Especificación de Requerimientos de Software

Aplicación web para la clasificación de cáncer de mama usando Deep Learning.

Integrantes:

De la vega hernández lucia iveth

ruiz correa matias fernando

zaPATA ARTEAGA jorge ulises

Contenido

[1. Introducción. 4](#_Toc21455730)

[1.1 Propósito. 4](#_Toc21455731)

[1.2 Alcance. 4](#_Toc21455732)

[1.3 Definiciones, acrónimos y abreviaturas. 4](#_Toc21455733)

[1.4 Referencias. 4](#_Toc21455734)

[1.5 Vista general. 5](#_Toc21455735)

[2. Descripción General. 5](#_Toc21455736)

[2.1. Perspectiva del producto. 5](#_Toc21455737)

[2.2. Funcionalidad del producto. 6](#_Toc21455738)

[2.3. Características del usuario. 6](#_Toc21455739)

[2.4. Restricciones generales. 7](#_Toc21455740)

[2.5. Presunciones y dependencias. 7](#_Toc21455741)

[3. Especificación de requerimientos. 8](#_Toc21455742)

[3.1. Requerimientos Funcionales. 9](#_Toc21455743)

[3.2. Requerimientos de desempeño. 9](#_Toc21455744)

[3.3. Requerimientos de la base de datos lógica. 10](#_Toc21455745)

[3.4. Restricciones de diseño. 10](#_Toc21455746)

[a. Cumplimiento de estándares. 11](#_Toc21455747)

[3.5. Atributos 11](#_Toc21455748)

[a. Confiabilidad. 11](#_Toc21455749)

[b. Disponibilidad. 11](#_Toc21455750)

[c. Seguridad. 11](#_Toc21455751)

[d. Mantenibilidad. 12](#_Toc21455752)

[e. Portabilidad. 12](#_Toc21455753)

[3.6. Requisitos de Interfaces externas. 12](#_Toc21455754)

[3.7. Organización específica de los requerimientos. 13](#_Toc21455755)

[a. Modo del sistema. 13](#_Toc21455756)

[b. Clases de usuario. 13](#_Toc21455757)

[c. Objetos. 13](#_Toc21455758)

[d. Característica. 13](#_Toc21455759)

[e. Estímulos (Eventos). 14](#_Toc21455760)

[f. Respuesta. 14](#_Toc21455761)

[g. Jerarquía funcional. 14](#_Toc21455762)

[h. Cometarios adicionales. 14](#_Toc21455763)

[Anexo 1. Plantillas de organización de SRS. 15](#_Toc21455764)

[Por modo del sistema 15](#_Toc21455765)

[opción 1. 15](#_Toc21455766)

[opción 2 16](#_Toc21455767)

[Por clases de usuario. 16](#_Toc21455768)

[Por objeto. 17](#_Toc21455769)

[Por característica. 18](#_Toc21455770)

[Por estímulo (evento) 19](#_Toc21455771)

[Por jerarquía funcional 20](#_Toc21455772)

[Por combinación de diferentes organizaciones 23](#_Toc21455773)

# Introducción.

Este documento tiene como finalidad servir de guía tanto para el equipo de desarrollo como para los interesados en el proyecto, proporcionando una visión clara de los objetivos, el alcance y los beneficios esperados del sistema. Asimismo, busca establecer una base sólida que facilite la planeación, implementación, pruebas y futura evolución del software:

## 1.1 Propósito.

El documento presente tiene como principal objetivo el dar a conocer las características y funcionamiento del proyecto “*Aplicación web para la clasificación de cáncer de mama usando Deep Learning*”, dirigido primordialmente al equipo de desarrollo del proyecto.

## 1.2 Alcance.

Se presenta una descripción general de aplicación y el alcance del proyecto de software “*Aplicación web para la clasificación de cáncer de mama usando Deep Learning”*, estableciendo las funcionalidades y limitaciones principales que incluirá y los beneficios que ofrecerá:

* Funcionalidades:
  + Analizar imágenes para la clasificación de cáncer de mama utilizando un modelo de Deep Learning previamente entrenado.
  + Carga de archivos (imágenes) en formato JPG y DICOM para su clasificación.
  + Presentación de resultados en el sitio web con los siguientes datos:
    - Clasificación de clase (benigno o malgino).
    - Porcentaje de clasificación.
  + Preprocesamiento automático de imágenes (eliminación de ruido, ajuste de contraste, normalización de tamaño).
  + Soporte multimodal: análisis de las dos vistas estándar de mamografía (CC y MLO) para mejorar la precisión.
* Limitaciones:
  + No realizará diagnóstico definitivo, solo servirá como apoyo basado en IA.
  + El modelo se entregará ya entrenado, sin posibilidad de reentrenamiento por parte del usuario.
  + El uso está limitado a imágenes en formato estándar (JPG y DICOM), no se aceptarán fotografías clínicas ni ultrasonidos.
  + No se almacenará información personal del paciente, ni se desarrollará un expediente clínico.
  + Uso exclusivo en dispositivos de cómputo.

Además, a continuación se describen los beneficios con los que contará el software:

* Apoyo de diagnóstico para el usuario visualizando los resultados de la clasificación para facilitar su interpretación.
* Accesible desde cualquier dispositivo de cómputo con conexión a internet.
* Sin costo de utilización.
* Interfaz intuitiva.

## 1.3 Definiciones, acrónimos y abreviaturas.

## *Inteligencia Artificial/ Machine Learning.*

## IA: Es la capacidad de las máquinas para usar algoritmos, aprender de los datos y utilizar lo aprendido en la toma de decisiones tal y como lo haría un ser humano.

## Deep Learning: Se trata de un subcampo del aprendizaje automático que se utiliza para resolver problemas muy complejos y que normalmente implican grandes cantidades de datos. El aprendizaje profundo se produce mediante el uso de redes neuronales, que se organizan en capas para reconocer relaciones y patrones complejos en los datos [1].

## CNN (Convolutional Neural Network) : Las redes neuronales convolucionales son redes neuronales artificiales que utilizan la operación de convolución en al menos una de sus capas. Donde se combinan los datos de entrada con un filtro para extraer únicamente la información útil [2]. Estándares de software médico.

## ISO 13485: Es una norma de control de calidad creada con el propósito de mejorar los requerimientos de los dispositivos de manufactura médicos. Garantizando la seguridad, eficacia y el cumplimiento de las normas durante todo el ciclo de vida del producto desarrollado [3].

## *Procesamiento de imágenes médicas.*

## DICOM (Digital Imaging and Communication in Medicine): Es un formato estándar de archivo de imagen usado por los dispositivos de hardware radiológicos [4].

## Normalización: Es un método del procesamiento de imágenes que consiste en llevar a un tamaño estándar las dimensiones de una imagen, sin provocar en ella alguna distorsión de importancia [5].

## Escalado: Es el proceso de cambiar el tamaño de una imagen digital, ya sea para hacerla más grande o más pequeña, multiplicando su anchura y altura por un valor específico.

## Segmentación: Es un proceso que consiste en dividir una imagen en regiones homogéneas con respecto a una o más características con el fin de facilitar su posterior análisis y reconocimiento. Puede verse como un proceso que, a partir de una imagen, produce otra en la que cada píxel tiene asociada una etiqueta distintiva del objeto al que pertenece [6].

## Filtrado: El filtrado de imágenes es una técnica utilizada en el procesamiento digital de imágenes que consiste en modificar los valores de los píxeles de una imagen a partir de una operación matemática, con el fin de resaltar o atenuar ciertas características.

## Extracción de características: Es una técnica que consiste en transformar los datos de entrada en un conjunto reducido de descriptores que conservan la información más relevante para una tarea de análisis [7].

## *Validación y evaluación de modelos.*

## ROC: La curva ROC es una herramienta estadística que se utiliza para evaluar la capacidad discriminativa de una prueba diagnóstica dicotómica. En otras palabras, es una herramienta que permite evaluar el rendimiento del modelo a través de una clasificación. binaria.

## AUC: Es el área bajo la curva ROC que mide la facultad discriminativa del test, es decir, su capacidad numérica de clasificar[7].

## CAD: Es un sistema informático que automatiza el proceso de diseño de algún tipo de ente físico en 2D o en 3D.

## CADe: Se refiere a sistemas que ayudan al radiólogo a señalar regiones sospechosas en imágenes médicas que podrían representar posibles anomalías. Estos sistemas no emiten un diagnóstico final, sino que sirven como una “segunda opinión” que incrementa la sensibilidad del especialista [8].

## CADx: Es un sistema que permite detectar regiones de interés, clasificar o evaluar la probabilidad de que una región pertenezca a una clase en concreto. Su objetivo es ayudar en la toma de decisiones clínicas, reduciendo falsos positivos y apoyando el diagnóstico diferencial [9].

## Modelos de clasificación de imágenes(ResNet, Inception).

## *Base de datos.*

## DDSM: El DDSM (Base de Datos Digital para el Cribado de Mamografía) es una base de datos con 2,620 estudios de mamografía en película escaneada, que incluyen casos normales, benignos y malignos con información patológica verificada.

## CBIS-DDSM: Es una versión actualizada de la base de datos DDSM que ofrece datos de fácil acceso y una segmentación mejorada de las regiones de interés (ROI). Incluye imágenes descomprimidas, selección y curación de datos por mamógrafos capacitados, segmentación actualizada de masas y cuadros delimitadores, así como diagnóstico patológico para los datos de entrenamiento, formateados de manera similar a los conjuntos de datos modernos de visión por computadora [10].

## MLO (vista oblicuo medio lateral): Es una proyección mamografía realizada en un ángulo de 45° aunque puede variar entre 40° y 60° según la anatomía de la paciente. Esta vista muestra una parte del músculo pectoral, que permite evaluar el tejido mamario y localizar lesiones en los cuadrantes superiores e inferiores de la mama.

## CC (Craneocaudal): Es una proyección mamográfica que recoge la mayor parte de la mama en una vista de arriba hacia abajo, permitiendo la localización de lesiones en los cuadrantes interno (inferior en la placa) y externo (superior en la placa) de la mama [11].

## *Desarrollo web.*

## Arquitectura cliente-servidor: Es una arquitectura distribuida en el que las funciones y responsabilidades se dividen en dos entidades principales cliente y servidor. En el modelo cliente-servidor, el cliente envía un mensaje solicitando un determinado servicio a un servidor que envía uno o varios mensajes con la respuesta o el servicio. En este modelo, el cliente se encarga de la interfaz de usuario y la interacción directa, mientras que el servidor gestiona el procesamiento central, la lógica de negocio y el acceso a la base de datos [12].

## HTTPS:El protocolo de transferencia de hipertexto seguro (HTTPS) es la versión segura de HTTP. El protocolo HTTPS está encriptado para aumentar la seguridad de las transferencias de datos. Esto es especialmente importante cuando los usuarios transmiten datos confidenciales[13].

## 1.4 Referencias.

## *[1]. INTELIGENCIA ARTIFICIAL 101 COSAS QUE DEBES SABER HOY SOBRE NUESTRO FUTURO (1.a ed.). (2018). Alienta.* [*https://planetadelibrosar0.cdnstatics.com/libros\_contenido\_extra/40/39307\_Inteligencia\_artif icial.pdf*](https://planetadelibrosar0.cdnstatics.com/libros_contenido_extra/40/39307_Inteligencia_artif%20icial.pdf)*.*

## *[2]. REDES NEURONALES CONVOLUCIONALES Y APLICACIONES. (2020). [Universidad Computense de Madrid]. https://docta.ucm.es/rest/api/core/bitstreams/27b92c22-b95a-4992-8408-2962a474bd1e/conte nt. [3].Shanmugam, P. S. T., Thangaraju, P., Palani, N., & Sampath, T. (s. f.). Medical Device Guidelines and Regulations Handbook. Springer Nature.*

## *[4]. DICOM Demystified: A review of digital file formats and their use in radiological practice. (2005, noviembre). https://www.sciencedirect.com/science/article/pii/S0009926005002199). ScienceDirect.*

## *[5]. Castellano, E. G., Alonso, E. C., Mata, F. J. S., Díaz, Y. M., & Rivero, R. S. (s. f.). Normalización de imágenes de orejas para el Reconocimiento Biométrico. Scielo, 40(3), 28-38.*

## *[6]. Algoritmo de segmentación de imágenes basado en información de color y textura. (s. f.). Recuperado 11 de septiembre de 2025, de https://biblus.us.es/bibing/proyectos/use/abreproy/10799/fichero/memoria%252FCap%C3%A Dtulo3.pdf.*

## *[7]. R. C. Gonzalez and R. E. Woods, Digital Image Processing, 4th ed. Pearson, 2018.*

## *[8]. La curva ROC. (2023). Medicina de Familia SEMERGEN, 49.* [*https://www.sciencedirect.com/science/article/abs/pii/S1138359322001952?via%3Dihub*](https://www.sciencedirect.com/science/article/abs/pii/S1138359322001952?via%3Dihub)*.*

## *[9]. K. Doi, "Computer-aided diagnosis in medical imaging: historical review, current status and future potential," Computerized Medical Imaging and Graphics, vol. 31, no. 4–5, pp. 198–211, 2007.*

## *[10]. R. S. Lee, F. Gimenez, A. Hoogi, K. K. Miyake, M. Gorovoy, y D. L. Rubin, «A curated mammography data set for use in computer-aided detection and diagnosis research», Scientific Data, vol. 4, n.o 1, Art. n.o 1, dic. 2017, doi: 10.1038/sdata.2017.177.*

## *[11]. A. J. Bekker, H. Greenspan, y J. Goldberger, «A multi-view deep learning architecture for classification of breast microcalcifications», en \_2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI)\_, IEEE, abr. [10.1109/ISBI.2016.7493369](https://doi.org/10.1109/ISBI.2016.7493369). 2016, pp.*

## *[12].A. S. Tanenbaum and M. Van Steen, Distributed Systems: Principles and Paradigms, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2007.*

## *[13]. HTTP y HTTPS: diferencia entre los protocolos de transferencia. AWS. (s. f.). Amazon Web Services, Inc.* [*https://aws.amazon.com/es/compare/the-difference-between-https-and-http/*](https://aws.amazon.com/es/compare/the-difference-between-https-and-http/).

*[14]. He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep Residual Learning for Image Recognition (ResNet). arXiv:1512.03385 https://arxiv.org/abs/1512.03385*

*[15]. Y. S. Leong, K. Hasikin, K. W. Lai, N. M. Zain, and M. M. Azizan, “Microcalcification Discrimination in mammography using deep convolutional neural network: Towards rapid and early breast cancer diagnosis,” Frontiers in Public Health, vol. 10, Apr. 2022, doi: 10.3389/fpubh.2022.875305. https://pmc.ncbi.nlm.nih.gov/articles/PMC9096221/*

*[16]. Tan, M., & Le, Q. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. arXiv:1905.11946 https://arxiv.org/abs/1905.11946*

*[17]. D. G. P. Petrini, C. Shimizu, R. A. Roela, G. V. Valente, M. A. A. K. Folgueira, and H. Y. Kim, “Breast cancer diagnosis in Two-View mammography using End-to-End trained EfficientNeT-Based convolutional network,” IEEE Access, vol. 10, pp. 77723–77731, Jan. 2022, doi: 10.1109/access.2022.3193250. https://arxiv.org/abs/2110.01606*

*[18]. M. A. Al-masni et al., «Simultaneous detection and classification of breast masses in digital mammograms via a deep learning YOLO-based CAD system», Computer Methods and Programs in Biomedicine, vol. 157, pp. 85-94, abr. 2018, doi: 10.1016/j.cmpb.2018.01.017.*

*[19]. Gómez, A. B. (s. f.). Investigación y desarrollo de metodología avanzada de segmentación de la médula espinal cervical a partir de imágenes RM para la ayuda al diagnóstico y seguimiento de pacientes de Esclerosis Múltiple. https://doi.org/10.4995/thesis/10251/205742.*

*[20]. Ultralytics. (2025, 30 marzo). Ultralytics YOLO11 Modes. https://docs.ultralytics.com/es/modes/#introduction.*

*[21]. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. IEEE Xplore, 2818-2826. https://doi.org/10.1109/cvpr.2016.308*

*[22]. Chelloug, S. A., Mahel, A. S. B., Alnashwan, R., Rafiq, A., Muthanna, M. S. A., & Aziz, A. (2025). Enhanced breast cancer diagnosis using modified InceptionNet-V3: a deep learning approach for ultrasound image classification. Frontiers In Physiology, 16. https://doi.org/10.3389/fphys.2025.1558001.*

*[23]. Team, K. (s. f.). Keras documentation: InceptionV3. https://keras.io/2/api/applications/inceptionv3/*

*[24] Alzubaidi, L., Zhang, J., Humaidi, A.J. et al. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. J Big Data 8, 53 (2021). https://doi.org/10.1186/s40537-021-00444-8.*

*[25] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” arXiv preprint arXiv:1409.1556, 2015.*

## 1.5 Vista general.

En esta sección se describe el contenido del resto del SRS, así como su organización.

# Descripción General.

En esta sección se describirán de manera general los factores que afecten el producto y sus requerimientos. Provee un soporte (background) de los requerimientos que se detallan en la sección 3 de este documento, y que permiten comprenderlos de manera sencilla. Por lo general esta sección contiene las siguientes subsecciones:

## Perspectiva del producto.

El desarrollo del sistema, fundamentado en la implementación de modelos de Deep Learning, se apoya en arquitecturas previamente utilizadas que han demostrado altos niveles de precisión en el ámbito médico y en el reconocimiento de imágenes. A continuación, se presentan los principales modelos considerados, los cuales servirán como base para realizar una elección fundamentada y adecuada para la implementación del proyecto.

### ***ResNet 50.***

ResNet (*Residual Network*) fue introducido por He et al. (2015) para resolver el problema del *vanishing gradient*. Su característica principal son los **bloques residuales**, que introducen conexiones de salto (*skip connections*), permitiendo que la red aprenda una función de residuo (F(x)+x) en lugar de una transformación directa. Esto facilita el entrenamiento de redes muy profundas [14].

ResNet50 es una versión con 50 capas (convolucionales y de *pooling*) organizadas en bloques tipo *bottleneck*, los cuales reducen y expanden dimensiones para mejorar la eficiencia [14]. Según *Y. S. Leong et. Al [15]*, este modelo ha demostrado ser efectivo en tareas de clasificación de microcalcificaciones en mamografías logrando una precisión del 97.58% respecto a otros modelos como ResNet34 (97.35%) y VGG16 (96.97%), confirmando su aplicabilidad en el diagnóstico asistido de cáncer de mama.

#### Parámetros y características técnicas del modelo.

* Número de parámetros: ~25 millones [14]
* Datos de entrada: imágenes RGB o escala de grises de 224×224 píxeles.
* Salida: vector de probabilidades de clase (benigno/maligno en nuestro caso).
* Ventaja técnica: puede aprovechar transfer learning con pesos preentrenados en ImageNet[14].
* Función de activación: ReLU en capas ocultas; Softmax en la capa final.
* Optimizadores comunes: SGD con momentum, Adam.

#### Ventajas y limitaciones.

##### Ventajas:

* Preciso y probado en múltiples dominios[14].
* Compatible con datasets grandes y modelos transferidos[15].

##### Limitaciones:

* Pesado (25M parámetros). Requiere GPU para entrenar [15].
* Puede ser menos eficiente para despliegue en sistemas con recursos limitados.

### ***EfficientNet 50.***

EfficientNet fue introducido por Tan & Le (2019), quienes propusieron el concepto de compound scaling, que escala profundidad, anchura y resolución de forma equilibrada en lugar de solo una dimensión [16]. Esta arquitectura utiliza bloques MBConv (*Mobile Inverted Bottleneck Convolution*), derivados de MobileNet, con convoluciones separables en profundidad que reducen el número de operaciones. Además, emplea squeeze-and-excitation (SE) blocks, que reponderan los canales para mejorar la extracción de características [16].

#### Compound Scaling (Escalado compuesto):

Antes de EfficientNet, las arquitecturas proponían escalar una de 3 posibles propiedades importantes: más largo (más profundidad de capas); más ancho (más filtro de capas); más resolución (procesamiento de imágenes más grandes). Ahora EfficientNet escala y mejora las 3 características logrando un mejor rendimiento sin desperdiciar recursos [16].

#### Bloques MBConv (Mobile Inverted Bottleneck Convolution):

Estos bloques vienen de MobileNet y son como filtros de paso eficiente.

En lugar de procesar toda la información de golpe (lo cual cuesta mucho), primero comprimen la información, luego hacen las operaciones necesarias, y finalmente la expanden de nuevo.

Es como empaquetar un archivo pesado en .zip, procesarlo, y después descomprimirlo: ahorras tiempo y espacio.

*Squeeze-and-Excitation (SE) Blocks:*

Estos bloques actúan como un director de orquesta que decide qué instrumentos deben sonar más fuerte.

En términos técnicos: “reponderan” los canales, es decir, les asignan más o menos importancia dependiendo de cuáles son más útiles para la tarea de clasificación [16].

Así, el modelo no trata todos los canales igual, sino que prioriza los más relevantes para la imagen.

En el contexto de imágenes médicas, Petrini et al. [17] proponen un sistema de diagnóstico de cáncer de mama en mamografías que reemplaza arquitecturas previas como ResNet o VGG por EfficientNet, demostrando mejoras en desempeño. Sin embargo, a diferencia del trabajo de Tan & Le, el reescalado de las imágenes fue mayor, teniendo imágenes de 1152 × 896 pixels.

#### Parámetros y características técnicas.

* Versiones: EfficientNet-B0 a EfficientNet-B7 (de más pequeño a más grande).
* Parámetros:
  + B0: ~5 millones (muy ligero).
  + B7: ~66 millones (más potente).
* Entrada típica: imágenes a color o escala de grises de 224×224 (B0) hasta 600×600 (B7). En algunas literaturas se encuentran de  1152 × 896 [17].
* Salida: vector de probabilidades.
* Funciones de activación: Swish/SiLU (mejor que ReLU en aprendizaje profundo).
* Optimizadores comunes: RMSprop, Adam.

#### Ventajas y limitaciones.

##### Ventajas.

* Mejor relación rendimiento-parámetros (más ligero que ResNet) [16].
* Escalable: versiones pequeñas para móviles y grandes para servidores.
* Buen desempeño con datasets grandes como RSNA.
* Mayor desempeño con menos parámetros que ResNet o VGG, logrando AUC superiores en CBIS-DDSM [17]

##### Limitaciones:

* Más reciente: menos probado en ciertos entornos clínicos que ResNet.
* Las versiones grandes requieren hardware de alto rendimiento (B5–B7).
* Entrenamiento con imágenes de alta resolución demanda mayor memoria.

### ***YOLOv8.***

Entre las alternativas de modelos previamente entrenados se encuentra la arquitectura YOLO desarrollada por Ultralytics, reconocida por su efectividad en diversas aplicaciones de visión artificial, incluido el ámbito médico. Tal es el caso de la investigación realizada por Al masni et al [18], donde describe un sistema CAD completamente integrado para mamografías,  alcanzando una precisión de 99.7 % en la localización de masas, utilizando la base de datos CBIS-DDSM.

La arquitectura de YOLO está inspirada en el modelo GoogLeNet para la clasificación de imágenes, contando con 24 capas convolucionales seguidas de 2 capas completamente conectadas. Es decir, las primeras 24 capas convolucionales están enfocadas en la extracción de características, mientras que las siguientes 2 capas están enfocadas en la predicción y clasificación tomando como base la extracción de características previamente realizada.

Algunas versiones de YOLO requieren un preprocesamiento de imágenes específico debido a que, para segmentar las imágenes, requiere que las máscaras se encuentren en formato de etiquetas con la descripción de las coordenadas del área que se desea detectar.

Por lo tanto, el formato del conjunto de datos utilizado para entrenar los modelos de segmentación de YOLO es el siguiente:

* 1. Reescalado de imágenes: En principio es necesario reescalar las imágenes a un formato de 640x640 pixeles para ajustar de mejor manera los datos de entrada al modelo (YOLOv8).
  2. Obtención de coordenadas desde máscaras: Para utilizar YOLO, es necesario contar con un conjunto de coordenadas que definen las regiones de interés (ROI) en la imagen. Estas coordenadas indican la ubicación de los objetos de interés.
  3. Guardado en archivos de texto (.txt): Las coordenadas obtenidas se guardaron en archivos con extensión .txt, que posteriormente se usan como etiquetas para entrenar el modelo.
  4. Archivo YAML (.yaml): Ultralytics utiliza un formato de archivo YAML para definir el conjunto de datos y la configuración del modelo para entrenar modelos de detección[19].

De acuerdo con la documentación de Ultralytics [20], YOLO ha presentado una gran cantidad de versiones y mejoras. YOLOv11, la última versión del modelo, introduce mejoras significativas en la arquitectura y los métodos de entrenamiento, destacando algunos modos como:

* Modo de entrenamiento: Utilizado para entrenar un modelo con un conjunto de datos personalizado.
* Modo Val: Se utiliza para evaluar la precisión y rendimiento del modelo en general.
* Modo exportar; Se utiliza para exportar un modelo YOLO en un formato que permita la implementación en otras aplicaciones (TensorRT o ONNX).

Finalmente, es necesario destacar la fácil adaptación que presenta esta versión del modelo, pudiendo ser implementada en varios entornos como plataformas en la nube y sistemas que admiten GPU de NVIDIA*.*

### ***Inception V3.***

Inception V3 es una red neuronal convolucional diseñada para la clasificación de imágenes, destacando por su capacidad de expandir la profundidad del modelo sin aumentar el costo computacional y reducir significativamente la cantidad de datos de entrada.

De acuerdo con *Szegedy, C* [21] InceptionV3 está compuesto por 42 o 48 capas, dependiendo de la fuente. La red está diseñada para optimizar la eficiencia computacional y la precisión en el aprendizaje profundo costando solo 2.5 veces más que la versión 1 de Inception también conocida como GoogleNet. Este modelo V3 se caracteriza por aplicar varias ramas de convoluciones en paralelo (también llamados módulos inception), para posteriormente concatenar resultados con el objetivo de captar características sin aumentar el número de parámetros. Además, Inception es mucho más eficiente que versiones de VGGNet presentando resultados destacables en el ámbito del cáncer de mama, tal es el caso del modelo propuesto por Chelloug et al [22] obteniendo un 99.10% de exactitud utilizando una combinación de dos datasets con aproximadamente 850 mamografías.

De acuerdo con la documentación de Keras [23] los datos de entrada del model deben tener un tamaño por defecto: 299x299 pixeles con 3 canales RGB.

### ***CNN Model***

Una red neuronal convolucional (CNN) es un tipo especial de red neuronal diseñada para procesar datos con estructura de grilla, como imágenes (que son una grilla de píxeles).

Las CNN son muy buenas para reconocer patrones visuales, por ejemplo:

* Bordes y esquinas
* Texturas
* Formas y estructuras
* Objetos completos en una imagen

CNN model tiene una arquitectura basada en redes convolucionales utilizado para problemas de clasificación simple.

El modelo CNN\_model tiene varias partes:

1. Capa Convolucional (Conv2D): Usa filtros (matrices pequeñas, como 3x3) que recorren la imagen (operación de convolución).

* Cada filtro detecta un patrón específico: líneas horizontales, bordes, texturas.
* Se generan mapas de características (feature maps) que contienen la información detectada.

1. Capa de Submuestreo o Pooling (MaxPooling2D):Reduce el tamaño de los mapasde características.

* Conserva la información más importante (valores máximos).
* Hace que el modelo sea más eficiente y menos sensible a pequeños cambios en la imagen.

1. Más capas Convolucionales + Pooling.

* Cada capa aprende patrones más complejos.
* Al inicio detecta bordes → luego formas pequeñas → después estructuras completas.

1. Capa Flatten.

* Convierte la matriz en un vector plano.
* Prepara los datos para las capas densas (fully connected).

1. Capas Densas (Dense).

* Funcionan como un clasificador tradicional pero alimentado con características aprendidas.
* La última capa depende del problema:
  + Softmax: múltiples clases (ej. 10 dígitos en MNIST).
  + Sigmoid: clasificación binaria (ej. benigno vs maligno).

1. Entrenamiento.

* Optimizer (Adam): ajusta los pesos de la red.
* Loss function: mide el error entre la predicción y la realidad.
* Accuracy: mide el porcentaje de aciertos [24].

### ***VGG16.***

VGG16 es una red neuronal convolucional profunda propuesta por el grupo Visual Geometry Group (VGG) de la Universidad de Oxford, enfocándose en su estructura, funcionamiento y aplicaciones; presentada en 2014 en el ILSVRC (ImageNet Large Scale Visual Recognition Challenge). Introdujo la idea de que redes más profundas con filtros más pequeños (3×3) podían superar modelos anteriores como AlexNet.

#### Arquitectura.

* Input: imágenes de 224×224×3 píxeles.
* Consta de 16 capas entrenables (13 convolucionales + 3 densas).
* Estructura en bloques:
  + Bloque 1: 2 conv (64 filtros) + MaxPooling
  + Bloque 2: 2 conv (128 filtros) + MaxPooling
  + Bloque 3: 3 conv (256 filtros) + MaxPooling
  + Bloque 4: 3 conv (512 filtros) + MaxPooling
  + Bloque 5: 3 conv (512 filtros) + MaxPooling
* Parte final: Flatten + 2 capas densas (4096 neuronas) + capa softmax (1000 clases en ImageNet).

#### Características clave.

* Uso sistemático de filtros pequeños (3×3) en lugar de convoluciones grandes.
* Profundidad de 16 capas → captura características más complejas en cada nivel.
* Uniformidad en el diseño, lo que simplifica su implementación.

#### Ventajas.

* Mejor precisión que modelos previos (ej. AlexNet).
* Buen balance entre simplicidad y desempeño.
* Adecuado para transfer learning: se usa como base para otras aplicaciones.

#### Limitaciones.

* Modelo muy grande (~528 MB de pesos).
* Entrenamiento costoso en cómputo y memoria.
* Puede ser menos eficiente que arquitecturas más recientes (ej. ResNet).

#### Aplicaciones.

* Clasificación de imágenes generales (ImageNet).
* Adaptaciones médicas: detección de cáncer de mama, clasificación de tumores, segmentación de imágenes médicas.
* Reconocimiento facial, visión en robótica, sistemas de seguridad [24].

## Funcionalidad del producto.

El sistema tiene como objetivo principal la clasificación automática de lesiones en mamografías como benignas o malignas, mediante la integración de modelos de Deep Learning multimodales. La aplicación web permitirá la interacción con los usuarios, la gestión de imágenes médicas y la presentación de resultados.

Las principales funcionalidades del producto se detallan a continuación:

**Carga y administración de imágenes médicas**

* Subida de mamografías en formatos compatibles (DICOM, JPG).
* Validación automática de la integridad y formato de las imágenes cargadas.

**Preprocesamiento de imágenes**

* Normalización de resolución y calidad de las imágenes.
* Aplicación de técnicas de reducción de ruido y realce de contraste.
* Segmentación automática de la región de interés (ROI) en la mamografía.

**Clasificación mediante Deep Learning**

* Análisis multimodal que integra diferentes vistas de la mamografía.
* Clasificación automática de lesiones en dos categorías: benignas o malignas.
* Estimación de la probabilidad de clasificación (nivel de confianza del modelo).

**Generación y visualización de resultados**

* Generación automática de reporte de resultados.
* Panel de resultados interactivo accesible vía navegador web.

**Seguridad y confidencialidad de la información**

* Cumplimiento de normativas de privacidad aplicables (HTTPS).

Para visualizar el flujo del sistema véase [Anexo B. Diagramas de apoyo.](#FLUJO)

## Características del usuario.

El sistema será diseñado para ser utilizado por un público amplio, incluyendo a especialistas en el área de la salud como a personas sin formación médica.

El uso del sistema requiere únicamente conocimientos básicos de informática, tales como manejar un navegador web, cargar un archivo y leer información desplegada en pantalla. No se necesita formación médica ni conocimientos en inteligencia artificial para interactuar con la aplicación.

Por otro lado, los usuarios pueden variar desde profesionales con experiencia clínica en la interpretación de imágenes médicas hasta personas cuya experiencia se limita al uso cotidiano de programas y aplicaciones web en un equipo de cómputo. En cualquier caso, el sistema ofrecerá una interfaz sencilla y uniforme. Así mismo, no se requiere experiencia técnica avanzada. La interacción con el software se limita a operaciones simples como subir imágenes en formatos permitidos (JPG y DICOM) y consultar el resultado generado por el sistema.

De manera general, el sistema estará diseñado para una usabilidad general, garantizando accesibilidad y facilidad de uso independiente de nivel de conocimiento y/o experiencias previas del usuario.

## Restricciones generales.

En esta sección se especifican los aspectos que podrían limitar el desarrollo y/o operación del producto de software, tales como:

* + 1. Políticas regulatorias.

De acuerdo con la norma ISO 13485:2016 el software debe contar con un sistema de gestión de calidad documentada, mostrado en un manual de calidad. En este marco, se deberá documentar los modelos de entrenamiento, los datasets utilizados y el desempeño del software, incluyendo la sensibilidad del software.

Asimismo, el software deberá mostrar un mensaje de advertencia al usuario, indicando que la aplicación proporciona únicamente un prediagnóstico y que no reemplaza la evaluación de un médico.

No se tomará en cuenta la ley Federal de Protección de Datos personales, debido a que la aplicación web no almacenará ningún tipo de datos relacionados con el paciente o la carga de imágenes.

* + 1. Limitaciones de hardware.

Mínimo 16 GB en RAM y procesador i7.

* + 1. Interfaces con otras aplicaciones.

No aplica.

* + 1. Operaciones en paralelo.

No aplica.

* + 1. Funciones de auditoría.

No aplica.

* + 1. Funciones de control.

Dado que el sistema no contará con funciones de autenticación ni control de roles de usuario ya que será un software público habrá controles básicos relacionados con la operación de la aplicación web tales como:

* + - Validación de formato de imágenes permitidos (JPG, DICOM) antes de obtener un resultado.
    - Rechazo de archivos de imagen incompletas o no compatibles.
    - Manejo de mensajes de errores en caso de que la imagen no cumpla con los criterios establecidos.
    1. Requerimientos de lenguajes de alto nivel.

El desarrollo del sistema se llevará a cabo utilizando lenguajes de programación enfocados a Inteligencia Artificial y desarrollo web:

* + - **Frontend (interfaz del usuario):**
      * HTML5.
      * CCS3.
    - **Backend (función lógica del sistema):**
      * Javascript.
    - **Comunicación cliente-servidor:**
      * API REST con Python (Fast API).
    1. Protocolos de señal “estrechamiento de manos” (handshake) en la comunicación.

Toda comunicación entre el cliente y el servidor se realizará mediante el protocolo HTTPS, el cual implementa TLS (Transport Layer Security) para establecer una conexión segura.

* + 1. Criticidad de la aplicación.

El sistema tiene una criticidad de nivel medio debido a que su funcionamiento impacta directamente en el análisis de imágenes médicas relacionadas con la detección de cáncer de mama.

El sistema está diseñado únicamente como herramienta de apoyo y no como un sustituto del diagnóstico médico.

Una falla en el sistema no constituye por sí misma un riesgo vital inmediato, siempre que los resultados sean interpretados por un profesional de la salud.

* + 1. Consideración de seguridad.

No aplica.

## Presunciones y dependencias.

El desarrollo y operación del sistema estarán condicionados por ciertos factores externos que se considerarán al momento de la implementación, pero que podrían variar en el futuro y afectar el cumplimiento de los requerimientos aquí establecidos. Entre ellos se incluyen:

| **Presunción / Dependencia** | **Descripción** | **Impacto en el sistema** |
| --- | --- | --- |
| Navegadores web modernos | El usuario utilizará navegadores actualizados (Chrome, Firefox, Edge) compatibles con HTML5, CSS3 y JavaScript. | La interfaz podría no funcionar correctamente en navegadores obsoletos. |
| Acceso a internet estable | El sistema requiere conexión constante a internet para enviar y procesar imágenes en el servidor. | Sin internet no es posible cargar imágenes ni obtener resultados. |
| Compatibilidad de formatos de imagen | Se cargarán únicamente imágenes en formato JPG o DICOM compatibles con librerías (OpenCV, pydicom). | Formatos no estándar o corruptos impedirán el procesamiento correcto. |
| Librerías y frameworks externos | Dependencia de TensorFlow/PyTorch (Deep Learning), Flask/FastAPI (API) y librerías de imágenes. | Cambios, errores o descontinuación de estas herramientas afectarían la operación y mantenimiento. |
| Modelo entrenado | El sistema depende de un modelo de Deep Learning previamente entrenado y validado. | Cambios en el modelo requerirán ajustes en los requerimientos y pruebas adicionales. |

# Especificación de requerimientos.

## La presente sección detalla los requerimientos que debe cumplir el sistema de software “Aplicación web para la clasificación de cáncer de mama usando Deep Learning”. Estos requerimientos abarcan tanto las funcionalidades principales que deberá ofrecer la aplicación, como el desempeño esperado, las restricciones técnicas y regulatorias, y los atributos de calidad que permitirán evaluar el correcto funcionamiento del producto. La organización de esta sección tiene como objetivo facilitar la trazabilidad de cada requerimiento, garantizando que sean necesarios, claros, completos, verificables y consistentes.

## Requerimientos Funcionales.

En este apartado se describen las funciones específicas que debe ejecutar el sistema en relación con la entrada de imágenes médicas, su procesamiento y la generación de resultados. Los requerimientos funcionales establecen el qué debe hacer el sistema para cumplir con su propósito principal: apoyar la clasificación de imágenes de mamografía. Cada requerimiento se presenta en un formato estructurado que incluye entradas, salidas y las funciones necesarias para transformar las entradas en resultados verificables.

|  |  |
| --- | --- |
| Identificador de requerimiento: RF-01 | |
| Nombre corto: | Validación de formato de imagen |
| Estatus: | Pendiente |
| Descripción: | El sistema debe validar que las imágenes cargadas por el usuario sean exclusivamente imágenes en formatos estándar (DICOM o JPG) antes de ser procesadas y queda en responsabilidad del usuario el subir imágenes correctas de mamografías. |
| Necesidades que resuelve: | Evita errores de entrada y asegura que solo imágenes válidas sean analizadas por la red neuronal. |
| Métrica de satisfacción: | El 100% de los intentos de carga de archivos no válidos deben ser rechazados con un mensaje de error claro. |
| Entradas | Archivo de imagen en formato JPG o DICOM. |
| Salidas | Confirmación de validez (imagen aceptada) o rechazo con mensaje de error. |
| Funciones | Verificación de extensión y encabezado del archivo.  Validación de metadatos DICOM (cuando aplique).  Rechazo de imágenes no compatibles. |

|  |  |
| --- | --- |
| Identificador de requerimiento: RF-02 | |
| Nombre corto: | Preprocesamiento automático |
| Estatus: | Pendiente |
| Descripción: | El sistema debe aplicar automáticamente técnicas de preprocesamiento a las imágenes cargadas (normalización, eliminación de ruido, ajuste de contraste, reescalado). |
| Necesidades que resuelve: | Mejora la calidad de las imágenes para aumentar la precisión del modelo. |
| Métrica de satisfacción: | Todas las imágenes procesadas deben cumplir con contraste y tamaño definidos. |
| Entradas | Imagen mamográfica validada. |
| Salidas | Imagen preprocesada lista para análisis por el modelo. |
| Funciones | Normalización de niveles de gris.  Reducción de ruido mediante filtros.  Ajuste de contraste adaptativo.  Reescalado a tamaño estándar del modelo. |

|  |  |
| --- | --- |
| Identificador de requerimiento: RF-03 | |
| Nombre corto: | Soporte multimodal |
| Estatus: | Pendiente |
| Descripción: | El sistema debe analizar de manera igualitaria las vistas ya sean CC (cráneo-caudal) y MLO (medio-lateral oblicua) en cada mamografía. |
| Necesidades que resuelve: | Aumenta la precisión del análisis al integrar correlaciones bilaterales. |
| Métrica de satisfacción: | Para cada caso cargado, el sistema debe asegurar el análisis de las dos vistas estándar. |
| Entradas | Paquete de imágenes de la misma mama (CC y MLO). |
| Salidas | Resultado combinado de clasificación (benigno/maligno) con probabilidad. |
| Funciones | Lectura simultánea de múltiples vistas.  Extracción de características independientes por vista. |

## Requerimientos de desempeño.

|  |  |
| --- | --- |
| Identificador de requerimiento: RNF-01 | |
| Nombre corto: | Tiempo de respuesta |
| Estatus: | Pendiente |
| Descripción: | El tiempo de respuesta de la aplicación web deberá de ser máximo de 1 minuto con posibles variaciones dependiendo del modelo seleccionado. |
| Necesidades que resuelve: | El sistema cuenta con un tiempo de respuesta aceptable para mostrar los resultados. |
| Métrica de satisfacción: | Se cumple la métrica de tiempo de espera apta para el usuario. |

|  |  |
| --- | --- |
| Identificador de requerimiento: RNF-02 | |
| Nombre corto: | Clasificación con Deep Learning |
| Estatus: | Pendiente |
| Descripción: | El sistema debe clasificar las imágenes como *benignas* o *malignas* utilizando un modelo CNN previamente entrenado. |
| Necesidades que resuelve: | Apoya al radiólogo reduciendo la carga de trabajo y mejorando la precisión diagnóstica. |
| Métrica de satisfacción: | El modelo debe lograr al menos 85% de exactitud en pruebas internas sobre el conjunto de validación. |
| Entradas | Imagen mamográfica preprocesada. |
| Salidas | Clase de la lesión (*benigno* o *maligno*) y porcentaje de certeza. |
| Funciones | Extracción de características mediante red neuronal convolucional.  Capa de clasificación softmax para asignar probabilidad a cada clase.  Generación del resultado de predicción. |

## Requerimientos de la base de datos lógica.

No aplica

## Restricciones de diseño.

|  |  |
| --- | --- |
| Identificador de requerimiento: RNF-03 | |
| Nombre corto: | Especificaciones del diseño de la aplicación web |
| Estatus: | Pendiente |
| Descripción: | La aplicación web tiene un diseño no respomsivo la cual está disponible para que los usuarios accedan mediante un equipo de cómputo. |
| Necesidades que resuelve: | Se optimiza el desarrollo de la apliación web para un único dispositivo. |
| Métrica de satisfacción: | La interfaz se muestra de manera correcta en los equipos de cómputo. |

## Atributos

Existen algunos atributos del sistema de software que pueden servir como requerimientos. Es importante que los atributos requeridos sean perfectamente especificados para que su logro pueda ser verificado objetivamente. Algunos de ellos son los siguientes:

### Confiabilidad.

|  |  |
| --- | --- |
| Atributo 01 | |
| Nombre corto: | Precisión del modelo |
| Estatus: | Pendiente |
| Descripción: | La precisión del modelo deberá ser mínimo de 85% en la clasificación. |
| Necesidades que resuelve: | Ayudar al diagnóstico de cáncer de mama con un nivel de confianza aceptable. |
| Métrica de satisfacción: | Se cumple la métrica de precisión propuesta para el modelo. |

### Disponibilidad.

|  |  |
| --- | --- |
| Atributo 02 | |
| Nombre corto: | Disponibilidad de servidor |
| Estatus: | Pendiente |
| Descripción: | El servidor de alojamiento estará activo el 99% del tiempo. |
| Necesidades que resuelve: | Disposición de la aplicación web en cualquier momento. |
| Métrica de satisfacción: | Acceso a la aplicación con el servidor operando. |

### Accesibilidad.

|  |  |
| --- | --- |
| Atributo 03 | |
| Nombre corto: | Interfaz de usuario intuitiva |
| Estatus: | Pendiente |
| Descripción: | La interaz gráfica diseñada para la aplicación web será intutitiva y fácil de usar para los usuarios que accedan al sistema. |
| Necesidades que resuelve: | No se requiere de un manual de usuario o un tutorial para el manejo de la aplicación web. |
| Métrica de satisfacción: | Los usuarios interactuan de manera satisfactoria con la aplicación web. |

### Seguridad.

|  |  |
| --- | --- |
| Atributo 04 | |
| Nombre corto: | Seguridad (Protocolo HTTPS). |
| Estatus: | Pendiente |
| Descripción: | Toda interacción entre el cliente y la página web se realiza mediante el protocolo HTTPS otorgando una capa extra de seguridad. |
| Necesidades que resuelve: | Garantiza que los datos transmitidos (mamografías) estén cifradas evitando la retransmisión de la información |
| Métrica de satisfacción: | El 100% de las imágenes se cifran correctamente. |

### Mantenibilidad.

|  |  |
| --- | --- |
| Atributo 05 | |
| Nombre corto: | Modularización de sistema. |
| Estatus: | Pendiente |
| Descripción: | Se segmentará el sistema por módulos de funcionalidad por cada característica o componente de este. |
| Necesidades que resuelve: | Facilidad de mantenimiento, corrección o actualización en futuras versiones. |
| Métrica de satisfacción: | El software se adapta a actualizaciones posteriores. |

### Portabilidad.

|  |  |
| --- | --- |
| Atributo 06 | |
| Nombre corto: | Aplicación web multinavegador |
| Estatus: | Pendiente |
| Descripción: | La aplicación web será funcional en los navegadores web de Google Chrome, Safari, Microsoft Edge y Mozilla Firefox. |
| Necesidades que resuelve: | La aplicación web está disponible para la mayor parte de la población. |
| Métrica de satisfacción: | Correcto funcionamiento de la aplicación web en los navegadores de Google Chrome, Safari, Microsoft Edge y Mozilla Firefox. |

## Requisitos de Interfaces externas.

No aplica

## Organización específica de los requerimientos.

No existe una organización óptima de los requerimientos de software; algunas de las formas de organización sugeridas son las siguientes:

### Modo del sistema. (No aplica)

Los sistemas de software tienen distintos modos de operación. Por ejemplo, un sistema de control podría tener los modos de entrenamiento, normal, o de emergencia. La selección dependerá de la forma en que las interfaces o desempeño dependan del modo de operación.

### Clases de usuario. (No aplica)

Algunos sistemas proveen distintas funcionalidades para diferentes tipos de usuarios, por ejemplo, funciones específicas al nivel operativo, de supervisión, se aprobación, etc.

### Objetos. (No aplica)

Esta forma de organización va de la mano con el paradigma de la programación orientada a objetos.

### Característica.

Para este proyecto, los requerimientos se organizan **por característica**, dado que el sistema se compone de funciones claramente diferenciadas que responden a un flujo de trabajo definido. Cada característica representa un bloque funcional que abarca desde la interacción inicial del usuario hasta la obtención de los resultados.

Este documento incluye en su sección [*Anexo 1*](#_Anexo_1._Plantillas) las diferentes plantillas de organización de requerimientos propuestas, tales como organización por modo del sistema, por clases de usuario, por objeto, por característica, por eventos o por jerarquía funcional.

Para el caso de este proyecto, se ha optado por la [organización por características](#_Por_característica.), dado que permite estructurar los requerimientos en bloques funcionales claramente diferenciados: carga de imágenes, preprocesamiento, clasificación mediante Deep Learning, soporte multimodal y visualización de resultados.

Así bien, podemos definir las características principales de la siguiente manera:

1. **Carga de imágenes médicas**
   * Permite la subida de archivos en formato DICOM o JPG.
   * Valida la integridad y compatibilidad de los archivos cargados.
2. **Preprocesamiento automático**
   * Normaliza la resolución y ajusta la calidad de las imágenes.
   * Aplica técnicas de eliminación de ruido y mejora de contraste.
   * Reescala las imágenes al tamaño estándar requerido por el modelo.
3. **Clasificación con Deep Learning**
   * Analiza las imágenes mediante un modelo previamente entrenado.
   * Genera una predicción binaria (benigno/maligno).
   * Incluye el porcentaje de confianza asociado al resultado.
4. **Soporte multimodal**
   * Procesa de manera conjunta las vistas estándar CC y MLO.
   * Combina los resultados parciales para obtener una clasificación integrada.
5. **Visualización de resultados**
   * Muestra de forma clara el diagnóstico asistido (clase y probabilidad).
   * Genera un reporte accesible a través del navegador web.

### Estímulos (Eventos). (No aplica)

Algunos sistemas pueden ser organizados a través de la descripción de las funciones en términos de eventos.

### Respuesta. (No aplica)

Algunos sistemas pueden ser organizados a través de la descripción de las funciones tomando como base la generación de respuestas.

### Jerarquía funcional. (No aplica)

Cuando ninguna de las anteriores formas de organización es aplicable al sistema de software, la totalidad de las funciones pueden ser organizadas en una jerarquía funcional considerando ya sea entradas comunes, salidas comunes, o acceso de datos internos comunes. Los diagramas de flujo de datos y diccionarios de datos pueden ser usados para mostrar las interrelaciones entre las funciones y los datos.

### Cometarios adicionales.

# Anexo 1. Plantillas de organización de SRS.

Según las formas de organización explicadas en el presente documento, existen varias opciones para estructurar dicha organización los requerimientos de software, es decir la sección tres. A continuación, se presentan varias opciones de estructuración que corresponden a cada forma de organización.

## Por modo del sistema.

Si se elige está forma de organización, hay 2 opciones de estructuración:

### opción 1.

1. Especificación de requerimientos.
   1. Requerimientos de interfaces externas.
      1. Interfaces de usuario.
      2. Interfaces de hardware.
      3. Interfaces de software.
      4. Interfaces de comunicaciones.
   2. Requerimientos funcionales.
      1. Modo 1.
         1. Requerimientos funcionales.
         2. …
      2. Modo 2.
         1. Requerimientos funcionales.
         2. …
      3. Modo m.
         1. Requerimientos funcionales.
   3. Requerimientos de desempeño.
   4. Restricciones de diseño.
   5. Atributos de sistema de software.
   6. Otros requerimientos.

### opción 2

1. Especificación de requerimientos.
   1. Requerimientos funcionales.
      1. Modo 1.
         1. Interfaces externas.
            1. Interfaces de usuario.
            2. Interfaces de hardware.
            3. Interfaces de software.
            4. Interfaces de comunicación.
         2. Requerimientos funcionales.
            1. Requerimiento funcional 1.
            2. …
            3. Requerimiento funcional n.
         3. Requerimientos de desempeño.
      2. Modo 2.
         1. …
   2. Restricciones de diseño.
   3. Atributos del sistema de software.
   4. Otros requerimientos.

## Por clases de usuario.

Para esta opción, esta es la sugerencia:

1. Especificación de requerimientos.
   1. Requerimientos de interfaces externas.
      1. Interfaces de usuario.
      2. Interfaces de hardware.
      3. Interfaces de software.
      4. Interfaces de comunicaciones.
   2. Requerimientos funcionales.
      1. Clase de usuario 1.
         1. Requerimiento funcional 1.
         2. …
         3. Requerimiento funcional n.
      2. Clase de usuario 2.
         1. …
      3. Clase de usuario n.
         1. …
   3. Requerimientos de desempeño.
   4. Restricciones de diseño.
   5. Atributos del sistema de software.
   6. Otros requerimientos.

## Por objeto.

Para esta opción, esta es la sugerencia:

1. Especificación de requerimientos (casos de uso).
   1. Requerimientos de interfaces externas.
      1. Interfaces de usuario.
      2. Interfaces de hardware.
      3. Interfaces de software.
      4. Interfaces de comunicaciones.
   2. Clases / objetos.
      1. Clase / objeto 1.
         1. Atributos.
            1. Atributo 1
            2. …
            3. Atributo n
            4. …
      2. Clase / objeto n
         1. …
   3. Requerimientos de desempeño.
   4. Restricciones de diseño.
   5. Atributos del sistema de software.
   6. Otros requerimientos.

## Por característica.

Se muestra la organización general de esta opción de estructuración. Además de los anexos propios del proyecto:

1. Especificación de requerimientos.
   1. Requerimientos de interfaces externas.
      1. Interfaces de usuario.
      2. Interfaces de hardware.
      3. Interfaces de software.
      4. Interfaces de comunicaciones.
   2. El sistema ofrece:
      1. característica 1
         1. Introducción / propósito de la característica 1
         2. Secuencia de estímulo / respuesta
         3. Requisitos funcionales asociados
            1. Requisito funcional 1
            2. …
            3. Requisito funcional *n*.
      2. Característica 2
      3. …
      4. Característica *n*
   3. Requerimientos de desempeño.
   4. Restricciones de diseño.
   5. Atributos del sistema de software.
   6. Otros requerimientos.

## **Anexo A. Ejemplos de entrada y salida del sistema**

## Ejemplo de entrada: Mamografía en formato DICOM y JPG.

## Ejemplo de salida: Resultado de clasificación (benigno/maligno) junto con el porcentaje de confianza directamente en la pantalla.

**Anexo B. Diagramas de apoyo**

Se muestra el diagrama de flujo general del sistema que permite la identificación y clasificación de cáncer de mama.

Interfaz de usuario gráfica, Word

El contenido generado por IA puede ser incorrecto.

## Por estímulo (evento).

Para esta opción, esta es la sugerencia:

1. Especificación de requerimientos.
   1. Requerimientos de interfaces externas.
      1. Interfaces de usuario.
      2. Interfaces de hardware.
      3. Interfaces de software.
      4. Interfaces de comunicaciones.
   2. Requisitos funcionales
      1. estímulo 1
         1. Requisito funcional 1
         2. …
         3. Requisito funcional *n*.
      2. estímulo 2
      3. …
      4. estímulo *n*
   3. Requerimientos de desempeño.
   4. Restricciones de diseño.
   5. Atributos del sistema de software.
   6. Otros requerimientos.

## Por jerarquía funcional

Para esta opción, esta es la sugerencia:

1. Especificación de requerimientos.
   1. Requerimientos de interfaces externas.
      1. Interfaces de usuario.
      2. Interfaces de hardware.
      3. Interfaces de software.
      4. Interfaces de comunicaciones.
   2. Requisitos funcionales
      1. Fluido de la información
         1. Flujo de datos diagrama 1
            1. Entidades de los datos
            2. Procesos pertinentes
            3. Topología
         2. Flujo de datos diagrama 2
            1. Entidades de los datos
            2. Procesos pertinentes
            3. Topología
         3. Flujo de datos diagrama *n*
            1. Entidades de los datos
            2. Procesos pertinentes
            3. Topología
      2. Descripción del proceso
         1. Proceso 1
            1. Entidades de datos de entrada
            2. Algoritmo o fórmula del proceso
            3. Entidades de los datos afectados
         2. Proceso 2
            1. Entidades de datos de entrada
            2. Algoritmo o fórmula del proceso
            3. Entidades de los datos afectados
         3. Proceso *n*
            1. Entidades de datos de entrada
            2. Algoritmo o fórmula del proceso
            3. Entidades de los datos afectados
      3. Especificaciones de construcción de datos
         1. Estructura 1
            1. Tipo del registro
            2. Tipos constituyentes
         2. Estructura 2
            1. Tipo del registro
            2. Tipos constituyentes
         3. Estructura *n*
            1. Tipo del registro
            2. Tipos constituyentes
      4. Diccionario de datos
         1. Elemento de datos 1
            1. Nombre
            2. Representación
            3. Unidad/formato
            4. Precisión / exactitud
            5. alcance
         2. Elemento de datos 2
            1. Nombre
            2. Representación
            3. Unidad/formato
            4. Precisión / exactitud
            5. alcance
         3. Elemento de datos *n*
            1. Nombre
            2. Representación
            3. Unidad/formato
            4. Precisión / exactitud
            5. alcance
   3. Requerimientos de desempeño.
   4. Restricciones de diseño.
   5. Atributos del sistema de software.
   6. Otros requerimientos.

## Por combinación de diferentes organizaciones

Para esta opción, esta es la sugerencia:

1. Especificación de requerimientos.
   1. Requerimientos de interfaces externas.
      1. Interfaces de usuario.
      2. Interfaces de hardware.
      3. Interfaces de software.
      4. Interfaces de comunicaciones.
   2. Requisitos funcionales
      1. Clase de usuario 1
         1. Característica 1
            1. Introducción / propósito de la característica
            2. Secuencia de estímulo / respuesta
            3. Requisitos funcionales asociados
         2. Característica 1.2
            1. Introducción / propósito de la característica
            2. Secuencia de estímulo / respuesta
            3. Requisitos funcionales asociados
         3. Característica *n*
            1. Introducción / propósito de la característica
            2. Secuencia de estímulo / respuesta
            3. Requisitos funcionales asociados
      2. Clase de usuario 2
      3. Clase de usuario *n*
   3. Requerimientos de desempeño.
   4. Restricciones de diseño.
   5. Atributos del sistema de software.
   6. Otros requerimientos.