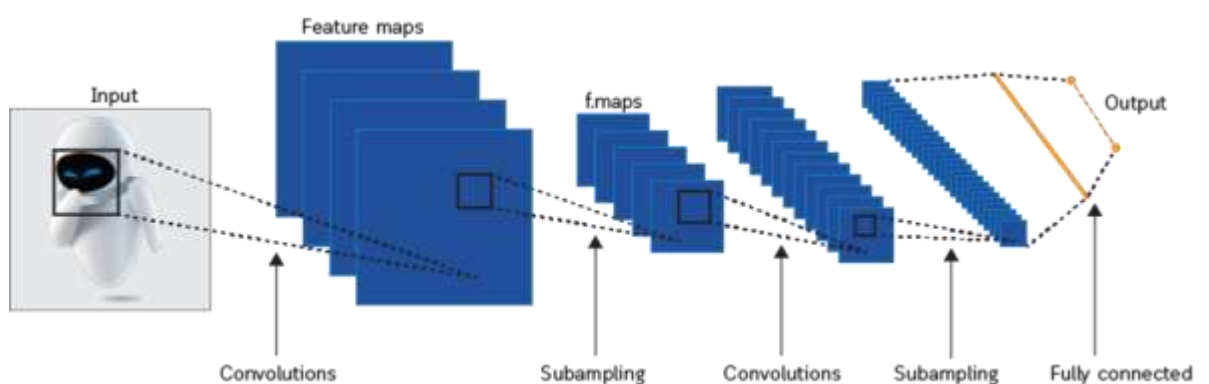


Figura 9. Procesamiento de imágenes.

Fuente: (IAAR, 2019).

2.1.2.3.1. Componentes de las CNN

- **Capa convolution:** Es la primera capa después de la entrada de una imagen, tiene un filtro que escanea un sector de la imagen este filtro no es grande de altura y ancho extendiéndose totalmente en todo el largo de la capa.
- **Capa pooling:** La capa de agrupación ocupa los mapas de características producidas en la capa convolution realizando la reducción de dimensionalidad, reduciendo la complejidad del modelo. Esta capa tiene un filtro de 2 x 2 y una zancada, cada corte de profundidad en la entrada es reducida de manera que se mueve de 2 x 2 pixeles de acuerdo a su altura y ancho. Ayuda a descartar el 75% de la función/activación.
- **Capa fully connected:** Determinada al final de la capa convolution y pooling que determina el usuario, las imágenes en esta capa serán más reducidas de acuerdo a las capas anteriores, su resultado está en una lista de valores. Donde tiene el resultado final.

2.1.3. INCEPTION

Esta arquitectura perteneciente a las CNN gano el ImageNet, concurso de reconocimiento visual a gran escala (ILSVRC) en el 2014 (Bosca, 2017).

El modelo Inception, estructura en la figura 10.

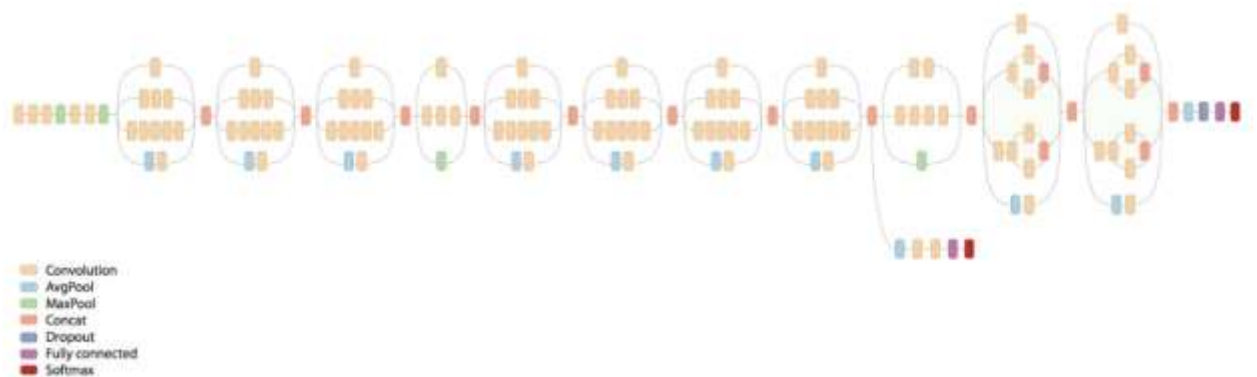


Figura 10. Modelo inception.

Fuente: (IAAR, 2019).

Utilizan varios núcleos convolucionales al inicio y sus tamaños difieren de la dimensión para capturar las características de una imagen en distintas escalas, apilando sus salidas como se representa en la figura 11 el módulo de Inception.

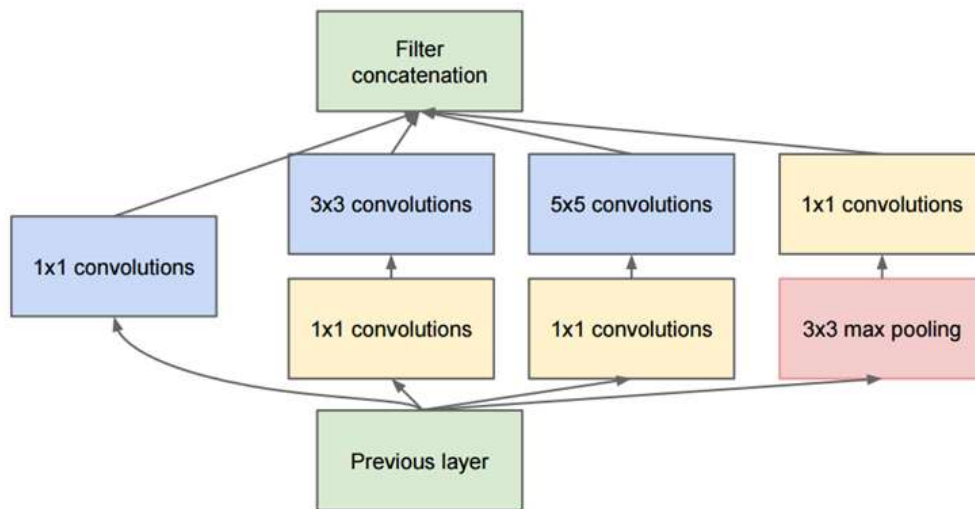


Figura 11. Módulo de inception.

Fuente: (IAAR, 2019).

En la investigación “Ship Classification in High-Resolution SAR Images Using Deep Learning of Small Datasets” manifiesta la evolución del modelo inception mientras más profunda se la red neuronal mejor (Yuanyuan Wang, 2016), donde comparan otras arquitecturas similares a inception demostrando su mejora su clasificación en la tabla 2.

Tabla 2. Clasificación de modelos CNN.

Modelo	Top-1 Precisión (%)	Top-5 Precisión (%)	Profundidad
Inception	78.8	94.4	159
VGG16	71.5	90.1	23
VGG19	72.7	91.1	26
Xception	79	94.5	126

Fuente: Elaboración propia.

2.1.4. PROCESO AGIL UNIFICADO

De nominado también como AUP nos da un mayor marco de buenas prácticas en cuando se refiere a el desarrollo de software, y es más liviano y simple que la metodología RUP. De igual forma presenta la misma semejanza de procesos (Implementación, Pruebas, Despliegue, Gestión de Configuración, Gestión y Entorno) se ajustan con el resto de RUP, además que AUP agrupa en una singular disciplina las disciplinas de Modelado de Negocio, Requisitos y Análisis y Diseño (Ambler, 2002).

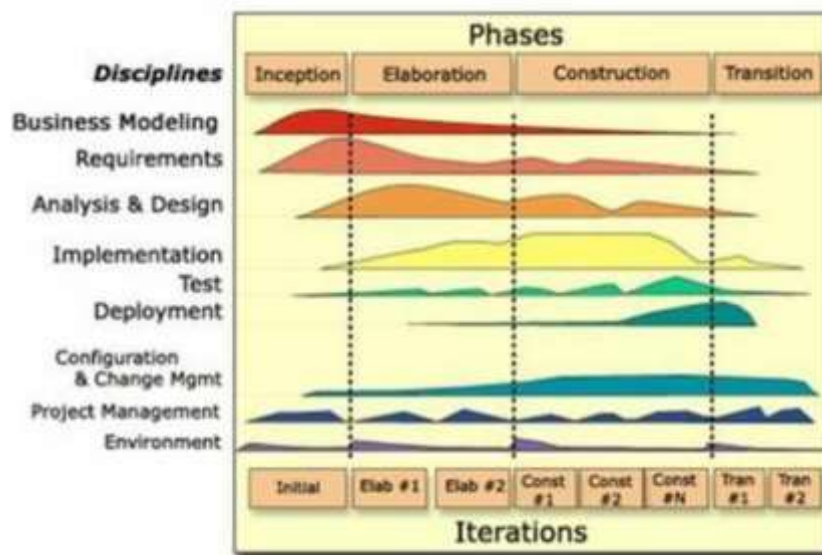


Figura 12. Ciclo de vida de AUP.

Fuente: (AUP, 2014).

Como se muestra en la figura 12 se distinguen las diferentes fases, así como sus disciplinas en el desarrollo de un proyecto de software.

2.1.5. METODOLOGÍA FUNDAMENTAL PARA LA CIENCIA DE DATOS

Para desarrollar una asimilación de los datos se construyen modelos para predecir un resultado donde se necesita una metodología que oriente varias tecnologías agrupando volúmenes de datos y enfoques donde se implican algunos campos como la analítica predictiva, la estadística, data mining y machine learning (Rollins, 2015).

La metodología trata de 10 fases que constituye un pensamiento iterativo para emplear los datos y revelar información. Las distintas fases juegan tienen una labor significativa en el entorno de la metodología general.

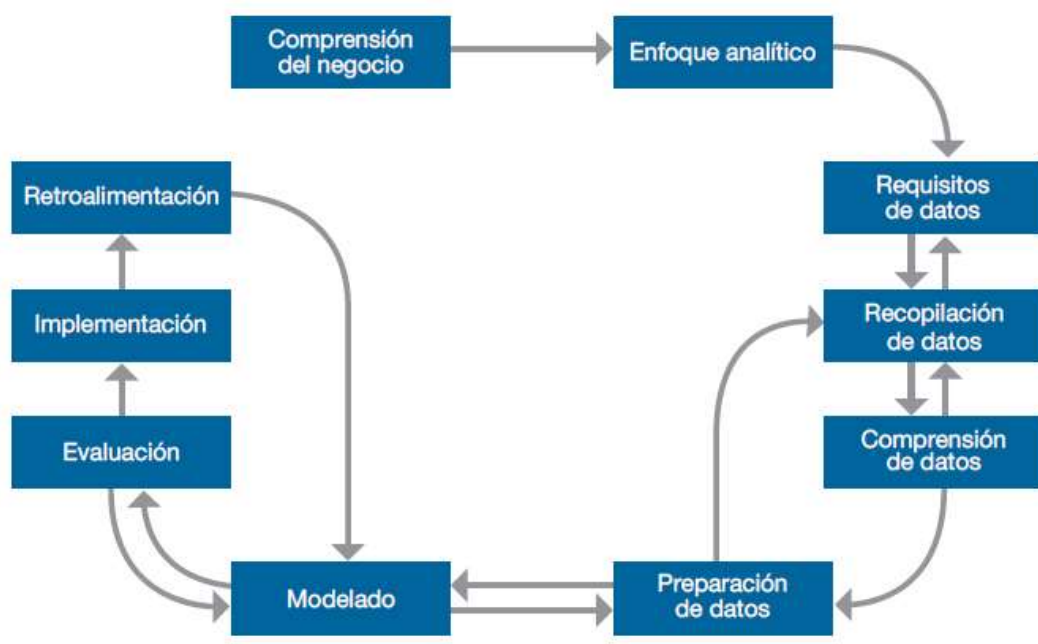


Figura 13. Fases de la metodología fundamental para la ciencia de datos.

Fuente: (IBM, 2015).

2.1.5.1. Comprensión del negocio

Generalmente los proyectos inician con el entendimiento del negocio. La comprensión de negocios que es imprescindible para la opción analítica desempeña la labor más interesante de esta fase, estableciendo el dilema, la meta del proyecto y los requerimientos de la solución visto de una vertiente empresarial. Esta fase inicial sienta los inicios para que el dilema empresarial de un resultado con éxito.

Para defender a asegurar que el proyecto tenga el triunfo, los impulsores tienen que colaborar entretanto tanto siga el proyecto para favorecer madurez en el dominio, probar los descubrimientos intermedios y asegurar la continuidad del proyecto hasta producir la solución esperada.

2.1.5.2. Enfoque analítico

Una vez identificado el dilema empresarial, el investigador de datos puede determinar métodos de análisis para solucionar el problema. Esta fase compromete manifestar el dilema en el entorno de estadísticas y técnicas de machine learning para que la organización permita determinar el método que mejor se adapta al resultado anhelado. Por ejemplo, si la meta es pronosticar la respuesta, como "sí" o "no", entonces el método de análisis puede definirse como la construcción, prueba e implementación de modelos de clasificación.

2.1.5.3. Requisitos de datos

El método de análisis seleccionado determina los requisitos de datos. Más específicamente, el método de análisis que se utilizará requiere cierto contenido, formato y representación de datos bajo la guía del discernimiento del dominio.

2.1.5.4. Recopilación de datos

La fase base de la compilación de datos, los investigadores de datos reconocen y recopilan los recursos de datos aptos (organizados, no organizados y semi-organizados) relacionados con el entorno del problema. Generalmente, deben seleccionar si realizar otras inversiones en elementos de información difíciles de obtener. Es mejor posponer las decisiones de inversión hasta que comprenda mejor los datos y los modelos. Si hay alguna laguna en la compilación de datos, es oportuno que el investigador tenga que verificar los requerimientos de los datos y compilar más datos o datos nuevos.

Sin embargo, el muestreo y la segmentación de datos continúan siendo sustanciales, las propuestas de alta productividad y las capacidades de análisis actuales en la base de datos facilitan a los investigadores de datos a emplear grupos de datos más extensos que albergan grandes proporciones de datos aptos, o incluso totalmente. Al fusionar más datos, los modelos predictivos pueden producir notables eventos extraños, como la incidencia de enfermedades o fallas de los sistemas.

2.1.5.5. Comprensión de datos

Posteriormente de la compilación inicial de datos, estos suelen emplear estadísticas descriptivas, visualización para discernir la entraña de los datos, valorar su calidad y revelar conocimientos preliminares sobre estos. Para ocupar el vacío, puede ser necesario volver a recopilar datos.

2.1.5.6. Preparación de datos

La preparación de datos suele ser el adelanto más extenso de un proyecto de ciencia de datos. En diferentes campos, ciertos avances en la elaboración de datos son habituales para diferentes problemas. Automatizar ciertos avances de elaboración de datos por adelantado puede apresurar el proceso al disminuir el tiempo de elaboración individualizado. Gracias a los sistemas masivamente paralelos de alto rendimiento y las capacidades analíticas almacenadas en los datos actuales, los investigadores de datos pueden elaborar datos de manera más sencilla y ágil empleado grupos de datos muy extensos.

2.1.5.7. Modelado

La etapa de modelado emplea la primera versión del grupo de datos listos y se centra en el avance de modelos predictivos o descriptivos basados en los métodos de análisis previamente definidos. En modelos predictivos, los investigadores de datos emplean grupos de entrenamiento (referencias históricas con resultados de interés conocidos) para construir modelos. A medida que la organización adquiere conocimientos intermedios, el proceso de modelado suele ser muy iterativo, lo que lleva a organizar los datos y limitar la especificación del modelo. Para un procedimiento determinado, los investigadores de datos pueden ensayar con varios algoritmos con sus propios parámetros para hallar el modelo superior de variables utilizables.

2.1.5.8. Evaluación

Entretanto el desarrollo del modelo y anteriormente a su implementación, el investigador de datos valora el modelo para discernir su calidad y respaldar que aborda el dilema empresarial de manera apropiada e íntegramente. La valoración del modelo compromete la deducción de muchas medidas de diagnóstico y de otros rendimientos, como tablas y gráficos, que le concede al investigador de datos deducir la calidad y la eficacia del modelo en la decisión del problema.

Para los modelos de pronóstico, los investigadores de datos usan un grupo de pruebas, que es autónomo del grupo de capacitación, pero sigue una semejante distribución de expectativa y tiene un resultado notable. El grupo de pruebas se utiliza para valorar el modelo para adecuarlo según las exigencias. A veces, el modelo término también se aplica a un grupo de validación para proceder una evaluación término. Mientras el desarrollo del modelo y anteriormente de la implementación, los científicos de datos evalúan el modelo para discernir su calidad y respaldarlo, que resuelva de manera adecuada y completa los problemas comerciales. La evaluación del modelo implica la deducción de diversas medidas de evaluación y otros resultados, como tablas y gráficos, que permiten a los científicos de datos explicar la calidad y eficacia del modelo en la decisión de problemas. Para los modelos de pronóstico, los investigadores de datos utilizan un grupo de ensayos, que es independiente del grupo de entrenamiento, pero sigue una semejante distribución de expectativa y tiene resultados conocidos. El grupo de ensayos se utiliza para valorar el modelo y adecuarlo según sea el requisito. A veces, el modelo término también se aplica al grupo de validación para la evaluación término.

2.1.5.9. Implementación

Una vez que se ha desarrollado y aprobado el modelo exitoso, se puede implementar en un ambiente de producción o un ambiente de prueba comparable. Generalmente, se implementa de forma condicionada hasta que se evalúa completamente su desempeño. La implementación puede ser tan sencilla como generar una documentación con recomendaciones, o tan compleja como introducir el modelo en una puntuación y una corriente de trabajo complejos administrados por una aplicación individualizada. La implementación de un modelo en una fase de operación comercial generalmente envuelve a otros grupos, competencias y tecnologías en el negocio.

Por ejemplo, el equipo de ventas puede implementar un modelo de tendencia de respuesta a través de una serie de gestión de actividades creado por el equipo de desarrollo y gestionado por el equipo de marketing.

2.1.5.10. Retroalimentación

La recopilación de los resultados del modelo implementado proporciona a la organización información sobre el rendimiento del modelo y su impacto en el entorno en el que se implementó. Por ejemplo, la retroalimentación puede ser en forma de porcentajes de respuesta a una campaña publicitaria dirigida a un grupo de clientes identificados por el modelo como posibles respondedores. Los científicos de datos pueden analizar esta retroalimentación para refinar el modelo y mejorar su precisión y utilidad. Puede automatizar algunos o todos los pasos de la evaluación del modelo y la recopilación, conciliación y reimplementación de comentarios del modelo para acelerar el proceso de actualización del modelo y obtener óptimos resultados.

CAPÍTULO III. Materiales y métodos

1.12. Metodología de la investigación

La presente investigación está clasificada de tipo propositiva y tecnológica según Viera (2017) al utilizar algoritmos de machine learning. Dado que es un apoyo de tecnología para el entorno de producción.

Según Fidias (2017) la investigación tecnológica difiere de una investigación científica, detallando que la investigación científica busca nuevos conocimientos mientras la investigación tecnológica obtiene conocimiento para un beneficio. Demostrando que la investigación de tipo tecnológica produce sistemas, software para solucionar y anticipar consecuencias. Señalando las siguientes características:

- Su función es buscar y aplicar el conocimiento para elaborar soluciones.
- Los conocimientos se utilizan de forma rápida y en un reducido tiempo.
- Une métodos, técnicas y procesos de investigación, como también de desarrollo, construcción y test de artefactos (Software, IA, Ciencia de datos, Inteligencia de negocios).

3.1. Arquitectura de la solución

Se detalla en la figura 14 la arquitectura de la solución de la investigación, para ser desarrollada.

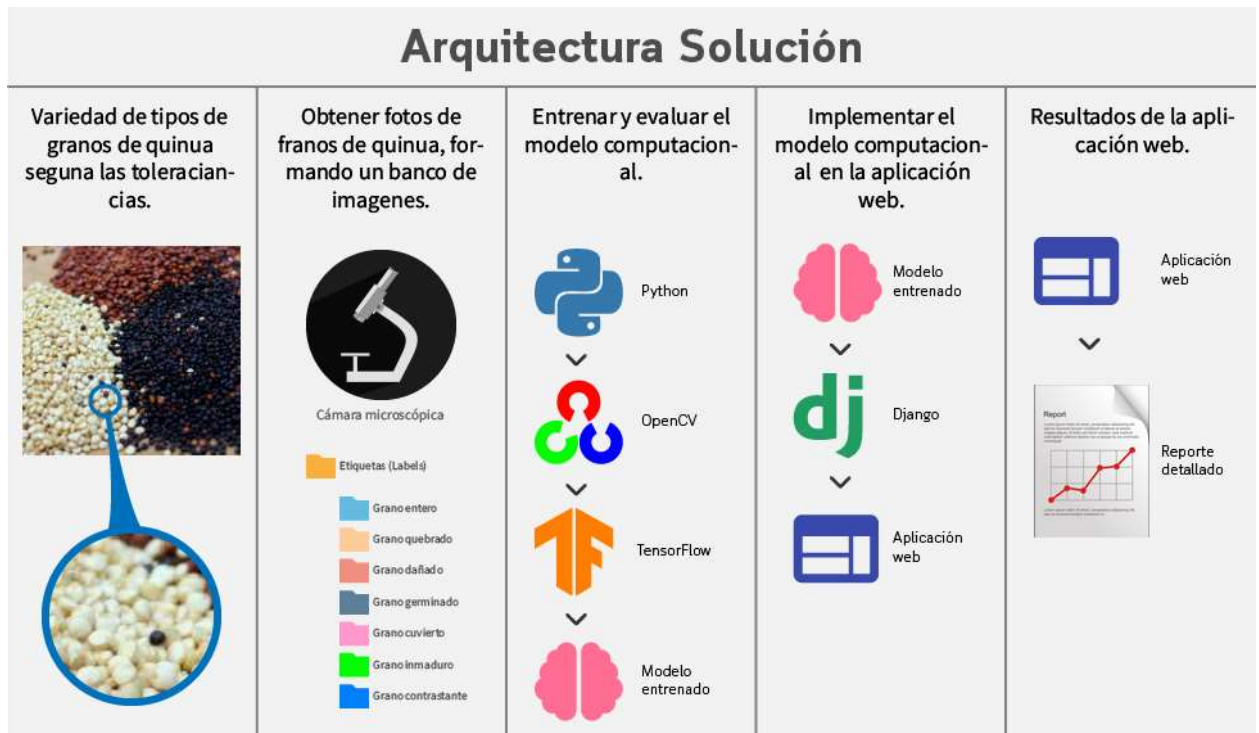


Figura 14. Arquitectura solución.

Fuente: Elaboración propia.

- La recolección de la gran variedad de quinua, los granos de quinua no pueden ser evaluados manualmente de acuerdo a su tamaño.
- Obtener variedad de granos de quinua creando un banco de imágenes, obteniendo imágenes la cámara microscópica, definiendo cada grano por las siguientes etiquetas (labels):
 - Grano Entero.
 - Grano Dañado.
 - Grano Germinado.
 - Grano Inmaduro.
 - Grano Contrastante.

- Desarrollo empleando Python utilizando la arquitectura de las CNN Inception (TensorFlow), adaptando con el procesamiento de OpenCV, de acuerdo a la muestra para el análisis de quinua.
- Entrenamiento y testeo de Inception, con el banco de imágenes.
- Implementación de Inception en la aplicación web desarrollado con el framework Django.
- Análisis detallado de la muestra de granos de quinua analizados.

1.13. Herramientas

3.1.1. Software

En la tabla 3 se describen las herramientas de software, y sus versiones.

Tabla 3. Descripción del software.

Software	Descripción	Versión
OpenCV	<p>Es considerada una de las mejores librerías de software de código abierto para visión artificial y machine learning. Ayudando a construir proyectos completos relacionado a el procesamiento de imágenes, detección de movimiento o la segmentación de imágenes. Permitiendo ejecutar algoritmos de visión artificial, combinando la mejor API de C++ de OpenCV y Python (Villán, 2019).</p> <p>OpenCV es una herramienta muy poderosa, es la librería más extensa de visión por computadora. Ayuda en la portabilidad de modelos y a través de varios lenguajes de programación y frameworks (Aditya Sharma, 2019).</p>	1.2.3
Python	<p>El lenguaje de código abierto (open source) además de ser un lenguaje multi paradigma, tener una gran calidad en su sintaxis. Ofreciéndonos un tipado dinámico fuerte (Spraud, 2018).</p>	3.5
TensorFlow	<p>Es ampliamente utilizado por ser una librería de machine learning y deep learning, creado por Google en uno de sus proyectos Google Brain disponiéndolo más tarde como código abierto. Convirtiéndose en una librería más popular en las comunidades de inteligencia artificial, principalmente porque Google utiliza TensorFlow en la mayoría de sus productos como Google Maps, Gmail entre otras de sus aplicaciones (Pramod Singh, 2020).</p>	91.1
Django Framework	<p>Es un poderoso framework desarrollado en Python que impulsa el desarrollo veloz y un diseño limpio y pragmático, al mismo tiempo ofrece una curva de aprendizaje trivial. Desarrollando fácilmente aplicaciones web simples en poco tiempo, también es un framework robusto y escalable; desarrollando aplicaciones con requisitos complejos e integraciones (Melé, 2020).</p>	94.5
Windows	OS.	10
Ubuntu	OS Xenial.	16.04

Fuente: Elaboración propia.

1.14. Hardware

Las herramientas físicas, están detalladas en la tabla 4.

Tabla 4. Especificación de hardware.

Hardware	Descripción
Laptop	Procesador core i7. Memoria ram 12gb.
Cámara Microscópica	Resolución de 1600X.
Tubo pvc	2" (5cm de diámetro / altura de 15 cm).

Fuente: Elaboración propia.

1.15. Métodos

3.1.2. Transfer learning

También llamado transferencia de aprendizaje es el proceso de copiar el conocimiento de una red ya entrenada a una nueva red para resolver problemas similares. Como es el modelo computacional Inception. Donde se encuentran los siguientes procesos de este método:

- Feature extraction approach.

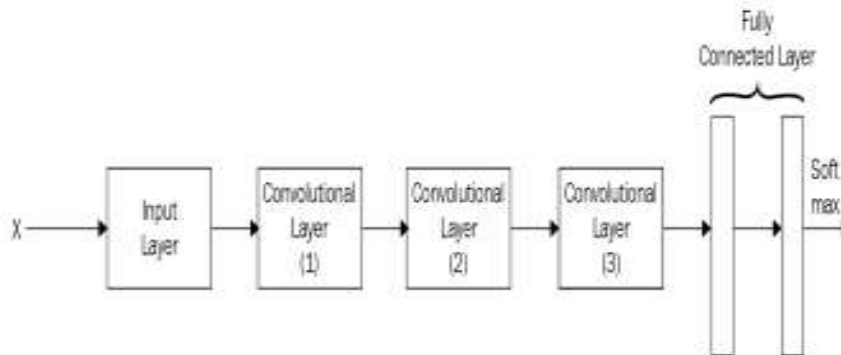


Figura 15. Pre-trained generic CNN.

Fuente: (Rezaul 2019).

- Transfer learning figura 16, un breve ejemplo.

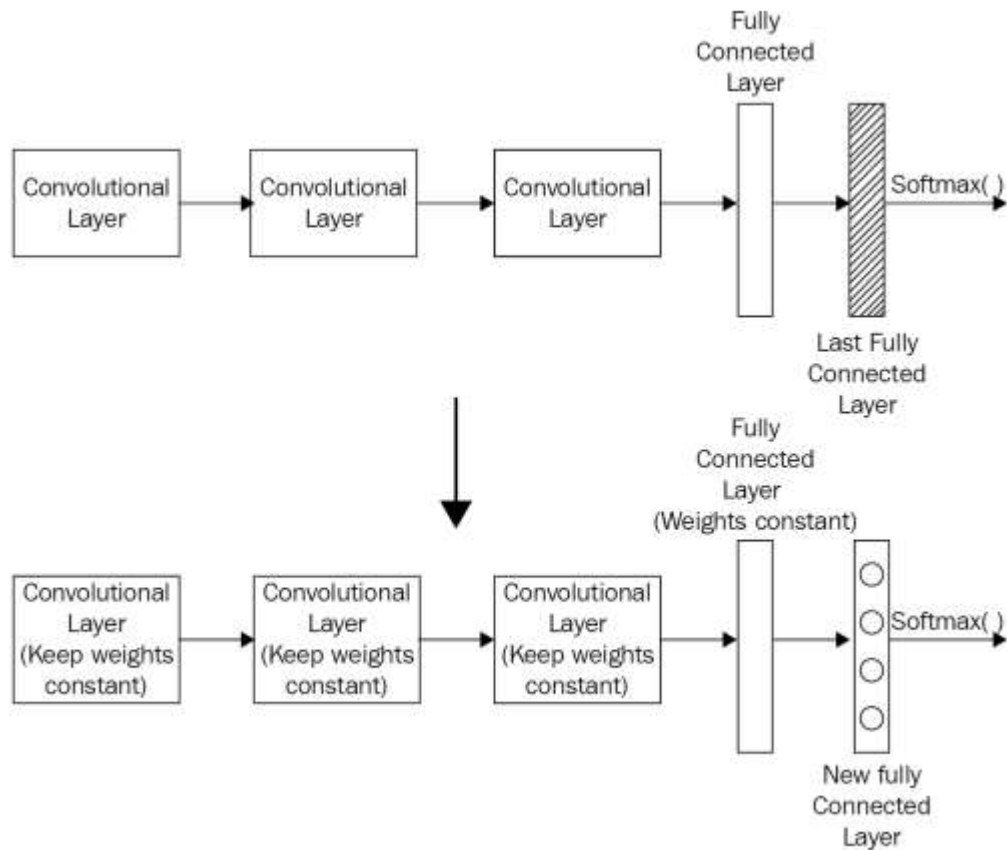


Figura 16. Transfer learning.

Fuente: (Rezaul 2019).

- Multi-task learning: En el aprendizaje multitarea, el aprendizaje por transferencia pasa de ser de un modelo previamente entrenado a muchas tareas simultáneamente. Por ejemplo, en los vehículos autónomos, la red neuronal profunda detecta señales de tráfico, peatones y otros vehículos que circulan por delante al mismo tiempo. El reconocimiento de voz también se beneficia del aprendizaje multitarea.

1.16. Metodología fundamental para la ciencia de datos

3.1.3. Comprensión del negocio

La primera fase en el proceso de comprensión del negocio donde se implica determinar requisitos de parte del negocio.

La comprensión inicia con la NTP 2014. donde se encuentran los requisitos de la clasificación del grano de quinua donde se fijan estándares, los cuales denominamos parámetros, dentro de dichos parámetros nos ayudara a extraer los datos suficientes para su mejor conocimiento y su posterior uso, como se menciona la NTP 2014 donde nos importa el para metro de tipo del grano de quinua:

- Granos de quinua enteros.
- Granos de quinua quebrados.
- Granos de quinua dañados.
- Granos de quinua germinados.
- Granos de quinua recubiertos.
- Granos de quinua inmaduros.
- Granos de quinua contrastantes.

Al definir que solo es un parámetro denominado como tipo de grano, son el que es debidamente que se deben considerar para abstraer los datos necesarios para pasar a la siguiente fase.

3.1.4. Requisitos de datos

Los requisitos principales para poder construir el proyecto son:

- Disponer de los distintos tipos de granos de quinua.
- Cámara microscópica 1600X, a una distancia elevada de 6 centímetros, por un tubo pvc, para su posterior uso.
- Las imágenes de granos de quinua deben estar en el formato JPEG.

3.1.5. Recopilación de datos

Se determina tener imágenes de los distintos tipos de granos de quinua, adquiriendo por cada tipo 1000 imágenes, donde se determinará ordenar cada tipo de grano en carpetas mostradas en la figura 17.

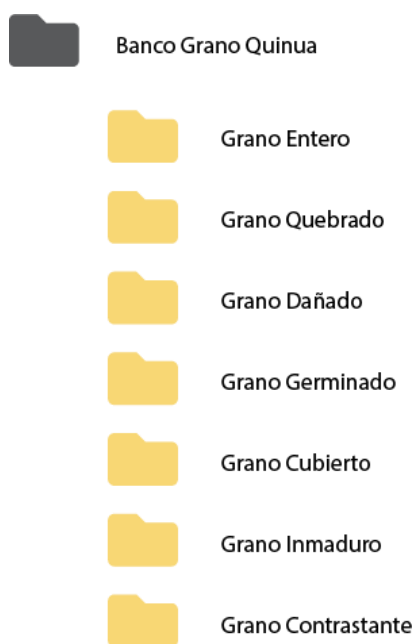


Figura 17. Organización de adquisición de imágenes.

Fuente: Elaboración propia.

3.1.6. Comprensión de datos

Una vez que contamos con una estructura de carpetas elaboradas en la fase de recopilación, nos basamos en la NTP 205.062:2014 para realizar la comprensión de los datos según la clasificación del grano de quinua y sus tolerancias.

- Quinua grano grande - grado 1.
- Quinua grano mediano - grado 2.
- Quinua grano pequeño - grado 3.

3.1.7. Preparación de datos

Una vez se hizo la adquisición de fotos de los granos de quinua se definió que se haga un tratamiento de recorte de imagen por cada grano. Creando una función que recorte las imágenes según encuentre el grano de quinua que se obtuvo en la fase de recopilación de datos.

```
“def split_sample(image_file):
    image_list = {}
    grains = {}
    base_media = os.path.dirname(os.path.dirname(os.path.abspath(__file__)))
    image_file = base_media + image_file
    image_master = cv2.imread(image_file)
    image_gray = cv2.cvtColor(image_master.copy(), cv2.COLOR_BGR2GRAY)
    image_rgb = cv2.cvtColor(image_master.copy(), cv2.COLOR_BGR2RGB)
    image_copy = image_rgb.copy()
    image_outline = image_rgb.copy()
    image_blur = cv2.GaussianBlur(image_gray, (3, 3), cv2.BORDER_DEFAULT)
    image_canny = cv2.Canny(image_blur, 50, 150)
    outlines, hierarchies = cv2.findContours(
        image_canny.copy(), cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE
    )
    grain_counter = 0
    pixels_milimeter = 24.257”
```

Imagen de grano de quinua adquirida será cortada por el algoritmo, y segmentada en la cantidad de granos de quinua que encuentre.



Figura 18. reorganización de DATA_SET.

Fuente: Elaboración propia.

El tratamiento de corte de imágenes se vuelca en la carpeta data_set como se muestra en la figura 18 donde vemos una estructura similar a la fase de recopilación de datos.

3.1.8. Modelado

Se adapta el modelo computacional Inception pre entrenado junto con OpenCV en la aplicación web.

La estructura del módulo IA se aprecia en la figura 19.

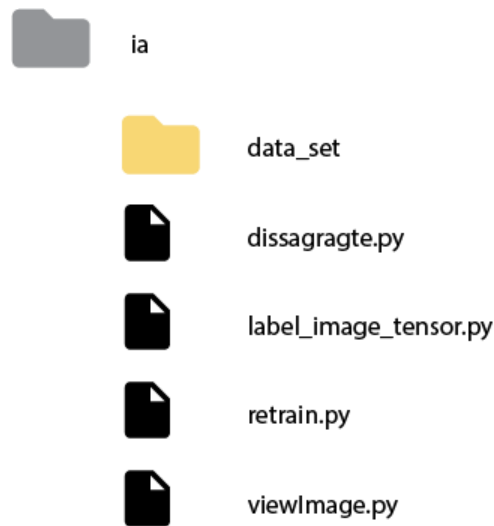


Figura 19. Estructura de carpetas de IA.

Fuente: Elaboración propia.

- **Función mid_point:** Determina el punto medio de un grano de quinua según los ejes “x” y “y”.

```
def mid_point(pointX, pointY):  
    return ((pointX[0] + pointY[0]) * 0.5, (pointX[1] + pointY[1]) * 0.5)
```

- **Función split_sample:** Recibe el parámetro de la imagen completa para su descomposición, según los objetos encuentre. Se definió una variable inmutable para el parámetro “pixels_milimeter” determinando la medida de un grano de quinua, donde se retornará la medida en milímetros. Retornando un Diccionario completo del área del grano, medias del grano (ancho y alto), cantidad de grano encontrados.


```
“from imutils import perspective
from imutils import contours
from scipy.spatial import distance as dist
from .viewImage import viewImage
from .viewImage import viewShow

def split_sample(image_file):
    image_list = {}
    grains = {}
    base_media = os.path.dirname(os.path.dirname(os.path.abspath(__file__)))
    image_file = base_media + image_file
    image_master = cv2.imread(image_file)
    image_gray = cv2.cvtColor(image_master.copy(), cv2.COLOR_BGR2GRAY)
    image_rgb = cv2.cvtColor(image_master.copy(), cv2.COLOR_BGR2RGB)
    image_copy = image_rgb.copy()
    image_outline = image_rgb.copy()
    image_blur = cv2.GaussianBlur(image_gray, (3, 3), cv2.BORDER_DEFAULT)
    image_canny = cv2.Canny(image_blur, 50, 150)
    outlines, hierarchies = cv2.findContours(
        image_canny.copy(), cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE
    )
    grain_counter = 0
    pixels_milimeter = 24.257”
```

```

“for outline in outlines:
    image_list_detail = []
    measures_grain = {}
    area = cv2.contourArea(outline)
    if area > 24.0:
        grain_counter += 1
        x, y, w, h = cv2.boundingRect(outline)
        crop_image = image_copy[y:y+h, x:x+w]

        cv2.drawContours(image_outline, [outline], 0, (0, 255, 0), 1, cv2.LINE_AA)

        box = cv2.minAreaRect(outline)
        box = cv2.boxPoints(box) if im.is_cv2() else cv2.boxPoints(box)
        box = np.array(box, dtype="int")
        box = perspective.order_points(box)

        (tl, tr, br, bl) = box
        (tltrX, tltrY) = mid_point(tl, tr)
        (blbrX, blbrY) = mid_point(bl, br)
        (tlblX, tlblY) = mid_point(tl, bl)
        (trbrX, trbrY) = mid_point(tr, br)

        dA = dist.euclidean((tltrX, tltrY), (blbrX, blbrY))
        dB = dist.euclidean((tlblX, tlblY), (trbrX, trbrY))

        dim_width = dA / pixels_milimeter
        dim_height = dB / pixels_milimeter

        measures_grain['area'] = area
        measures_grain['width'] = dim_width
        measures_grain['height'] = dim_height
        measures_grain['image'] = crop_image
        g_c = str(grain_counter)
        grains[grain_counter] = measures_grain

image_list['total_grains'] = grain_counter
image_list['grains_result'] = grains

return image_list”

```

- **Función load_graph:**

```
“from __future__ import absolute_import
from __future__ import division
from __future__ import print_function

import os
import argparse

import numpy as np
# import tensorflow as tf
import tensorflow.compat.v1 as tf
tf.disable_v2_behavior()

def load_graph(model_file):
    graph = tf.Graph()
    graph_def = tf.GraphDef()

    with open(model_file, "rb") as f:
        graph_def.ParseFromString(f.read())
    with graph.as_default():
        tf.import_graph_def(graph_def)

    return graph”
```

- **Función read_tensor_from_image_file:**

```
“def read_tensor_from_image_file(file_name,
                                input_height=299,
                                input_width=299,
                                input_mean=0,
                                input_std=255):
    input_name = "file_reader"
    output_name = "normalized"
    file_reader = tf.io.read_file(file_name, input_name)
    if file_name.endswith(".png"):
        image_reader = tf.image.decode_png(
            file_reader, channels=3, name="png_reader")
    elif file_name.endswith(".gif"):
        image_reader = tf.squeeze(
            tf.image.decode_gif(file_reader, name="gif_reader"))
    elif file_name.endswith(".bmp"):
        image_reader = tf.image.decode_bmp(file_reader, name="bmp_reader")
    else:
        image_reader = tf.image.decode_jpeg(
            file_reader, channels=3, name="jpeg_reader")
    float_caster = tf.cast(image_reader, tf.float32)
    dims_expander = tf.expand_dims(float_caster, 0)
    resized = tf.image.resize_bilinear(dims_expander, [input_height, input_width])
    normalized = tf.divide(tf.subtract(resized, [input_mean]), [input_std])
    sess = tf.Session()
    result = sess.run(normalized)

    return result”
```

- **Función load_labels:** Lee las etiquetas asignadas de los tipos de quinua. También denominados aquí como labels, como resultado se retornan los labels en una lista.

```
“def load_labels(label_file):  
    label = []  
    proto_as_ascii_lines = tf.io.gfile.GFile(label_file).readlines()  
    for l in proto_as_ascii_lines:  
        label.append(l.rstrip())  
    return label”
```

- **Función prediction_grain:** Se incorporan las funciones load_graph, read_tensor_from_image_file y load_labels, el parámetro de la función es la imagen de la muestra de quinua desagregada echa por la función split_sample se carga la salida graph (output_grahp.pb) y la salida de labels (output_labels.txt) que se obtuvieron durante el entrenamiento del banco de imágenes con el modelo Inception. Obteniendo como resultado la predicción de la imagen de entrada seguido del porcentaje de acierto, retornando los valores mencionados.

```

“def prediction_grain(fn):
    base_media = os.path.dirname(os.path.dirname(os.path.abspath(__file__)))
    file_name = base_media+fn
    # file_name+=fn
    print(file_name)
    model_file = "/tmp/output_graph.pb"
    label_file = "/tmp/output_labels.txt"
    input_height = 299
    input_width = 299
    input_mean = 0
    input_std = 255
    input_layer = "Placeholder"
    output_layer = "final_result"

    graph = load_graph(model_file)
    t = read_tensor_from_image_file(
        file_name,
        input_height=input_height,
        input_width=input_width,
        input_mean=input_mean,
        input_std=input_std)

    input_name = "import/" + input_layer
    output_name = "import/" + output_layer
    input_operation = graph.get_operation_by_name(input_name)
    output_operation = graph.get_operation_by_name(output_name)

    with tf.Session(graph=graph) as sess:
        results = sess.run(output_operation.outputs[0], {
            input_operation.outputs[0]: t
        })
    results = np.squeeze(results)

    top_k = results.argsort()[-5:][::-1]
    labels = load_labels(label_file)
    master_dict = {}
    for i in top_k:
        print(labels[i], results[i])
        master_dict[labels[i]] = results[i]
    return (master_dict)”

```

El entrenamiento se realiza de acuerdo al archivo retrain.py detallado en la figura 19 en la fase del modelado, en donde este dispone del directorio data_set para su entrenamiento.

```

INFO:tensorflow:2019-03-29 11:36:51.761275: Step 210: Validation accuracy = 86.0% (
N=100)
INFO:tensorflow:2019-03-29 11:36:53.264417: Step 220: Train accuracy = 96.0%
INFO:tensorflow:2019-03-29 11:36:53.264417: Step 220: Cross entropy = 0.104192
INFO:tensorflow:2019-03-29 11:36:53.426324: Step 220: Validation accuracy = 77.0% (
N=100)
INFO:tensorflow:2019-03-29 11:36:55.021397: Step 230: Train accuracy = 95.0%
INFO:tensorflow:2019-03-29 11:36:55.021397: Step 230: Cross entropy = 0.122524
INFO:tensorflow:2019-03-29 11:36:55.157317: Step 230: Validation accuracy = 82.0% (
N=100)
INFO:tensorflow:2019-03-29 11:36:56.568511: Step 240: Train accuracy = 99.0%
INFO:tensorflow:2019-03-29 11:36:56.568511: Step 240: Cross entropy = 0.044709
INFO:tensorflow:2019-03-29 11:36:56.709433: Step 240: Validation accuracy = 93.0% (
N=100)
INFO:tensorflow:2019-03-29 11:36:58.025678: Step 250: Train accuracy = 94.0%
INFO:tensorflow:2019-03-29 11:36:58.025678: Step 250: Cross entropy = 0.146836
INFO:tensorflow:2019-03-29 11:36:58.154606: Step 250: Validation accuracy = 85.0% (
N=100)
INFO:tensorflow:2019-03-29 11:36:59.507833: Step 260: Train accuracy = 98.0%
INFO:tensorflow:2019-03-29 11:36:59.507833: Step 260: Cross entropy = 0.063185
INFO:tensorflow:2019-03-29 11:36:59.644754: Step 260: Validation accuracy = 84.0% (
N=100)
INFO:tensorflow:2019-03-29 11:37:00.935929: Step 270: Train accuracy = 98.0%
INFO:tensorflow:2019-03-29 11:37:00.935929: Step 270: Cross entropy = 0.126858
INFO:tensorflow:2019-03-29 11:37:01.093837: Step 270: Validation accuracy = 88.0% (
N=100)
INFO:tensorflow:2019-03-29 11:37:02.490654: Step 280: Train accuracy = 99.0%
INFO:tensorflow:2019-03-29 11:37:02.491660: Step 280: Cross entropy = 0.063198
INFO:tensorflow:2019-03-29 11:37:02.658580: Step 280: Validation accuracy = 84.0% (
N=100)
INFO:tensorflow:2019-03-29 11:37:04.269643: Step 290: Train accuracy = 99.0%
INFO:tensorflow:2019-03-29 11:37:04.269643: Step 290: Cross entropy = 0.067366
INFO:tensorflow:2019-03-29 11:37:04.401502: Step 290: Validation accuracy = 87.0% (
N=100)
INFO:tensorflow:2019-03-29 11:37:05.680773: Step 300: Train accuracy = 100.0%
INFO:tensorflow:2019-03-29 11:37:05.681771: Step 300: Cross entropy = 0.052228
INFO:tensorflow:2019-03-29 11:37:05.808702: Step 300: Validation accuracy = 92.0% (
N=100)
INFO:tensorflow:2019-03-29 11:37:07.167923: Step 310: Train accuracy = 96.0%
INFO:tensorflow:2019-03-29 11:37:07.168923: Step 310: Cross entropy = 0.116766
INFO:tensorflow:2019-03-29 11:37:07.302845: Step 310: Validation accuracy = 81.0% (
N=100)

```

Figura 20. Entrenamiento de inception.

Fuente: Elaboración propia.

En la figura 20 indica cómo se realiza el entrenamiento del modelo computacional, y una breve parte de pasos que generando según su data set.

3.1.9. Evaluación

En la evaluación se tuvo en cuenta el número de pasos(steps), train accuracy, cross entropy y validation accuracy. representado en una matriz de confusión.

Tabla 5. Matriz de confusión.

	Confusión	Matrix	
Whole	99.1	0.04	0.02
Break	1.00	99.3	0.01
Coated	0.00	0.00	99.6
Germin	0.90	0.03	0.02
	whole	break	coated

Fuente: Elaboración propia.

Donde se muestra el alto índice de acercamiento de decidir qué tipo de label es exactamente según se aprecia en la tabla 5 se verifica el indicio de confusión que puede tener el modelo computacional Inception y el alto porcentaje de predicción.

3.1.10. Implementación

Una vez terminada la fase de la evaluación el siguiente paso es el funcionamiento del proyecto donde procedió a la implementación del algoritmo dentro de la aplicación web llamada AGRAIN, se pasa los resultados a dicha aplicación web como la predicción del tipo de los granos de quinua como el resultado final del proyecto. Para una funcionalidad unificada que funcione dentro de la aplicación web como uno solo.

3.2. Metodología AUP para el desarrollo de la aplicación web

Para el avance de la aplicación web también definido como la interfaz gráfica del modelo computacional Inception, se utilizó la metodología AUP

3.3. Fase I: Iniciación

La fase iniciación determina el enfoque de la aplicación web y establecer los casos de uso y los actores definiendo los procesos que realiza cada actor de manera que justifique el desarrollo del proyecto.

3.3.1. Requerimientos

Los requerimientos se detallan de dos tipos los funcionales definidos como prioridad y los no funcionales no son necesarios para la aplicación web.

3.3.1.1. Requerimientos funcionales

Tabla 6. Condiciones funcionales de la aplicación web.

<i>Id</i>	<i>Modulo</i>	<i>Funcionalidad</i>	<i>Descripción</i>
AGRAIN001	Grain	Selección de una carpeta de entrenamiento.	El usuario podrá tener la opción de elegir un directorio de entrenamiento donde se añadirá las imágenes.
AGRAIN002	Grain	Almacenar una de imagen.	Registrar una imagen (Muestra de quinua) para ser desagregada.
AGRAIN003	Grain	Detalle de la imagen muestra.	La imagen muestra debe desagregarse, teniendo detalle de cada sub imagen.
AGRAIN004	Grain	CRUD de imágenes de entrenamiento.	El usuario registrara imágenes para el entrenamiento, así como poder bórralas.
AGRAIN005	Grain	CRUD de etiquetas.	Registrar etiquetas, listarlas y bórralas del conjunto de etiquetas (Labels).
AGRAIN006	Grain	CRUD análisis de sub imagen.	Es importante mostrar la sub imagen, su alto su ancho y el tipo de etiqueta asignado, el porcentaje de predicción.
AGRAIN007	Grain	Realizar predicción de imagen.	Selección de imagen comprobar la predicción.
AGRAIN008	Grain	Reporte pdf.	el análisis de predicción, de la imagen mostrando detallando el análisis de la sub imagen y la imagen base.
AGRAIN009	Provider	CRUD proveedor.	

			Registrar la persona proveedora de la imagen muestra, listar y borrar.
AGRAIN010	City	CRUD ciudad.	Registrar ciudad de donde proviene la imagen muestra
AGRAIN011	Auth	Login.	El usuario debe tener u nombre de usuario y una contraseña, permitiendo autenticarse.
AGRAIN012	Auth	Permisos de usuario.	el usuario debe tener accesos según se designé.

Fuente: Elaboración propia.

3.3.1.2. Requerimientos no funcionales

Tabla 7. Condiciones no funcionales de la aplicación web.

<i>Id</i>	<i>Funcionalidad</i>	<i>Descripción</i>
NOO001	Adaptar a resolución de dispositivo.	la aplicación debe poder adaptarse a cualquier resolución de dispositivo.
NOO002	Notificación de Errores.	se debe proporcionar errores que oriente al usuario a seguir con un procedimiento.
NOO003	Despliegue en Internet.	La aplicación debe estar desplegado en internet para poder ser accesible desde cualquier dispositivo.

Fuente: Elaboración propia.

3.4. Fase II: Elaboración

3.4.1. Actores

Se identificaron los siguientes actores (usuarios del negocio) que relación con la aplicación web que se manifiesta en la figura 21.

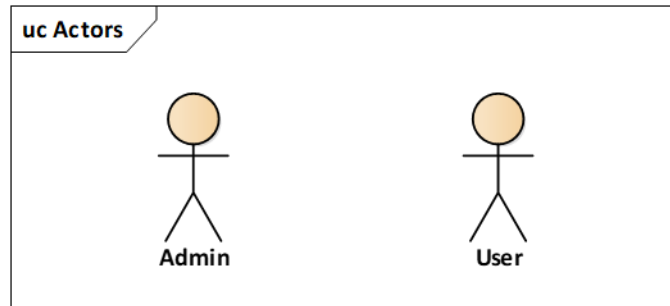


Figura 21. Actores de AGRAIN.

Fuente: Elaboración propia.

La tabla 8 muestra la descripción de los actores identificados.

Tabla 8. Definición de actores.

Actor	Descripción
Admin	Tiene el control absoluto de la aplicación web.
User	Realiza el registro de los proveedores e imágenes, obteniendo reportes del análisis de la imagen.

Fuente: Elaboración propia.

3.4.2. Diagrama de casos de uso

Se muestra en la figura 22 la relación de dos actores identificados en modelo del negocio.

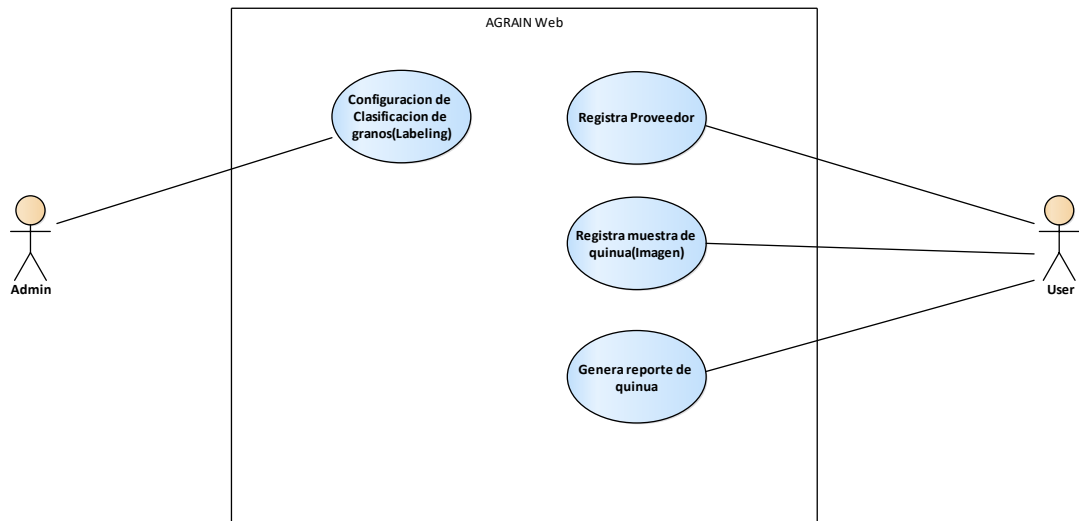


Figura 22. Diagrama de Usuario y Administrador

Fuente: Elaboración propia.

3.4.2.1. Especificación de casos de uso

Detallamos los casos de uso definiendo cada uno de los procesos que tiene cada uno de los actores.

- **Admin:** el actor admin está definido en la tabla 9.

Tabla 9. Especificación de caso de uso para la clasificación de granos quinua.

Actor: Admin																
Nombre	Configuración y clasificación de granos (Labeling).															
Propósito	Crear un entorno de configuración para clasificar y hacer el labeling de las imágenes clasificadas.															
Resumen	Configurar la aplicación web para la clasificación de los granos.															
	<table border="0" style="width: 100%;"> <thead> <tr> <th style="width: 30%;"></th> <th style="width: 35%; text-align: center;">Evento Actor</th> <th style="width: 35%; text-align: center;">Evento Web</th> </tr> </thead> <tbody> <tr> <td>Flujo Principal</td> <td>El admin registra la clasificación de granos.</td> <td>La aplicación web sube las imágenes y registra el tipo de clasificación de grano.</td> </tr> <tr> <td>Flujo Alternativo</td> <td colspan="2">Tener un banco de imágenes previamente, con el labeling por grano.</td> </tr> <tr> <td>Pre Condición</td> <td colspan="2">Se debe tener previamente el labeling de cada imagen.</td> </tr> <tr> <td>Pos Condición</td> <td colspan="2">Ninguna.</td> </tr> </tbody> </table>		Evento Actor	Evento Web	Flujo Principal	El admin registra la clasificación de granos.	La aplicación web sube las imágenes y registra el tipo de clasificación de grano.	Flujo Alternativo	Tener un banco de imágenes previamente, con el labeling por grano.		Pre Condición	Se debe tener previamente el labeling de cada imagen.		Pos Condición	Ninguna.	
	Evento Actor	Evento Web														
Flujo Principal	El admin registra la clasificación de granos.	La aplicación web sube las imágenes y registra el tipo de clasificación de grano.														
Flujo Alternativo	Tener un banco de imágenes previamente, con el labeling por grano.															
Pre Condición	Se debe tener previamente el labeling de cada imagen.															
Pos Condición	Ninguna.															

Fuente: Elaboración propia.

- **User:** En la tabla 10, 11 y 12 se especifica los casos de uso de un usuario común, detallando las acciones que puede realizar.

Tabla 10. Especificación de caso de uso para el registro de proveedor.

Actor: User		
Nombre	Registrar Proveedor	
Propósito	El registro de los datos de proveedores, simplifica el control de proveedores.	
Resumen	Registro de proveedores y control (Datos personales).	
	Evento Actor	Evento Web
Flujo Principal	El user registra los datos personales de un proveedor nuevo.	La aplicación web almacena los datos personales de cada proveedor, generando un registro.
Flujo Alternativo	Ninguna.	
Pre Condición	Disponer de los datos esenciales del proveedor.	
Pos Condición	Ninguna.	

Fuente: Elaboración propia.

Tabla 11. Especificación de caso de uso para el registro de la muestra.

Actor: User		
Nombre	Registrar muestra de quinua (Imagen).	
Propósito	La muestra tomada de quinua, se registra para analizar detalladamente cada grano para un posterior análisis.	
Resumen	Registro de la muestra de quinua mediante una imagen.	
	Evento Actor	Evento Web
Flujo Principal	El user sube la imagen de la muestra de quinua. Guarda el registro.	La aplicación web almacena la imagen y datos adicionales previo al análisis de la imagen.
Flujo Alternativo	Ninguna.	
Pre Condición	Obtener una imagen previa de una muestra de quinua.	
Pos Condición	Ninguna.	

Fuente: Elaboración propia.

Tabla 12. Especificación de caso de uso para generar el reporte del análisis.

Actor: User		
Nombre	Genera reporte de quinua.	
Propósito	Genera reportes por cada muestra registrada con el detalle de las medidas y clasificación de granos.	
Resumen	Generación de documentación de la clasificación.	
	Evento Actor	Evento Web
Flujo Principal	El user elige una muestra previamente almacenada. Y se dirige la opción reporte.	La aplicación web verifica la imagen, empieza a crear el reporte en formato pdf.
Flujo Alternativo	Ninguna.	
Pre Condición	Tener el registro de una muestra de quinua.	
Pos Condición	Ninguna.	

Fuente: Elaboración propia.

3.4.3. Arquitectura de la aplicación web propuesta

La aplicación web funciona con el framework Django que maneja la arquitectura que enfatiza en el patrón modelo, vista controlador (MVC). La figura 23 describe la arquitectura.

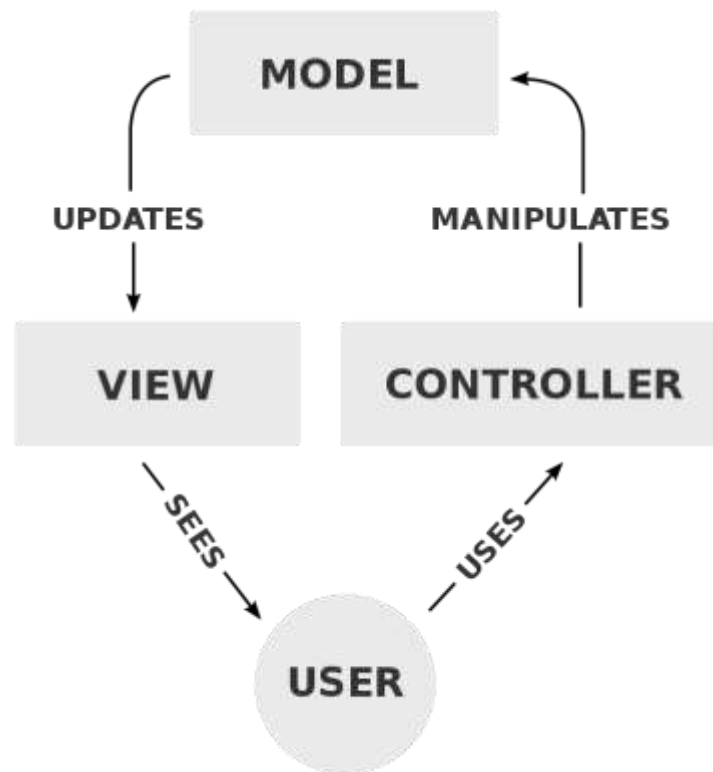


Figura 23. Arquitectura MVC.

Fuente: (DjangoProject, 2019).

La organización de carpetas de la aplicación web, contiene los componentes del proyecto y la app creada por el framework Django señalada en la figura 24.

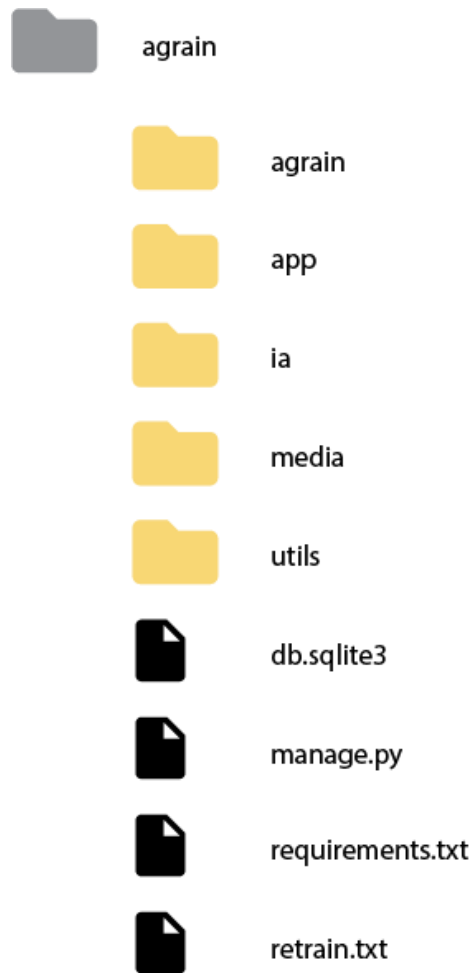


Figura 24. Carpetas de la aplicación web.

Fuente: Elaboración propia.

3.5. Fase III: Construcción

3.5.1. Modelo de base de datos

El diagrama de la base de datos, define la relación de las tablas como se percibe en la figura 25 de acuerdo a la aplicación web.

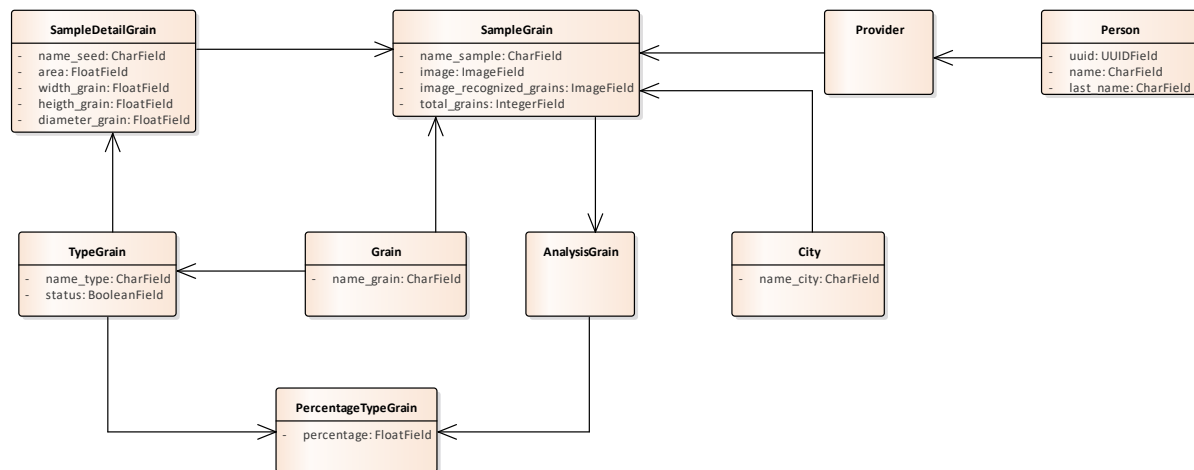


Figura 25. Modelo de la base de datos.

Fuente: Elaboración propia.

3.5.2. Diagrama de componentes

Se detalla los componentes de AGRAIN para su posterior implementación.

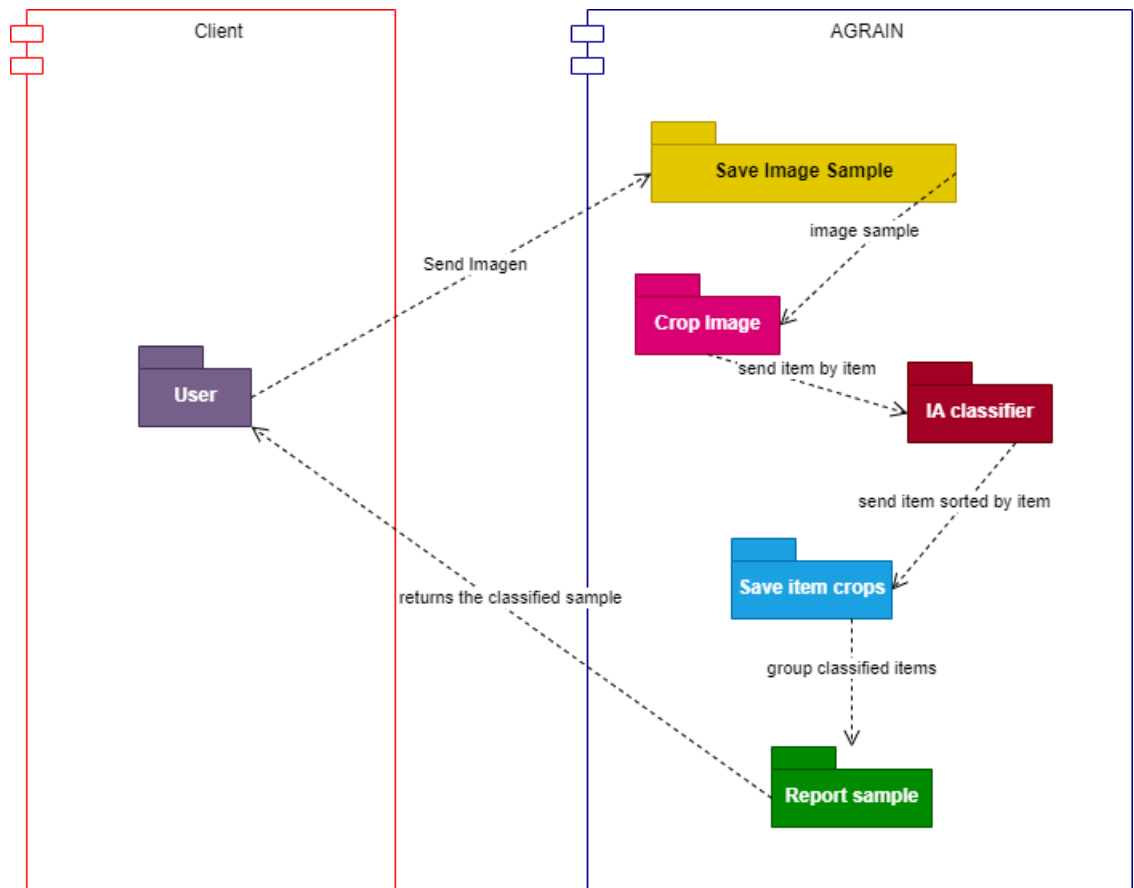


Figura 26. Componentes de AGRAIN.

Fuente: Elaboración propia.

Los componentes detallados en la figura 26 muestran los módulos de AGRAIN, con el módulo computacional Inception.

3.6. Aplicación web

La aplicación web presenta los siguientes módulos:

- Authentication and Authorization: Permisos de usuario y autenticación.
- Grain: contiene toda la funcionalidad del análisis completo de la muestra de quinua.
- Provider: registro de proveedores de quinua.

El código desarrollado de la aplicación web esta publicado en el repositorio de GitHub:

- <https://github.com/ZeroPaul/Agrain-US>.

Las imágenes obtenidas de los granos de quinua agrupadas en el directorio datan set esta publicado en:

- <https://github.com/ZeroPaul/Agrain-US/tree/master/ia/bank%20grain>.

3.6.1. Interfaz login

La interfaz login autentifica al usuario, redirigiendo al inicio para poder acceder a la funcionalidad de la aplicación web. Solicitando al usuario un username (nombre de usuario) y un password (contraseña), donde en la figura 27 se puede apreciar.



Figura 27. Login de la aplicación AGRAIN.

Fuente: Elaboración propia.

3.6.2. Interfaz menú

Esta interfaz muestra dos menús principales la del admin (administrador) y la de user (usuario). En la figura 28 se observa el menú admin con todos los accesos a la aplicación web y en la figura 29 muestra el menú user con los accesos delimitados.



Figura 28. Menú del admin (administrador).

Fuente: Elaboración propia.

La interfaz del usuario muestra los menús con los permisos esenciales para poder realizar un análisis de granos de quinua mostrado en la figura 29.

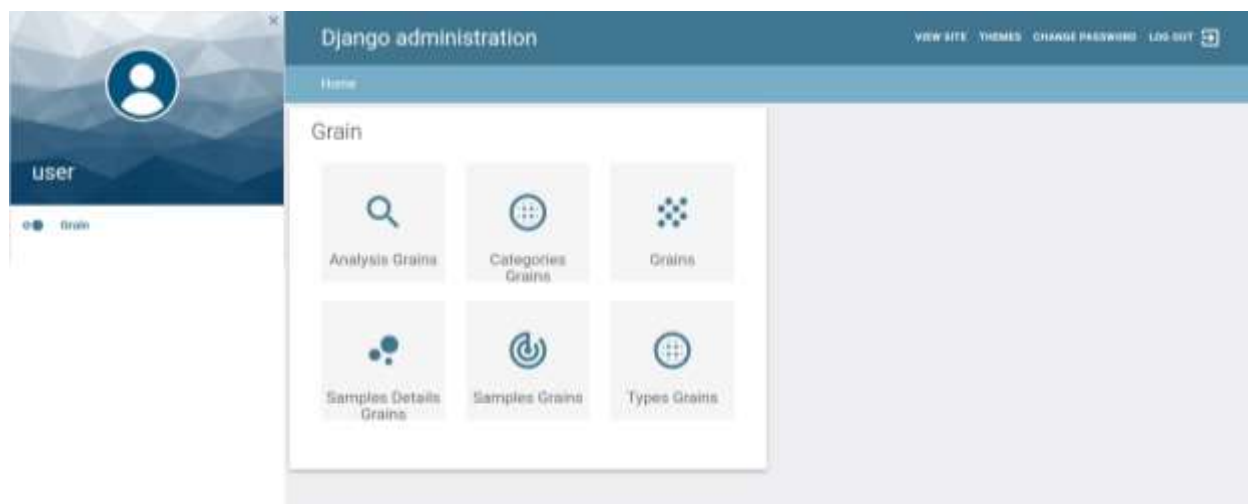


Figura 29. Menú principal de user(usuario).

Fuente: Elaboración propia.

3.6.3. Interfaz grain

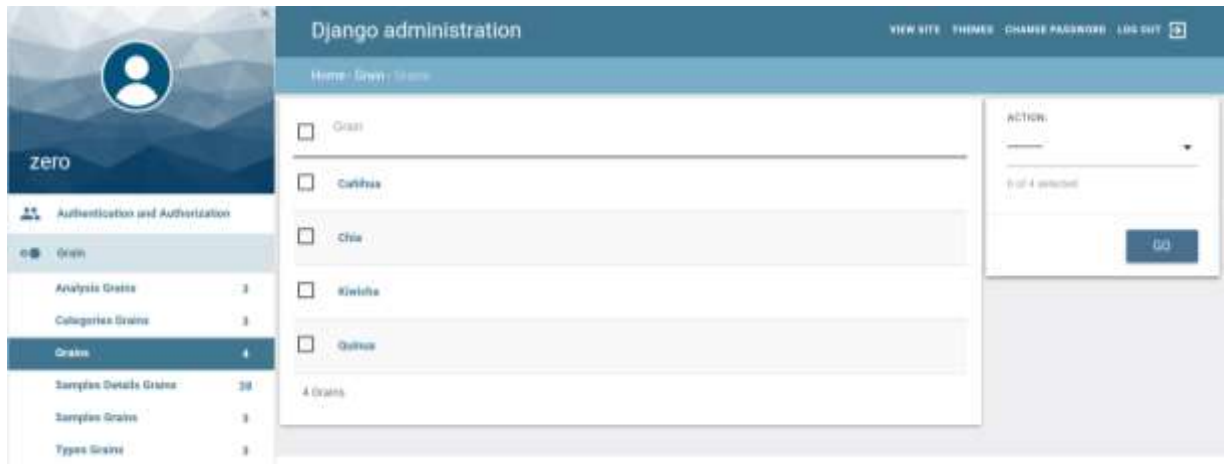


Figura 30. CRUD grain (granos andinos).

Fuente: Elaboración propia.

Las siguientes interfaces como se ve en la figura 30 muestran el módulo de granos de quinua y su respectiva clasificación.

La figura 31 se observa las etiquetas agregadas, de acuerdo al tipo de grano.

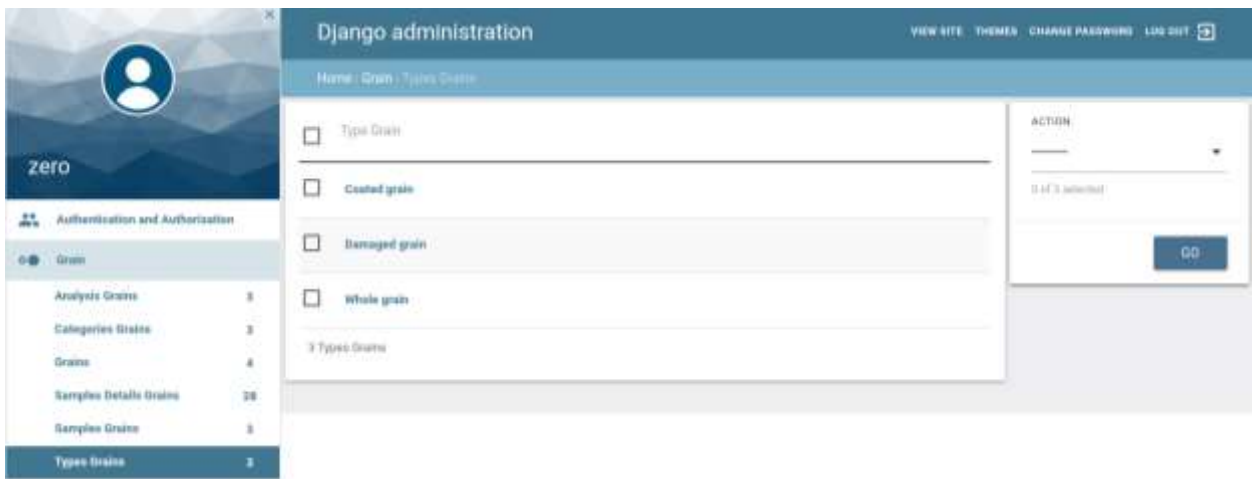


Figura 31. CRUD type grains (tipos de granos).

Fuente: Elaboración propia.

El listado de muestras de granos de quinua se aprecia en la figura 32 donde se ve las muestras registradas.

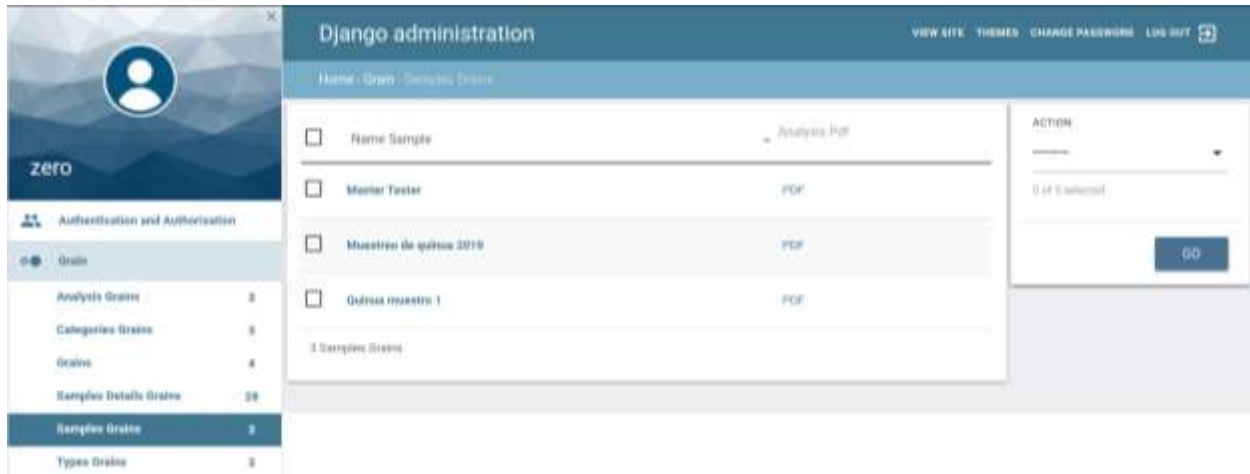


Figura 32. CRUD sample grain (muestra del grano).

Fuente: Elaboración propia.

El análisis de muestra de grano de quinua se puede ver más detallado según las características que encuentre como en la figura 33.

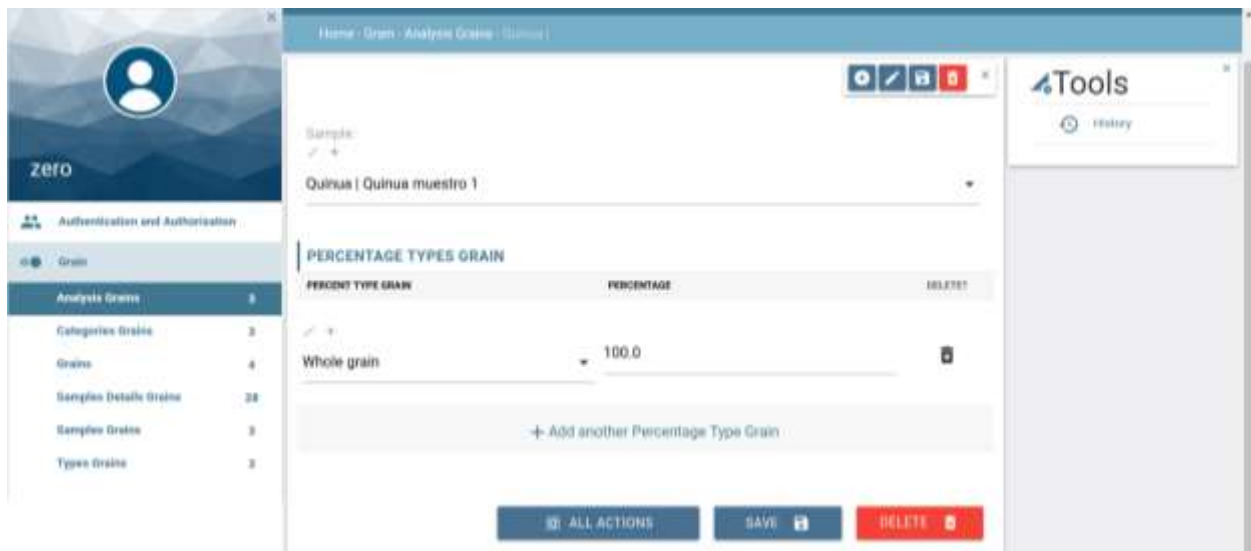


Figura 33. CRUD analisis grain (análisis de granos).

Fuente: Elaboración propia.

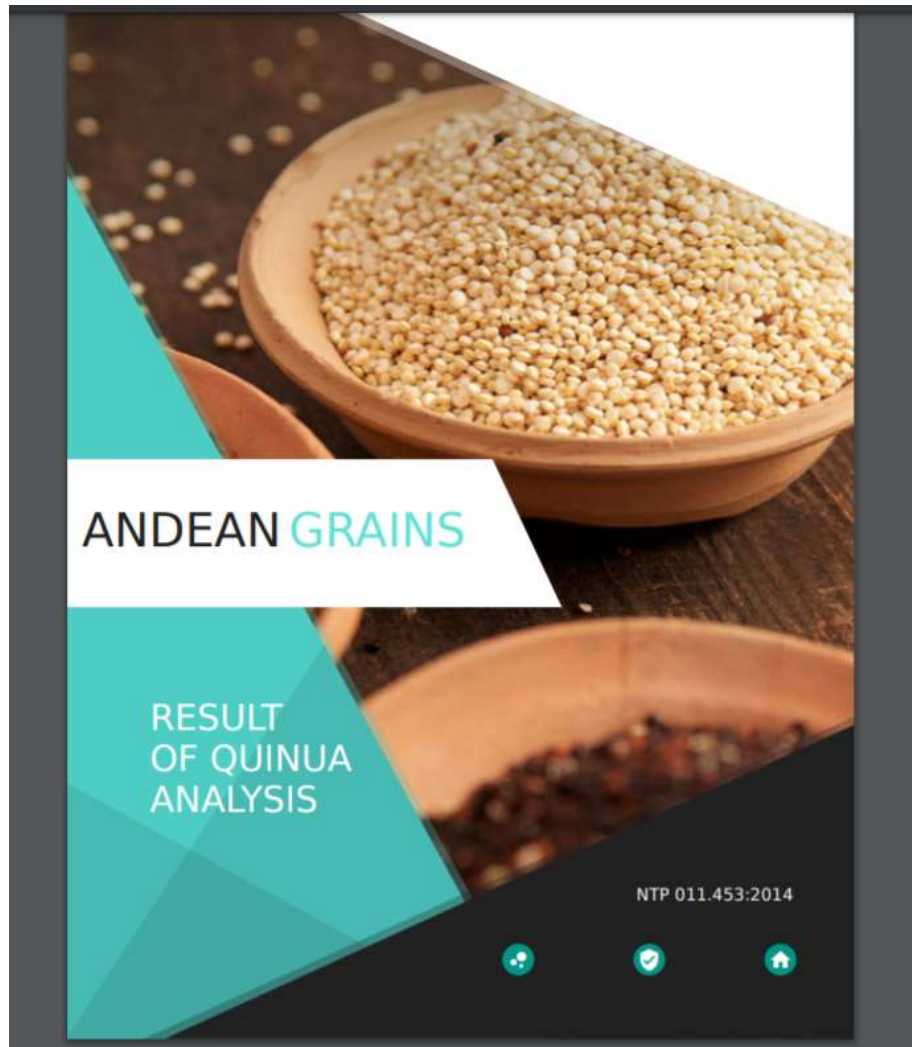


Figura 34. Reporte del análisis de la muestra de los granos de quinua.

Fuente: Elaboración propia.

El reporte del análisis de la muestra de los granos de quinua se exporta en formato pdf. La figura 34 muestra la portada del reporte.

CAPÍTULO IV. Resultados y Discusión

4.1. Resultado N° 1: Formar un banco de imágenes de muestras de quinua.

Se obtuvo quinua procesada en las ciudades de la región Puno (Cabana, Cabanillas, Lampa) para crear el banco de imágenes adquiriendo 1 kg. De distintas variedades como: la quinua perlada, quinua orgánica, quinua blanca y quinua roja.

Las imágenes digitales de granos de quinua se hicieron con una cámara microscópica de 500xp de con una limitada variedad de granos de quinua en estado procesado y no procesado. Obteniendo un banco de imágenes organizado.



Figura 35. Organización de folders.

Fuente: Elaboración propia.

En la figura 35 se muestra la organización de granos de quinua según la clasificación de la NTP 205.062.2014, donde cada carpeta cuenta con 1000 imágenes de 640 x 480 pixeles.



Figura 36. Grano de quinua.

Fuente: Elaboración propia.

En la figura 36 se muestra la imagen digital de un grano de quinua, en lo que definimos imagen sin procesar donde posteriormente se procesó la imagen obteniendo una imagen procesada para el reentrenamiento de Inception como se muestra en la figura 37.

4.2. Resultado N° 2: Identificar características de la quinua y sus impurezas para la clasificación

Para la identificación de características se procesó cada imagen de cada etiqueta (label) recortando áreas que no son necesarias para obtener las características del grano de quinua.



Figura 37. Imagen digital procesada de grano de quinua.

Fuente: Elaboración propia.

La imagen procesada es de 56 x 56 píxeles como se presenta en la figura 37. Obteniendo un mejor acercamiento de la característica de cada grano.

El procesamiento de las imágenes de granos de quinua según la primera organización de carpetas mostradas en la figura 35 se reestructuro para la implementación web, obteniendo un banco de imágenes con la estructura que presenta la figura 38 las etiquetas representan el nombre de cada carpeta, donde cada grano está identificado con las características indicadas de acuerdo a la NTP 205.062:2014.



Figura 38. Organización de carpetas para inception.

Fuente: Elaboración propia.

Con la organización de las carpetas mostradas en la figura 38 se obtiene una clasificación de granos de quinua aptos para el reentrenamiento de Inception.

4.3. Resultado N° 3: Modelar, entrenar y validar el modelo computacional

Se usó el modelo Inception, donde se hizo el entrenamiento con imágenes de 56 x 56 pixeles donde Inception cuenta con 48 capas de profundidad y parametrizándolo con 500 pasos. Con 7000 imágenes de quinua para su posterior entrenamiento.

El entrenamiento estima 1 hora con 40 min de acuerdo al banco de 7000 imágenes, con el uso de la metodología para la ciencia de datos se realizó una matriz de confusión para modelar, entrenar y evaluar el modelo computacional Inception.

Obteniéndose en la clasificación una predicción de en promedio del 99.1 % con 50 imágenes diferentes de quinuas. Identificándolas con sus respectivas etiquetas.

Tabla 13. Resultado de entrenamiento de Inception.

N°	Etiqueta	Precisión (%)
1	whole_grain	0.99910432
2	whole_grain	0.99744034
3	whole_grain	0.99969089
4	whole_grain	0.99992502
5	coated_grain	0.9999727
6	whole_grain	0.99975675
7	whole_grain	0.99946576
8	whole_grain	0.99973589
9	coated_grain	0.9999733
10	whole_grain	0.9999311
11	whole_grain	0.99543154
12	whole_grain	0.99996984
13	broken_grain	0.99833435
14	coated_grain	0.9966858
15	whole_grain	0.9961772
16	whole_grain	0.99980611
17	whole_grain	0.99957961
18	whole_grain	0.9998883
19	whole_grain	0.94624907
20	coated_grain	0.9868722
21	whole_grain	0.99997318
22	whole_grain	0.99940503
23	whole_grain	0.99981481
24	whole_grain	0.99993622
25	whole_grain	0.99985349
26	coated_grain	0.99993026
27	whole_grain	0.99932146
28	whole_grain	0.9944883
29	broken_grain	0.9999733
30	coated_grain	0.9999311
31	whole_grain	0.99543154
32	whole_grain	0.99996984
33	whole_grain	0.99833435
34	whole_grain	0.9966858
35	whole_grain	0.9961772
36	coated_grain	0.99980611
37	no_detect	0.505864
38	whole_grain	0.9998883
39	whole_grain	0.94624907
40	whole_grain	0.9868722

41	coated_grain	0.99997318
42	whole_grain	0.99940503
43	whole_grain	0.99981481
44	whole_grain	0.99993622
45	coated_grain	0.99985349
46	whole_grain	0.99993026
47	whole_grain	0.99932146
48	broken_grain	0.9944883
49	coated_grain	0.9999727
50	whole_grain	0.99975675

Fuente: Elaboración propia.

Según la tabla 13 se muestra una predicción con una tasa alta de 99.1% en promedio y el 0.09 % no se detectó previsto como un error donde se le presento al modelo otro una imagen de quinua pequeña de 20 x 20 pixeles, resaltando algo importante que se hizo solo 50 pruebas donde resulta un 99.1% dando a conocer que el número de pruebas haciendo el porcentaje de predicción podría disminuir hasta un 87 % que es el margen máximo margen de error que Inception ha tenido. Como se detalla es un error por debajo del 1% lo donde Inception se reafirma como un modelo de las CNN adecuado para la clasificación de quinua mencionando que la versión es InceptionV3.

El modelo Inception no tendrá que reentrenarse una y otra vez, solo si hay nuevas variedades de quinua que debe aprender.

4.4. Resultado N° 4: Implementar el modelo computacional en una aplicación web

La implementación del modelo Inception con la aplicación web se muestra el proceso de una imagen en la figura 39.

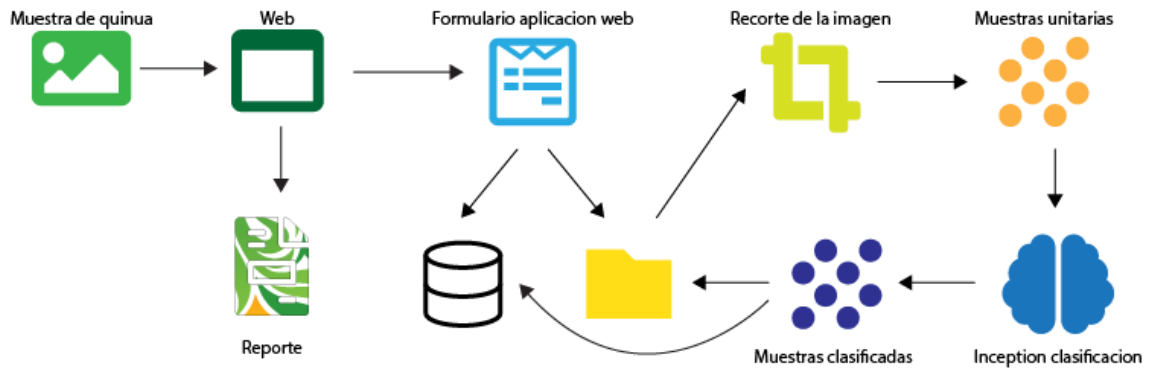


Figura 39. Modelo de la implementación completa.

Fuente: Elaboración propia.

Se definió un módulo independiente dentro de la aplicación web para integrar Inception obteniendo un buen resultado estimado de tiempo de 15 o 45 segundos dependiendo de la muestra de entrada, a la vez la aplicación web obtiene las características necesarias para la clasificación de quinua de acuerdo a la NTP 205.026:2014 que se ve en la figura 40.



Figura 40. Análisis detallado de quinua por grano.

Fuente: Elaboración propia.

Tabla 14. Etiquetas estándar de los granos andinos.

Parameters	Unit	Percent (%)
Whole grain	%	50.0
Broken grain	%	0.00
Damaged grain	%	0.00
Sprouted grain	%	0.00
Coated grain	%	50.0
Immature grain	%	0.00
Contrasting grain	%	0.00

Fuente: Elaboración propia.

En la tabla 14 se presenta el resultado obtenido en porcentajes de una muestra de granos de quinua, donde según a las etiquetas estándar se obtiene un porcentaje total de toda la muestra. Identificando el porcentaje de tipos de granos de quinua.

Tabla 15. Tamaño estándar de granos andinos.

Grand Size	Unit	Percent (%)
Big (firts)	%	99.1
Medium (second)	%	0.00
Small (third)	%	0.00

Fuente: Elaboración propia.

El tamaño de granos es otro estándar donde se determina a que grado pertenece el grano de quinua según la muestra realizando una segunda clasificación manejado por la NTP como se muestra en la tabla 15. Dando esto como un reporte completo de la muestra ingresada y la cantidad de granos de quinua encontrados en la muestra.

CAPÍTULO V. Conclusiones y recomendaciones

5.1. Conclusiones

PRIMERA: La clasificación de los granos de quinua obtenidas con la cámara microscópica, utilizó un medio controlado por un fondo blanco un tubo de pvc de 5 cm de diámetro en donde la cámara estuvo elevada a 6 cm de altura desde la superficie ayudando a obtener una imagen con alta resolución haciendo visibles las texturas y defectos físicos de cada grano de quinua obtenido durante el proceso de formación de banco de imágenes. Además de aplicar las tolerancias y clasificación del tamaño por la NTP 205.026:2014, donde se obtuvo 99.1% de precisión de 50 imágenes evaluadas por el modelo computacional Inception previamente entrenado.

SEGUNDA: El modelo computacional Inception ganador del desafío de clasificación de imágenes ILSVRC2014 presentó una tasa de error significativamente baja con sus competidores, perteneciente a la arquitectura de las redes neuronales convolucionales, con un rendimiento alto, de esta manera cuando recibe el entrenamiento no necesita volverse a entrenar hasta una previa actualización del banco de imágenes reduciendo el tiempo de la clasificación de las imágenes muestra a clasificar. Obteniendo una buena predicción por cada grano de quinua.

TERCERA: Utilizar la metodología AUP, permitió el desarrollo de la integración de la aplicación web con el modelo computacional Inception, proporcionando al modelo computacional una interfaz gráfica, en donde en gran parte de investigaciones solo se realiza el entrenamiento y la evaluación del modelo computacional donde la entrada de datos es mediante una consola. El desarrollo de la interfaz gráfica realizado con Django y AUP dejan un proyecto escalable para una futura evolución de la aplicación web.

CUARTA: El desarrollo de una aplicación web específicamente para analizar los granos de quinua considero imprescindible no solo para la exportación si no para conocer a detalle este grano andino y fortificar su producción de acuerdo al análisis que puede hacer la aplicación web creando una retroalimentación.

5.2. Recomendaciones

Se recomienda actualizar la versión del modelo Inception, su creador Google mejora cada el modelo cada cierto tiempo lanzando una nueva versión mejorada, este proyecto inicio con la versión Inceptionv1 y en el transcurso del desarrollo empezó a cambiar de versión en la documentación oficial se puede ver los cambios nuevos y las mejoras de este modelo.

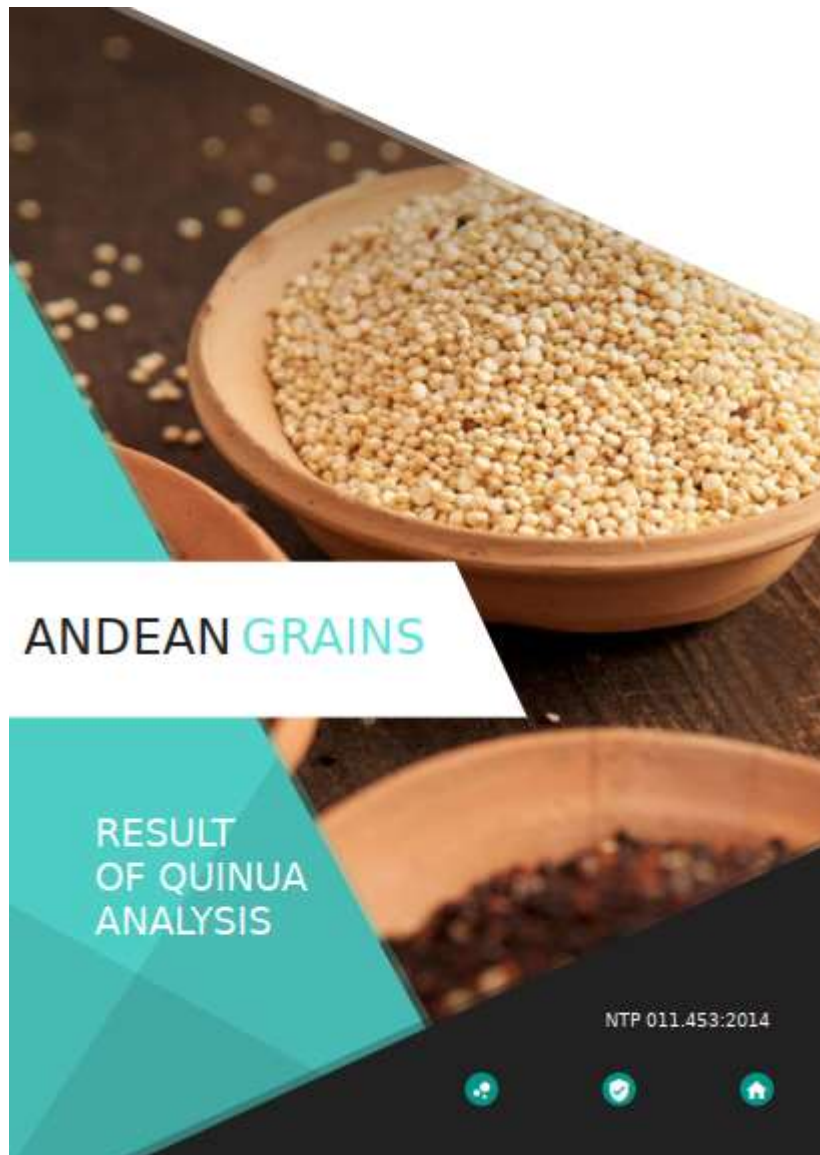
Para obtener una imagen de granos de quinua se recomienda utilizar una cámara microscópica de 1600X, utilizada a lo largo de la investigación. De esta manera se obtendrá una mejor respuesta de parte de la aplicación web, así como la clasificación cuando se analice la muestra de granos de quinua.

REFERENCIAS BIBLIOGRAFICAS

- Aditya Sharma, V. R. (2019). *Machine Learning for OpenCV 4 Second Edition*. Birmingham: Packt Publishing Ltd.
- Ambler, S. (2002). *Agile Modeling: Effective Practices for eXtreme Programming and the Unified Process*. Wiley.
- Beyeler, M. (2017). *Machine Learning for OpenC*. Packt Publishing.
- Beysolow, T. (2017). *Introduction to Deep Learning Using R*. San Francisco: Apress.
- Bonaccorso, G. (2017). *Machine Learning Algorithms*. Packt Publishing.
- Briega, R. E. (2018). IAAR. Obtenido de <https://iaarbook.github.io/>
- Cáceres, J. P. (2014). *Software automático de reconocimiento de patrones para la discriminación de granos de arroz pulido basado en procesamiento de imágenes digitales*.
- CECAGRO. (2016). *Sistema de acopio de quinua*.
- Fidias, A. (2017). *Efectividad y eficiencia de la investigación tecnológica en*. Caracas. INDECOPI. (2014). *GRANOS ANDINOS Quinua Requisitos*.
- Instituto Nacional de Estadística e Informática. (2014). *VOLUMEN EXPORTADO DE QUINUA*.
- Kamiya Matsuoka, N., Rida Gamero, N. L., & Torres Artica, P. E. (2013). *PLAN DE EXPORTACIÓN DE QUINUA Y DERIVADOS CON DESTINO A ESTADOS UNIDOS*. Lima.
- Ketkar, N. (2017). *Deep Learning with Python*. Bangalore: Apress.
- Maza, G. V. (2017). *Procesamiento de imágenes usando OPENCV aplicado en RASPBERRY PI para la clasificación del cacao*.
- Melé, A. (2020). *Django 3 By Example Third Edition*. Mumbai: Packt Publishing Ltd.
- Ministerio de Agricultura y Riego - MINAGRI. (2017). Puno.
- Ministerio de Agricultura y Riego. (2019). *Perú se consolida como primer exportador de quinua*. Lima.
- Núñez Dorantes, R. J. (2017). *Sistema de visión artificial y control para un microlaboratorio a bordo de un nanosatélite*. Mexico.
- Perez Miranda, S. C., & Rosell Llanos, L. A. (2016). *SELECCIÓN DE CARACTERISTICAS DE INTERÉS EN LA CLASIFICACIÓN DE GRANOS DE CAFÉ MEDIANTE UN SISTEMA DE VISIÓN POR COMPUTADORA*. Cajamarca.
- Pramod Singh, A. M. (2020). *Learn TensorFlow 2.0 (Implement Machine Learning and Deep Learning Models with Python)*. India: Apress.
- Ramar, S. (2019). *Artificial Intelligence How It Changes the Future*.
- Rodríguez, A. O. (2017). *Desarrollo de una aplicación de reconocimiento en imágenes utilizando Deep Learning con OpenCV*.
- Rollins, J. B. (2015). *Metodología Fundamental para la Ciencia de Datos*. IBM Analytics.
- Soto, J. L., Valdivia, R., & Solano, C. (2019). *ESTANDARIZANDO LA CALIDAD DE LOS GRANOS ANDINOS NORMAS TECNICAS PERUANAS*. Puno.
- Villán, A. F. (2019). *Mastering OpenCV 4 with Python*. Packt Publishing Ltd.
- Yuanyuan Wang, C. W. (2016). Ship Classification in High-Resolution SAR Images Using Deep Learning of Small Datasets. *Sensors MPI*, 15.

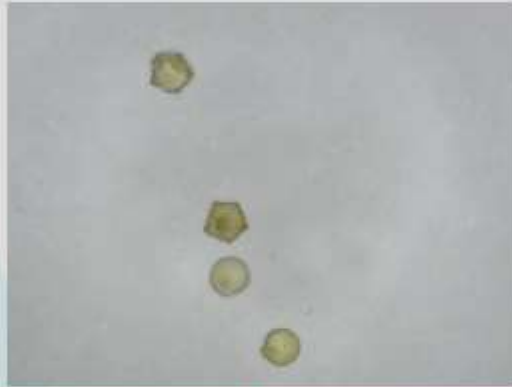
ANEXOS

Anexo A. Reporte de análisis de la muestra de granos de quinua en formato pdf.



Information General

SAMPLE IMAGE



Details	to deta
Description	Do commended an shameless we disposing do. Indulgence ten remarkably not are impression out. Power is lived means oh every in we quiet.
Date	to deta

Grain Detail



General Result

PARAMETERS	UNIT	PERCENT
Whole grain	%	50.0
Broken grain	%	0.0
Damaged grain	%	0.0
Sprouted grain	%	0.0
Coated grain	%	50.0
Immature grain	%	0.0
Contasting grain	%	0.0

GRAIN SIZE	UNIT	PERCENT
BIG (first)	%	99.1
MEDIUM (second)	%	00.0
SMALL (third)	%	00.0